Article

The International Journal of Robotics Research
1–23

DOI: 10.1177/0278364919853361
journals.sagepub.com/home/ijr
SAGE

Robocentric visual–inertial odometry

Zheng Huai and Guoquan Huang

Abstract

In this paper, we propose a novel robocentric formulation of the visual–inertial navigation system (VINS) within a sliding-window filtering framework and design an efficient, lightweight, robocentric visual–inertial odometry *(R-VIO) algorithm for consistent motion tracking even in challenging environments using only a monocular camera and a six-axis inertial measurement unit (IMU). The key idea is to deliberately reformulate the VINS with respect to a moving local frame, rather than a fixed global frame of reference as in the standard world-centric VINS, in order to obtain relative motion estimates of higher accuracy for updating global pose. As an immediate advantage of this robocentric formulation, the proposed R-VIO can start from an arbitrary pose, without the need to align the initial orientation with the global gravitational direction. More importantly, we analytically show that the linearized robocentric VINS does not undergo the observability mismatch issue as in the standard world-centric counterparts that has been identified in the literature as the main cause of estimation inconsistency. Furthermore, we investigate in depth the special motions that degrade the performance in the world-centric formulation and show that such degenerate cases can be easily compensated for by the proposed robocentric formulation, without resorting to additional sensors as in the world-centric formulation, thus leading to better robustness. The proposed R-VIO algorithm has been extensively validated through both Monte Carlo simulation and real-world experiments with different sensing platforms navigating in different environments, and shown to achieve better (or competitive at least) performance than the state-of-the-art VINS, in terms of consistency, accuracy, and efficiency.*

Keywords

visual–inertial odometry, robocentric formulation, extended Kalman filter, observability analysis, estimation consistency

1. Introduction

Enabling high-precision, energy-efficient, and robust motion tracking in 3D space on mobile platforms (e.g., robots) with minimal sensing holds potentially huge implications in many real applications ranging from mobile augmented reality to autonomous driving. To this end, inertial navigation offers a classical 3D localization solution which utilizes an inertial measurement unit (IMU) measuring the three-degree-of-freedom (DOF) angular velocity and 3-DOF linear acceleration of the sensing platform upon which it is rigidly attached. Typically, IMU works at high frequency (e.g., 100–1,000 Hz) which makes it suitable to sense highly dynamic motion, while due to the existence of sensor noise and bias purely integrating IMU measurements may easily result in unusable motion estimates. This necessitates the use of aiding information from *at least* a single camera to reduce the accumulated inertial navigation drifts, which results in the well-known visual–inertial navigation system (VINS).

Over the past decade, significant progress has been witnessed in the research domain of VINS, including the visual–inertial simultaneous localization and mapping (VI-SLAM) and the visual–inertial odometry (VIO), accordingly many different algorithms have been proposed to date (e.g., Mourikis and Roumeliotis, 2007; Jones and Soatto, 2011; Kelly and Sukhatme, 2011; Li and Mourikis, 2013a; Leutenegger et al., 2015; Usenko et al., 2016; Mur-Artal and Tardós, 2017; Qin et al., 2018, and references therein). However, almost all these algorithms are developed based on the standard *world-centric* VINS formulation; that is, to directly estimate absolute motion of the sensing platform with respect to a fixed, global frame of reference, such as the most widely used, Earth-centered Earth-fixed (ECEF) or north–east–down (NED) frame. In order to achieve accurate 3D localization such

Department of Mechanical Engineering, University of Delaware, Newark, DE, USA

Corresponding author:
Zheng Huai, Department of Mechanical Engineering, University of Delaware,130 Academy Street, Spencer Lab, Newark, DE 19716, USA.
Email: zhuai@udel.edu

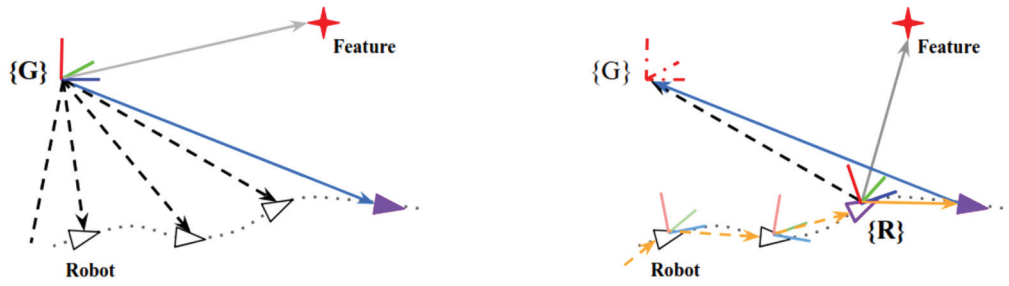


Fig. 1. World-centric versus robocentric formulation. The fixed global frame, $\{G\}$, is replaced by a local robot frame (e.g., the IMU frame), $\{R\}$, as the frame of reference of the VINS. Therefore, instead of estimating absolute motion in $\{G\}$ (blue), the relative motion of higher accuracy (yellow) is estimated with respect to $\{R\}$ and used for predicting and updating the global pose of robot.

world-centric VINS algorithms usually require a particular initialization procedure to estimate the sensor's starting pose in the fixed global frame of reference before the navigation, which, however, is hard to guarantee the accuracy in some cases (e.g., quick start, poor vision, or high latency of sensors). In addition, even the extended Kalman filter (EKF)-based VINS algorithms have the advantage of lower computational cost (e.g., Mourikis and Roumeliotis, 2007; Mourikis et al., 2009; Li and Mourikis, 2013a) in comparison with the batch optimization-based approaches whose relinearization incurs higher amount of computation (e.g., Leutenegger et al., 2015; Qin et al., 2018), they may become *inconsistent*, primarily due to the fact that the EKF linearized system has different observability properties from the corresponding original (nonlinear) system (Huang et al., 2009, 2010; Li and Mourikis, 2013a; Hesch et al., 2014b). To address this issue, the remedies include enforcing the correct observability constraints (Li and Mourikis, 2013a; Huang et al., 2014; Hesch et al., 2014b) or employing an invariant error representation (Zhang et al., 2017). However, one may ask: *Do we have to formulate the VINS in a world-centric form?* The answer is *no*. Intuitively, consider how humans navigate: we might not remember the starting pose after traveling a long distance, while knowing well the relative motion within a recent, short time interval. Bearing this in mind, we may relax this fixed global frame of the VINS, instead, choosing a moving local frame as the reference to better estimate the relative motion which can be used for global pose update.

Historically, the usage of sensor-centered formulation for mobile robot localization can be traced back to sonar/laser-based 2D SLAM in indoor environments (Tardós et al., 2002; Castellanos et al., 2007) where the global frame was included as one of the map points being observed from the robot. The outputs of odometry were fused with the sonar/laser-scanner observations via an EKF to estimate robot's relative motion which was then used to update the global pose of robot and change the local frame of reference for map joining through a composition step, as moving forward. With a similar idea, Civera et al. (2009) used a camera-centered formulation to show the potential of fusing the visual information with the proprioceptive information, such as given the rotational and linear velocity measurements. By performing mapping with respect to the (local) robot frames, these robocentric EKF-based SLAM algorithms were shown to be able to improve estimation consistency. It should be noticed that a different robocentric formulation was recently introduced in an EKF-based VINS (Bloesch et al., 2015, 2017). In this system, by representing the state in the current IMU (or camera) frame, consisting of the IMU navigation states, a set of observed landmarks' positions, as well as the camera-to-IMU spatial calibration parameters, the visual and inertial measurements were fused in a direct fashion by employing a photometric residual in the EKF update. In contrast to Castellanos et al. (2007) and Civera et al. (2009), this method stuck to estimating the (inverted) transformation between a fixed global frame and the current IMU frame via a standard iterated EKF, due to simply reversing the reference frame of the filter's states referring to the world-centric formulation. As a result, it did not perform the composition step to shift the frame of reference, however, incurring higher computational cost as it performs iterative linearization on both landmarks (features) and IMU states at each time step when doing EKF update.

In this paper, we introduce a new robocentric formulation of the VINS with respect to a local IMU frame of reference, as illustrated in Figure 1. Specifically, in contrast to Castellanos et al. (2007), Civera et al. (2009), and Bloesch et al. (2015, 2017) which keep the features (or map points) in the state vector and would inevitably encounter the issue of ever-increasing computational cost as more features are observed and included, we devote to an efficient robocentric filtering-based framework, akin to the multi-state constraint Kalman filter (MSCKF) (Mourikis and Roumeliotis, 2007). In the proposed estimator, stochastic cloning (Roumeliotis and Burdick, 2002) is employed so that hundreds of features can be processed simultaneously, instead only a small number of relative poses, that are between consecutive robot locations from which the features are observed, are kept in the state vector, hence significantly reducing the computational cost. More importantly, as the proposed robocentric VIO does not suffer from the observability mismatch issue that exists in the world-centric

counterparts, it possesses better consistency. In particular, the main contributions of this paper include the following:

- We propose a novel robocentric EKF-VINS formulation by reformulating the VINS problem with respect to a local IMU frame, which treats the global frame as the only "feature" and includes the local gravity (i.e., with respect to the local frame of reference) into the state vector. The local frame of reference is shifted at every image time through a *composition* step, and the relative motion estimate between the consecutive local frames is used for updating the global pose estimate.
- We develop an efficient and robust R-VIO algorithm within a sliding-window filtering framework, where a constant-size window of *relative poses*, instead of the observed features (or the global poses) by noting that the relative pose is of zero kinematics which avoids the linearization errors introduced by the change of frame of reference, is included in the filter's state vector and jointly estimated by tightly fusing the camera and IMU measurements in a local frame of reference. As such, a tailored *inverse depth*-based measurement model is proposed to fully utilize this state configuration, such that a dense connection is established between feature measurements and the state of the R-VIO considering the geometric relation between feature and the poses from which it has been observed. It should be pointed out that even in the degenerate cases (e.g., motionless) this model is still able to fuse the bearing information from the features, which is particularly useful in real applications.
- We study in-depth the observability properties of the proposed R-VIO's system model, which analytically shows that it has an *invariant* unobservable subspace, that is independent of the R-VIO linearization points, under generic motions. Thus, the resulting linearized robocentric system does not undergo the observability mismatch which has been identified as the main cause of inconsistency (Huang et al., 2010; Li and Mourikis, 2013a; Hesch et al., 2014b). More importantly, from the analysis, the proposed R-VIO is shown to have not only correct unobservable dimension but also constant unobservable directions, thus significantly improving the consistency of estimation. Furthermore, we investigate the unobservable directions under degenerate motions and show that the potential performance degradations existing in the world-centric formulation can easily be compensated for by the proposed R-VIO *without* using the information from any additional sensor.
- We perform extensive tests through both Monte Carlo simulation and real-world experiments running sensor data collected with different mobile platforms from the micro aerial vehicles (MAVs) flying indoor to ground vehicle driving in dynamic traffic environment. To further benefit our community, we *open source* our code at http://github.com/rpng/R-VIO.

The reminder of the paper is organized as follows: after a brief overview of related work in the next section, we present the proposed R-VIO algorithm in detail in Section 3, which is followed by a detailed observability analysis in Section 4. In Sections 5 and 6, we demonstrate the superior performance of the proposed algorithm through both simulations and real experiments. Finally, Section 7 concludes the work in this paper, as well as suggesting possible future research directions.

2. Related work

As mentioned earlier, the VINS algorithms generally include VI-SLAM (e.g., Lupton and Sukkarieh, 2012; Leutenegger et al., 2015; Qin et al., 2018) and VIO (e.g., Mourikis and Roumeliotis, 2007; Li and Mourikis, 2013a; Forster et al., 2015). The former jointly estimates the feature positions and camera/IMU pose that together form the system state vector, whereas the latter does not include the features in the state vector but utilizes the visual measurements to impose motion constraints for the camera/IMU pose estimation. In general, by performing mapping the VI-SLAM is able to achieve better accuracy from the feature map and/or possible loop closures while incurring higher computational cost compared with the VIO, although different approaches were proposed to address this issue (e.g., Leutenegger et al., 2015; Usenko et al., 2016; Mur-Artal and Tardós, 2017; Qin et al., 2018). While there were also efforts to integrate VIO and SLAM (e.g., Mourikis et al., 2009; Li and Mourikis, 2013b), in this paper we focus on the design of lightweight VIO that can either serve as a stand-alone motion estimator, or as an essential building block for large-scale navigation systems.

There are different schemes available for the VINS to fuse the visual and inertial measurements, which can be broadly categorized into the loosely-coupled and the tightly-coupled. The former processes the visual and inertial measurements separately to infer their own motion constraints which are to be fused later (e.g., Kneip et al., 2011; Weiss and Siegwart, 2011; Indelman et al., 2013). While this kind of method is computationally efficient, the decoupling of the visual and inertial constraints results in information loss. By contrast, the tightly-coupled approaches directly fuse the visual and inertial measurements within a single process and achieve higher accuracy (e.g., Mourikis and Roumeliotis, 2007; Li and Mourikis, 2013a; Forster et al., 2015; Leutenegger et al., 2015; Qin et al., 2018). As the embedded digital computing and sensing technology advance, the tightly-coupled VINS may even be able to run in real time on resource-constrained platforms, such as drones and smart phones, and the tightly-coupled method proposed in this paper is also shown to have such capability.

In particular, there exist two main approaches for tightly-coupled VINS estimation, the batch optimization-based and the EKF-based. Typically, bundle adjustment

(BA) (Triggs et al., 1999) is employed by the former to estimate the states involved in the constrains of all the available measurements by solving a nonlinear least-squares problem (e.g., Forster et al., 2015; Leutenegger et al., 2015; Qin et al., 2018). As the iterative linearization of nonlinear measurement models is carried out at each time step, such methods would incur higher computational cost as compared with the EKF-based counterparts (e.g., Mourikis and Roumeliotis, 2007; Li and Mourikis, 2013a). However, as mentioned previously, the standard world-centric EKF-based VINS would suffer from estimation inconsistency which is mainly due to the issue of observability mismatch with the EKF linearization (Huang et al., 2010; Hesch et al., 2014b); while based on world-centric formulation, the observability-constrained (OC)-VINS has been proposed to address this issue (Li and Mourikis, 2013a; Huang et al., 2014; Hesch et al., 2014b). In addition, for the EKF-based method Bloesch et al. (2015, 2017) recently proposed a VINS using a robocentric representation of the filter state including also the features, which followed the traditional EKF-SLAM approach to estimate absolute sensor motion while the evolution of the features in local frame introduced extra linearization errors. By contrast, we, inspired by the robocentric mapping that improved the EKF consistency in 2D SLAM (Castellanos et al., 2007), propose a new robocentric formulation within a sliding-window filtering-based VIO framework and based on that develop the R-VIO algorithm which, in what follows, is shown to have correct observability properties, and therefore improve estimation consistency and accuracy.

3. Estimator design

Consider a mobile platform equipped with an IMU and a single camera navigating in a 3D environment. In contrast to the standard world-centric VINS using a fixed global frame of reference $\{G\}$, in the proposed robocentric formulation, the frame $\{I\}$ affixed to IMU is set to be an immediate, local frame of reference for navigation, termed $\{R\}$. As a result, the global frame $\{G\}$ (or the first local frame of reference, $\{R_0\}$) turns into a stationary object from the perspective of $\{R\}$; and during the navigation, $\{R\}$ can be transferred from one IMU frame to another. In this section, we deliberately reformulate the VINS problem with respect to such a moving local, rather than the fixed global, frame of reference, and present in detail the proposed R-VIO algorithm with a sliding-window filtering framework.

3.1. State vector

The state vector of the proposed robocentric VINS consists of two parts: (i) the global state that maintains the motion information of the starting frame $\{G\}$ (e.g., $\{R_0\}$), and (ii) the IMU state that characterizes the motion from the local frame of reference to the current IMU frame. In particular, at time $t_\tau \in [t_k, t_{k+1}]$ the state expressed in the local frame of reference, $\{R_k\}$, is given by[1]

$$\begin{aligned} {}^{R_k}\mathbf{x}_\tau &= \begin{bmatrix} {}^{R_k}\mathbf{x}_G^\top & {}^{R_k}\mathbf{x}_{I_\tau}^\top \end{bmatrix}^\top \\ {}^{R_k}\mathbf{x}_G &= \begin{bmatrix} {}^{k}_{G}\bar{q}^\top & {}^{R_k}\mathbf{p}_G^\top & {}^{R_k}\mathbf{g}^\top \end{bmatrix}^\top \\ {}^{R_k}\mathbf{x}_{I_\tau} &= \begin{bmatrix} {}^{\tau}_{k}\bar{q}^\top & {}^{R_k}\mathbf{p}_{I_\tau}^\top & \mathbf{v}_{I_\tau}^\top & \mathbf{b}_{g_\tau}^\top & \mathbf{b}_{a_\tau}^\top \end{bmatrix}^\top \end{aligned} \tag{1}$$

where ${}^{k}_{G}\bar{q}$ is the 4×1 unit quaternion (Breckenridge, 1979) representing the orientation of $\{G\}$ in $\{R_k\}$, and ${}^{R_k}\mathbf{p}_G$ is the position of $\{G\}$ in $\{R_k\}$; ${}^{\tau}_{k}\bar{q}$ and ${}^{R_k}\mathbf{p}_{I_\tau}$ are the relative rotation and translation from $\{R_k\}$ to the current IMU frame $\{I_\tau\}$; $\mathbf{v}_{I_\tau}$ is the local velocity expressed in $\{I_\tau\}$, and $\mathbf{b}_{g_\tau}$ and $\mathbf{b}_{a_\tau}$ denote the IMU's gyroscope and accelerometer biases, respectively. It should be noted that the local gravity, ${}^{R_k}\mathbf{g}$, is also included as part of freedom of the proposed system and modeled as a 3×1 vector of known magnitude (e.g., 9.81). The corresponding error state is then given by

$$\begin{aligned} {}^{R_k}\tilde{\mathbf{x}}_\tau &= \begin{bmatrix} {}^{R_k}\tilde{\mathbf{x}}_G^\top & {}^{R_k}\tilde{\mathbf{x}}_{I_\tau}^\top \end{bmatrix}^\top \\ {}^{R_k}\tilde{\mathbf{x}}_G &= \begin{bmatrix} \delta\boldsymbol{\theta}_G^\top & {}^{R_k}\tilde{\mathbf{p}}_G^\top & {}^{R_k}\tilde{\mathbf{g}}^\top \end{bmatrix}^\top \\ {}^{R_k}\tilde{\mathbf{x}}_{I_\tau} &= \begin{bmatrix} \delta\boldsymbol{\theta}_\tau^\top & {}^{R_k}\tilde{\mathbf{p}}_{I_\tau}^\top & \tilde{\mathbf{v}}_{I_\tau}^\top & \tilde{\mathbf{b}}_{g_\tau}^\top & \tilde{\mathbf{b}}_{a_\tau}^\top \end{bmatrix}^\top \end{aligned} \tag{2}$$

In particular, the error quaternion is defined by $\bar{q} = \delta\bar{q} \otimes \hat{\bar{q}}$:

$$\delta\bar{q} \simeq \begin{bmatrix} \frac{1}{2}\delta\boldsymbol{\theta}^\top & 1 \end{bmatrix}^\top, \quad \mathbf{C}(\delta\bar{q}) = \mathbf{I}_3 - \lfloor \delta\boldsymbol{\theta} \times \rfloor \tag{3}$$

where $\otimes$ denotes the quaternion multiplication, $\delta\bar{q}$ is the error quaternion associated with the 3-DOF error angle $\delta\boldsymbol{\theta}$, and $\mathbf{C}(\cdot)$ denotes a 3×3 rotation matrix with $\lfloor \cdot \times \rfloor$ being the skew-symmetric operator (Trawny and Roumeliotis, 2005).

At time-step k when the corresponding IMU frame, $\{I_k\}$, becomes the frame of reference (i.e., $\{R_k\}$), a window of the relative poses between the last N local frames of reference is included in the state vector, as

$$\begin{aligned} \hat{\mathbf{x}}_k &= \begin{bmatrix} {}^{R_k}\hat{\mathbf{x}}_k^\top & \hat{\mathbf{w}}_k^\top \end{bmatrix}^\top \\ \hat{\mathbf{w}}_k &= \begin{bmatrix} {}^{2}_{1}\hat{\bar{q}}^\top & {}^{R_1}\hat{\mathbf{p}}_{R_2}^\top & \cdots & {}^{N}_{N-1}\hat{\bar{q}}^\top & {}^{R_{N-1}}\hat{\mathbf{p}}_{R_N}^\top \end{bmatrix}^\top \end{aligned} \tag{4}$$

where ${}^{i}_{i-1}\hat{\bar{q}}$ and ${}^{R_{i-1}}\hat{\mathbf{p}}_{R_i}$ represent the relative rotation and translation from $\{R_{i-1}\}$ to $\{R_i\}$, $i=2,\ldots,N$. To keep a state vector of constant size over time, we manage $\mathbf{w}$ in a sliding-window fashion; that is, to marginalize out the oldest relative pose once a new one is included in the window. Accordingly, the augmented error state is given by

$$\begin{aligned} \tilde{\mathbf{x}}_k &= \begin{bmatrix} {}^{R_k}\tilde{\mathbf{x}}_k^\top & \tilde{\mathbf{w}}_k^\top \end{bmatrix}^\top \\ \tilde{\mathbf{w}}_k &= \begin{bmatrix} \delta\boldsymbol{\theta}_2^\top & {}^{R_1}\tilde{\mathbf{p}}_{R_2}^\top & \cdots & \delta\boldsymbol{\theta}_N^\top & {}^{R_{N-1}}\tilde{\mathbf{p}}_{R_N}^\top \end{bmatrix}^\top \end{aligned} \tag{5}$$

3.2. Propagation

We first present the motion model for the robocentric state ${}^{R_k}\mathbf{x}_\tau$ (see (1)), then extend it to the augmented state $\mathbf{x}_\tau$ (see (4)). Note that during the time interval $[t_k, t_{k+1}]$ the global frame is static with respect to the local frame of reference, then ${}^{R_k}\dot{\tilde{\mathbf{x}}}_G = \mathbf{0}_{9\times 1}$. For the IMU state ${}^{R_k}\mathbf{x}_{I_\tau}$, we introduce a locally parameterized kinematic model:

$$\begin{aligned} {}^{\tau}_{k}\dot{\bar{q}} &= \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega})\,{}^{\tau}_{k}\bar{q}, \quad {}^{R_k}\dot{\mathbf{p}}_{I_\tau} = \mathbf{C}({}^{\tau}_{k}\bar{q})^\top \mathbf{v}_{I_\tau}, \\ \dot{\mathbf{v}}_{I_\tau} &= {}^{\tau}\mathbf{a} - \lfloor \boldsymbol{\omega}\times \rfloor \mathbf{v}_{I_\tau}, \quad \dot{\mathbf{b}}_g = \mathbf{n}_{wg}, \quad \dot{\mathbf{b}}_a = \mathbf{n}_{wa} \end{aligned} \tag{6}$$

where $\mathbf{n}_{wg}\sim\mathcal{N}(\mathbf{0}, \sigma^2_{wg}\mathbf{I}_3)$ and $\mathbf{n}_{wa}\sim\mathcal{N}(\mathbf{0}, \sigma^2_{wa}\mathbf{I}_3)$ are the zero-mean white Gaussian noises that drive the IMU biases, and $\boldsymbol{\omega}$ and ${}^{\tau}\mathbf{a}$ are the angular velocity and linear acceleration expressed in $\{I_\tau\}$, respectively. In addition, for $\boldsymbol{\omega} = [\omega_x, \omega_y, \omega_z]^\top$, we have

$$\boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} -\lfloor \boldsymbol{\omega}\times \rfloor & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^\top & 1 \end{bmatrix}, \quad \lfloor \boldsymbol{\omega}\times \rfloor = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}$$

Typically, IMU provides the gyroscope and accelerometer measurements, $\boldsymbol{\omega}_m$ and $\mathbf{a}_m$, expressed in the sensor frame:

$$\boldsymbol{\omega}_m = \boldsymbol{\omega} + \mathbf{b}_g + \mathbf{n}_g \tag{7}$$

$$\mathbf{a}_m = {}^{I}\mathbf{a} + {}^{I}\mathbf{g} + \mathbf{b}_a + \mathbf{n}_a \tag{8}$$

where $\mathbf{n}_g\sim\mathcal{N}(\mathbf{0}, \sigma^2_g\mathbf{I}_3)$ and $\mathbf{n}_a\sim\mathcal{N}(\mathbf{0}, \sigma^2_a\mathbf{I}_3)$ are the zero-mean white Gaussian sensor noises, and ${}^{I}\mathbf{g}$ characterizes the gravity effects on the sensor frame.

Evaluating (6) at the current state estimate, ${}^{R_k}\hat{\mathbf{x}}_{I_\tau}$, yields the following continuous-time IMU state propagation:

$$\begin{aligned} {}^{\tau}_{k}\dot{\hat{\bar{q}}} &= \frac{1}{2}\boldsymbol{\Omega}(\hat{\boldsymbol{\omega}})\,{}^{\tau}_{k}\hat{\bar{q}}, \quad {}^{R_k}\dot{\hat{\mathbf{p}}}_{I_\tau} = {}^{\tau}_{k}\mathbf{C}^\top_{\hat{\bar{q}}}\hat{\mathbf{v}}_{I_\tau}, \\ \dot{\hat{\mathbf{v}}}_{I_\tau} &= \hat{\mathbf{a}} - {}^{\tau}\hat{\mathbf{g}} - \lfloor \hat{\boldsymbol{\omega}}\times \rfloor \hat{\mathbf{v}}_{I_\tau}, \quad \dot{\hat{\mathbf{b}}}_g = \mathbf{0}_{3\times 1}, \quad \dot{\hat{\mathbf{b}}}_a = \mathbf{0}_{3\times 1} \end{aligned} \tag{9}$$

where for brevity we have defined $\hat{\boldsymbol{\omega}} = \boldsymbol{\omega}_m - \hat{\mathbf{b}}_g$ and $\hat{\mathbf{a}} = \mathbf{a}_m - \hat{\mathbf{b}}_a$, ${}^{\tau}_{k}\mathbf{C}_{\hat{\bar{q}}} = \mathbf{C}({}^{\tau}_{k}\hat{\bar{q}})$ and ${}^{\tau}\hat{\mathbf{g}} = {}^{\tau}_{k}\mathbf{C}_{\hat{\bar{q}}}\,{}^{R_k}\hat{\mathbf{g}}$. Accordingly, based on (6) and (9), we have continuous-time robocentric error-state model in the form of

$${}^{R_k}\dot{\tilde{\mathbf{x}}}_\tau = \mathbf{F}\,{}^{R_k}\tilde{\mathbf{x}}_\tau + \mathbf{G}\mathbf{n} \tag{10}$$

where $\mathbf{n} = [\mathbf{n}_g^\top \;\; \mathbf{n}_{wg}^\top \;\; \mathbf{n}_a^\top \;\; \mathbf{n}_{wa}^\top]^\top$ denotes the input noise vector, $\mathbf{F}$ is the robocentric error-state transition matrix, and $\mathbf{G}$ is the input noise Jacobian, respectively:

$$\mathbf{F} = \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & -\lfloor \hat{\boldsymbol{\omega}}\times \rfloor & \mathbf{0}_3 & \mathbf{0}_3 & -\mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & -{}^{\tau}_{k}\mathbf{C}^\top_{\hat{\bar{q}}}\lfloor \hat{\mathbf{v}}_{I_\tau}\times \rfloor & \mathbf{0}_3 & {}^{\tau}_{k}\mathbf{C}^\top_{\hat{\bar{q}}} & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & -{}^{\tau}_{k}\mathbf{C}_{\hat{\bar{q}}} & -\lfloor {}^{\tau}\hat{\mathbf{g}}\times \rfloor & \mathbf{0}_3 & -\lfloor \hat{\boldsymbol{\omega}}\times \rfloor & -\lfloor \hat{\mathbf{v}}_{I_\tau}\times \rfloor & -\mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ -\lfloor \hat{\mathbf{v}}_{I_\tau}\times \rfloor & \mathbf{0}_3 & -\mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{11}$$

For implementing the robocentric EKF, the discrete-time propagation model is needed. First, the IMU state estimate, ${}^{R_k}\hat{\mathbf{x}}_{I_\tau}$, is obtained as follows:

(i) ${}^{\tau}_{k}\hat{\bar{q}}$ is obtained by performing zeroth-order quaternion integration (Trawny and Roumeliotis, 2005) with

$$\begin{aligned} {}^{\tau}_{k}\hat{\bar{q}} &= \int_{t_k}^{t_\tau} {}^{s}_{k}\dot{\hat{\bar{q}}}\, ds \\ &= \int_{t_k}^{t_\tau} \frac{1}{2}\boldsymbol{\Omega}(\hat{\boldsymbol{\omega}})\,{}^{s}_{k}\hat{\bar{q}}\, ds \\ &= \int_{t_k}^{t_\tau} \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}_m - \hat{\mathbf{b}}_g)\,{}^{s}_{k}\hat{\bar{q}}\, ds \end{aligned} \tag{12}$$

(ii) ${}^{R_k}\hat{\mathbf{p}}_{I_\tau}$ and ${}^{R_k}\hat{\mathbf{v}}_{I_\tau}$ can be computed respectively using the following integrations with respect to $\{R_k\}$, as

$$\begin{aligned} {}^{R_k}\hat{\mathbf{p}}_{I_\tau} &= \hat{\mathbf{v}}_{I_k}\Delta t + \int_{t_k}^{t_\tau}\int_{t_k}^{s} {}^{\mu}_{k}\mathbf{C}^\top_{\hat{\bar{q}}}\,{}^{\mu}\hat{\mathbf{a}}\, d\mu ds \\ &= \hat{\mathbf{v}}_{I_k}\Delta t \\ &\quad + \int_{t_k}^{t_\tau}\int_{t_k}^{s} {}^{\mu}_{k}\mathbf{C}^\top_{\hat{\bar{q}}}({}^{\mu}\mathbf{a}_m - \hat{\mathbf{b}}_a - {}^{\mu}\hat{\mathbf{g}})\, d\mu ds \\ &= \hat{\mathbf{v}}_{I_k}\Delta t - \frac{1}{2}{}^{R_k}\hat{\mathbf{g}}\Delta t^2 \\ &\quad + \underbrace{\int_{t_k}^{t_\tau}\int_{t_k}^{s} {}^{\mu}_{k}\mathbf{C}^\top_{\hat{\bar{q}}}({}^{\mu}\mathbf{a}_m - \hat{\mathbf{b}}_a)\, d\mu ds}_{\Delta\mathbf{p}_{k,\tau}} \end{aligned} \tag{13}$$

$$\begin{aligned} {}^{R_k}\hat{\mathbf{v}}_{I_\tau} &= \hat{\mathbf{v}}_{I_k} + \int_{t_k}^{t_\tau} {}^{s}_{k}\mathbf{C}^\top_{\hat{\bar{q}}}\,{}^{s}\hat{\mathbf{a}}\, ds \\ &= \hat{\mathbf{v}}_{I_k} + \int_{t_k}^{t_\tau} {}^{s}_{k}\mathbf{C}^\top_{\hat{\bar{q}}}({}^{s}\mathbf{a}_m - \hat{\mathbf{b}}_a - {}^{s}\hat{\mathbf{g}})\, ds \\ &= \hat{\mathbf{v}}_{I_k} - {}^{R_k}\hat{\mathbf{g}}\Delta t \\ &\quad + \underbrace{\int_{t_k}^{t_\tau} {}^{s}_{k}\mathbf{C}^\top_{\hat{\bar{q}}}({}^{s}\mathbf{a}_m - \hat{\mathbf{b}}_a)\, ds}_{\Delta\mathbf{v}_{k,\tau}} \end{aligned} \tag{14}$$

where $\Delta t = t_\tau - t_k$; in particular, the preintegrated inertial terms, $\Delta\mathbf{p}_{k,\tau}$ and $\Delta\mathbf{v}_{k,\tau}$, can be recursively computed with all the incoming IMU measurements in $[t_k, t_\tau]$ (Eckenhoff et al., 2016); therefore, the estimate of velocity in the current IMU frame is obtained as

$$\hat{\mathbf{v}}_{I_\tau} = {}^{\tau}_{k}\mathbf{C}_{\hat{\bar{q}}}^{R_k}\hat{\mathbf{v}}_{I_\tau} \tag{15}$$

(iii) bias estimates are assumed constant over $[t_k, t_\tau]$; that is, $\hat{\mathbf{b}}_g = \hat{\mathbf{b}}_{g_k}$ and $\hat{\mathbf{b}}_a = \hat{\mathbf{b}}_{a_k}$ for both (i) and (ii).

Then, for covariance propagation, the discrete-time error-state transition matrix $\mathbf{\Phi}(t_\tau, t_{\tau-1})$ can be obtained using the forward Euler method over the time interval $[t_{\tau-1}, t_\tau]$:

$$\mathbf{\Phi}(t_\tau, t_{\tau-1}) = \exp(\mathbf{F}\delta t) \simeq \mathbf{I}_{24} + \mathbf{F}\delta t =: \mathbf{\Phi}_{\tau,\tau-1} \tag{16}$$

where $\delta t = t_\tau - t_{\tau-1}$. It results in the covariance propagation from time-step $\tau - 1$ to τ:

$$\mathbf{P}_{\tau|k} = \mathbf{\Phi}_{\tau,\tau-1}\mathbf{P}_{\tau-1|k}\mathbf{\Phi}_{\tau,\tau-1}^\top + \mathbf{G}\mathbf{Q}\mathbf{G}^\top \tag{17}$$

noting that $\mathbf{Q} = \mathbf{Diag}\left[\delta t\sigma_g^2\mathbf{I}_3 \;\; \delta t\sigma_{wg}^2\mathbf{I}_3 \;\; \delta t\sigma_a^2\mathbf{I}_3 \;\; \delta t\sigma_{wa}^2\mathbf{I}_3\right]$ is the discrete-time input noise covariance matrix, while the detailed derivations can be found in our companion technical report (Huai and Huang, 2018).

For the augmented state, $\hat{\mathbf{x}}_k$, we consider that the relative pose states in the sliding window are static so that $\hat{\mathbf{w}}_\tau = \hat{\mathbf{w}}_k$. The corresponding augmented covariance matrix, $\mathbf{P}_k$, can be partitioned according to the robocentric state and the sliding-window state (see (4)), as

$$\mathbf{P}_k = \begin{bmatrix} \mathbf{P}_{\mathbf{x}\mathbf{x}_k} & \mathbf{P}_{\mathbf{x}\mathbf{w}_k} \\ \mathbf{P}_{\mathbf{x}\mathbf{w}_k}^\top & \mathbf{P}_{\mathbf{w}\mathbf{w}_k} \end{bmatrix} \tag{18}$$

Thus, the augmented covariance matrix at time t_τ is given by

$$\mathbf{P}_{\tau|k} = \begin{bmatrix} \mathbf{P}_{\mathbf{x}\mathbf{x}_{\tau|k}} & \mathbf{\Phi}_{\tau,k}\mathbf{P}_{\mathbf{x}\mathbf{w}_k} \\ \mathbf{P}_{\mathbf{x}\mathbf{w}_k}^\top\mathbf{\Phi}_{\tau,k}^\top & \mathbf{P}_{\mathbf{w}\mathbf{w}_k} \end{bmatrix} \tag{19}$$

where $\mathbf{P}_{\mathbf{x}\mathbf{x}_{\tau|k}}$ can be recursively computed using (17) while the compound error-state transition matrix is obtained as

$$\mathbf{\Phi}_{\tau,k} = \mathbf{\Phi}_{\tau,\tau-1}\prod_{t_s\in[t_k,\,t_{\tau-1}]}\mathbf{\Phi}_{s,s-1} \tag{20}$$

with initial condition $\mathbf{\Phi}_{k,k} = \mathbf{I}_{24}$.

3.3. Update

3.3.1. Inverse-depth measurement model. We adopt the inverse depth parameterization (Civera et al., 2008) for the landmarks observed by a monocular camera, while being *tailored* for the proposed R-VIO. Assuming that a single landmark, L_j, has been observed from a set of n_j robocentric frames, $\mathcal{R}_j$, the measurements of L_j expressed in the corresponding n_j camera frames, $\mathcal{C}_j$, are given by the following perspective projection model in terms of *xyz* coordinates ($i \in \mathcal{C}_j$):

$$\mathbf{z}_{j,i} = \frac{1}{z_i}\begin{bmatrix} x_i \\ y_i \end{bmatrix} + \mathbf{n}_{j,i}, \quad {}^{C_i}\mathbf{p}_{L_j} = [\,x_i \quad y_i \quad z_i\,]^\top \tag{21}$$

where $\mathbf{n}_{j,i} \sim \mathcal{N}(\mathbf{0}, \sigma_{im}^2\mathbf{I}_2)$ is the additive image noise, and ${}^{C_i}\mathbf{p}_{L_j}$ denotes the position of L_j in the camera frame $\{C_i\}$.

Considering the relative poses between the camera frames in $\mathcal{C}_j$, ${}^{C_i}\mathbf{p}_{L_j}$ can also be expressed as

$${}^{C_i}\mathbf{p}_{L_j} = {}^{i}_{1}\bar{\mathbf{C}}_{\bar{q}}\,{}^{C_1}\mathbf{p}_{L_j} + {}^{i}\bar{\mathbf{p}}_1 \tag{22}$$

in terms of the following *inverse depth* parameters:

$${}^{C_1}\mathbf{p}_{L_j} = \frac{1}{\rho}\mathbf{e}(\phi,\psi), \quad \mathbf{e} = \begin{bmatrix} \cos\phi\sin\psi \\ \sin\phi \\ \cos\phi\cos\psi \end{bmatrix} \tag{23}$$

where ${}^{C_1}\mathbf{p}_{L_j}$ is the position of L_j in the first camera frame of $\mathcal{C}_j$ formed by $\mathbf{e}$ (i.e., the directional vector of ${}^{C_1}\mathbf{p}_{L_j}$ with ϕ and ψ the elevation and azimuth expressed in $\{C_1\}$) and ρ the inverse depth along $\mathbf{e}$. In particular, the relative poses between $\{C_1\}$ and $\{C_i\}$, $i = 2, \ldots, n_j$, are expressed using the camera-to-IMU calibration parameters, $\{{}^{C}_{I}\bar{q}, {}^{C}\mathbf{p}_I\}$, as

$${}^{i}_{1}\bar{\mathbf{C}}_{\bar{q}} = {}^{C}_{I}\mathbf{C}_{\bar{q}1}\,{}^{i}\mathbf{C}_{\bar{q}}\,{}^{I}_{C}\mathbf{C}_{\bar{q}} \tag{24}$$

$${}^{i}\bar{\mathbf{p}}_1 = {}^{C}_{I}\mathbf{C}_{\bar{q}1}\,{}^{i}\mathbf{C}_{\bar{q}}\,{}^{I}\mathbf{p}_C + {}^{C}_{I}\mathbf{C}_{\bar{q}}^{R_i}\mathbf{p}_{R_1} + {}^{C}\mathbf{p}_I \tag{25}$$

where we have used the following identities constructed with the sliding-window state, $\mathbf{w}$:

$$\begin{aligned} {}^{i}_{1}\mathbf{C}_{\bar{q}} &= {}^{i}_{i-1}\mathbf{C}_{\bar{q}}\,{}^{i-1}_{i-2}\mathbf{C}_{\bar{q}} \;\; \cdots \;\; {}^{2}_{1}\mathbf{C}_{\bar{q}} \\ &= \prod_{n=2}^{i}{}^{n}_{n-1}\mathbf{C}_{\bar{q}} \end{aligned} \tag{26}$$

$$\begin{aligned} {}^{R_i}\mathbf{p}_{R_1} &= -\left({}^{i}_{i-1}\mathbf{C}_{\bar{q}}^{R_{i-1}}\mathbf{p}_{R_i} + {}^{i}_{i-2}\mathbf{C}_{\bar{q}}^{R_{i-2}}\mathbf{p}_{R_{i-1}} + \ldots + {}^{i}_{1}\mathbf{C}_{\bar{q}}^{R_1}\mathbf{p}_{R_2}\right) \\ &= -\sum_{n=2}^{i}{}^{i}_{1}\mathbf{C}_{\bar{q}1}^{n-1}\mathbf{C}_{\bar{q}}^{\top R_{n-1}}\mathbf{p}_{R_n} \end{aligned} \tag{27}$$

Interestingly, if the landmark is in the distance (i.e., $\rho \to 0$), we can normalize (22) by premultiplying ρ to both sides to avoid the potential numerical issues, as

$$\begin{aligned} \rho\,{}^{C_i}\mathbf{p}_{L_j} &= {}^{i}_{1}\bar{\mathbf{C}}_{\bar{q}}\mathbf{e}(\phi,\psi) + \rho\,{}^{i}\bar{\mathbf{p}}_1 \\ &=: \mathbf{h}_i(\mathbf{w},\phi,\psi,\rho) = \begin{bmatrix} h_{i,1}(\mathbf{w},\phi,\psi,\rho) \\ h_{i,2}(\mathbf{w},\phi,\psi,\rho) \\ h_{i,3}(\mathbf{w},\phi,\psi,\rho) \end{bmatrix} \end{aligned} \tag{28}$$

Note that, this equation reserves the perspective geometry of (22) while encompassing two degenerate cases: (i) observing the landmarks at infinity (i.e., $\rho \to 0$), and (ii) having small parallax between the camera poses (i.e.,

$|{}^i\bar{\mathbf{p}}_1| \to 0$). For both cases, (28) can be approximated by $\mathbf{h}_i \simeq {}^i_1\bar{\mathbf{C}}_{\bar{q}}\, \mathbf{e}(\phi, \psi)$, and hence the corresponding measurements can still provide the constraints about the camera's orientation.

Therefore, we introduce the following inverse depth-based measurement model for the proposed R-VIO:

$$\mathbf{z}_{j,i} = \frac{1}{h_{i,3}} \begin{bmatrix} h_{i,1} \\ h_{i,2} \end{bmatrix} + \mathbf{n}_{j,i} \tag{29}$$

Denoting $\boldsymbol{\lambda} := [\phi, \psi, \rho]^\top$ and linearizing (29) at the current state estimate, $\hat{\mathbf{x}}$, and $\hat{\boldsymbol{\lambda}}$, we have the linearized measurement residual equation:

$$\mathbf{r}_{j,i} = \mathbf{z}_{j,i} - \hat{\mathbf{z}}_{j,i} \simeq \mathbf{H}_{\mathbf{x}_{j,i}} \tilde{\mathbf{x}} + \mathbf{H}_{\boldsymbol{\lambda}_{j,i}} \tilde{\boldsymbol{\lambda}} + \mathbf{n}_{j,i} \tag{30}$$

where the measurement Jacobians are given by

$$\begin{aligned}
&\mathbf{H}_{\mathbf{x}_{j,i}} = \mathbf{H}_{\mathrm{p}_{j,i}} \begin{bmatrix} \mathbf{0}_{3\times 24} & \mathbf{H}_{\mathbf{w}_{j,i}} \end{bmatrix}, \quad \mathbf{H}_{\boldsymbol{\lambda}_{j,i}} = \mathbf{H}_{\mathrm{p}_{j,i}} \mathbf{H}_{\mathrm{inv}_{j,i}} \\
&\mathbf{H}_{\mathrm{p}_{j,i}} = \frac{1}{\hat{h}_{i,3}} \begin{bmatrix} 1 & 0 & -\frac{\hat{h}_{i,1}}{\hat{h}_{i,3}} \\ 0 & 1 & -\frac{\hat{h}_{i,2}}{\hat{h}_{i,3}} \end{bmatrix}, \\
&\mathbf{H}_{\mathrm{inv}_{j,i}} = \frac{\partial \mathbf{h}_i}{\partial \tilde{\boldsymbol{\lambda}}} \\
&\quad = \begin{bmatrix} \frac{\partial \mathbf{h}_i}{\partial [\tilde{\phi}, \tilde{\psi}]^\top} & \frac{\partial \mathbf{h}_i}{\partial \tilde{\rho}} \end{bmatrix} \\
&\quad = \begin{bmatrix} {}^i_1\hat{\mathbf{C}}_{\hat{\bar{q}}} \begin{bmatrix} -\sin\hat{\phi}\sin\hat{\psi} & \cos\hat{\phi}\cos\hat{\psi} \\ \cos\hat{\phi} & 0 \\ -\sin\hat{\phi}\cos\hat{\psi} & -\cos\hat{\phi}\sin\hat{\psi} \end{bmatrix} & {}^i\hat{\bar{\mathbf{p}}}_1 \end{bmatrix}, \\
&\mathbf{H}_{\mathbf{w}_{j,i}} = \frac{\partial \mathbf{h}_i}{\partial \tilde{\mathbf{w}}} \\
&\quad = \begin{bmatrix} \frac{\partial \mathbf{h}_i}{\partial \delta\boldsymbol{\theta}_2} & \frac{\partial \mathbf{h}_i}{\partial {}^{R_1}\tilde{\mathbf{p}}_{R_2}} & \cdots & \cdots & \frac{\partial \mathbf{h}_i}{\partial \delta\boldsymbol{\theta}_N} & \frac{\partial \mathbf{h}_i}{\partial {}^{R_{N-1}}\tilde{\mathbf{p}}_{R_N}} \end{bmatrix}, \\
&\frac{\partial \mathbf{h}_i}{\partial \delta\boldsymbol{\theta}_n} = {}^C_I\mathbf{C}_{\bar{q}_1}\, {}^i_1\mathbf{C}_{\hat{\bar{q}}} \left\lfloor ({}^I_C\mathbf{C}_{\bar{q}}\hat{\mathbf{e}} + \hat{\rho}\, {}^I\mathbf{p}_C - \hat{\rho}\, {}^{R_1}\hat{\mathbf{p}}_{R_n}) \times \right\rfloor {}^n_1\mathbf{C}^\top_{\hat{\bar{q}}}, \\
&\frac{\partial \mathbf{h}_i}{\partial {}^{R_{n-1}}\tilde{\mathbf{p}}_{R_n}} = -\hat{\rho}\, {}^C_I\mathbf{C}_{\bar{q}_1}\, {}^i_1\mathbf{C}_{\hat{\bar{q}}_1}\, {}^{n-1}_1\mathbf{C}^\top_{\hat{\bar{q}}}, \quad n = 2, \ldots, i \leqslant N
\end{aligned} \tag{31}$$

Specifically, $\mathbf{H}_{\mathbf{x}_{j,i}}$ and $\mathbf{H}_{\boldsymbol{\lambda}_{j,i}}$ are the Jacobians with respect to the vectors of state and inverse depth, respectively. Note that through the Jacobian $\mathbf{H}_{\mathbf{w}_{j,i}}$ each measurement of L_j is correlated to a sequence of relative poses in $\mathbf{w}$, building up a *dense* connection between the measurement and the state, however, without increasing the computational complexity. This is also different from Mourikis and Roumeliotis (2007) where each measurement is only correlated to the global pose from which it is observed. Since an estimate of $\boldsymbol{\lambda}$ is needed for computing $\mathbf{r}_{j,i}$ and $\mathbf{H}_{\boldsymbol{\lambda}_{j,i}}$, first a local BA is solved for $\hat{\boldsymbol{\lambda}}$ with the measurements, $\{\mathbf{z}_{j,i},\ i \in \mathcal{C}_j\}$, and the relative pose estimates, $\hat{\mathbf{w}}$ (see Appendix B). After stacking the residuals (see (30)) for all $i \in \mathcal{C}_j$, we obtain

$$\mathbf{r}_j \simeq \mathbf{H}_{\mathbf{x}_j} \tilde{\mathbf{x}} + \mathbf{H}_{\boldsymbol{\lambda}_j} \tilde{\boldsymbol{\lambda}} + \mathbf{n}_j \tag{32}$$

Assuming the measurements obtained from different camera poses are independent, the covariance matrix of $\mathbf{n}_j$ is hence $\mathbf{R}_j = \sigma^2_{im}\mathbf{I}_{2n_j}$. As $\hat{\mathbf{x}}$ (precisely, $\hat{\mathbf{w}}$) is used to compute $\hat{\boldsymbol{\lambda}}$, the error of inverse depth is correlated with $\tilde{\mathbf{x}}$. In order to be a valid residual for EKF update, (32) is projected to the left nullspace of $\mathbf{H}_{\boldsymbol{\lambda}_j}$ (i.e., $\mathbf{O}^\top_{\boldsymbol{\lambda}_j}\mathbf{H}_{\boldsymbol{\lambda}_j} = \mathbf{0}$, with $\mathbf{O}^\top_{\boldsymbol{\lambda}_j}\mathbf{O}_{\boldsymbol{\lambda}_j} = \mathbf{I}$):

$$\bar{\mathbf{r}}_j = \mathbf{O}^\top_{\boldsymbol{\lambda}_j}\mathbf{r}_j \simeq \mathbf{O}^\top_{\boldsymbol{\lambda}_j}\mathbf{H}_{\mathbf{x}_j}\tilde{\mathbf{x}} + \mathbf{O}^\top_{\boldsymbol{\lambda}_j}\mathbf{n}_j = \bar{\mathbf{H}}_{\mathbf{x}_j}\tilde{\mathbf{x}} + \bar{\mathbf{n}}_j \tag{33}$$

In general, $\mathbf{H}_{\boldsymbol{\lambda}_j}$ is a $2n_j \times 3$ matrix of full column rank. The nullspace projection of dimension $2n_j - 3$ can be efficiently computed, for example, using the *Givens rotations* (Golub and Van Loan, 2012), with $O(n_j^2)$ complexity. Since $\mathbf{O}_{\boldsymbol{\lambda}_j}$ is unitary, the covariance matrix of $\bar{\mathbf{n}}_j$ becomes

$$\bar{\mathbf{R}}_j = \mathbf{O}^\top_{\boldsymbol{\lambda}_j}\mathbf{R}_j\mathbf{O}_{\boldsymbol{\lambda}_j} = \sigma^2_{im}\mathbf{I}_{2n_j-3} \tag{34}$$

At this point, let us examine the special cases where $\mathbf{H}_{\boldsymbol{\lambda}_{j,i}}$ (more precisely, $\mathbf{H}_{p_{j,i}}$ or $\mathbf{H}_{\mathrm{inv}_{j,i}}$) may become rank deficient (see (31)), which would affect computing the residual (33). First, if $\mathbf{H}_{p_{j,i}}$ becomes rank deficient, then we can find two possible causes about $\hat{\mathbf{h}}_i$: (i) $\hat{h}_{i,1} \approx \hat{h}_{i,2} \approx \hat{h}_{i,3}$, which implies that the image size should be at least $2f \times 2f$ with f the camera focal length, or (ii) $\hat{h}_{i,1} \to 0$ and $\hat{h}_{i,2} \to 0$, which means that the measurement of L_j is close to the principal point of camera image. Second, if $\mathbf{H}_{\mathrm{inv}_{j,i}}$ is rank deficient, we can also find two possible causes: (iii) $\cos\hat{\phi} \to 0$, which implies that we have either infinitely small focal length or infinitely large image size of the camera so that $|\hat{\phi}| \to \pi/2$ can happen, or (iv) $|{}^i\hat{\bar{\mathbf{p}}}_1| \to 0$, which means a small parallax between $\{C_1\}$ and $\{C_i\}$. Among these causes, (i) is about the selection of the lens, which must be restricted by the size of camera image, and (iii) is too ideal to be realized in the real world; while (ii) and (iv) are common in the vision-based navigation which can be effectively detected by checking the values of pixel measurements and relative pose estimates, respectively. Then, we might reject those measurements that meet (ii) when computing the Jacobians. However, in the case of (iv) (e.g., purely rotating or motionless) since the last column of $\mathbf{H}_{\boldsymbol{\lambda}_{j,i}}$ (and, hence, of $\mathbf{H}_{\boldsymbol{\lambda}_j}$) approaches zero, we perform the Givens rotations with only the first two columns of $\mathbf{H}_{\boldsymbol{\lambda}_j}$ to numerically guarantee a valid nullspace projection (see (33)) and thus the dimension of $\bar{\mathbf{r}}_j$ is increased by 1 (see (34)).

In addition, the *Mahalanobis distance* is checked for each landmark using all its measurements before the EKF update, serving as the probabilistic outlier rejection:

$$\mathcal{D}_j = \bar{\mathbf{r}}_j^\top (\bar{\mathbf{H}}_{\mathbf{x}_j}\mathbf{P}\bar{\mathbf{H}}_{\mathbf{x}_j}^\top + \bar{\mathbf{R}}_j)^{-1}\bar{\mathbf{r}}_j \leqslant \chi^2_{r,1-\alpha} \tag{35}$$

where $\chi^2_{r,1-\alpha}$ is a threshold obtained from the χ^2 distribution with $r = \dim(\bar{\mathbf{r}}_j)$ and α the significance level (e.g., 0.05). If (35) holds, then landmark L_j can be accepted as an inlier and used for EKF update. It should be noted that, in addition to the outlier rejection based on vision alone,

the state estimate, $\{\hat{\mathbf{x}}, \mathbf{P}\}$, is used as *a prior* to help identify the outliers. Thus, the measurements that render detrimental residuals given the current state estimate (e.g., may correspond to the features of objects that move slowly or in parallel with the camera in dynamic environments) can be more accurately detected and discarded to further ensure the health of the estimator.

3.3.2. EKF update. Assuming that at time-step $k+1$ we have the measurements of M landmarks (inliers) to process, thus we can stack the resulting $\bar{\mathbf{r}}_j$, $j = 1, \ldots, M$, to have

$$\bar{\mathbf{r}} = \bar{\mathbf{H}}_{\mathbf{x}}\tilde{\mathbf{x}} + \bar{\mathbf{n}} \tag{36}$$

of which the dimension $d = \sum_{j=1}^{M}(2n_j - 3)$. However, in practice, d could be a large number even if M is small (e.g., $d = 170$, for 10 landmarks observed from 10 camera poses). To reduce the computational complexity, QR decomposition is applied to (36) to compress the dimension of measurement residual. Note that $\bar{\mathbf{H}}_{\mathbf{x}}$ is *rank deficient* by the zero columns corresponding to the robocentric states, whereas the non-zero columns corresponding to the relative pose states within the sliding window are linearly independent. Therefore, to save the computational cost the QR decomposition can be applied only to the non-zero part of $\bar{\mathbf{H}}_{\mathbf{x}}$, as

$$\begin{aligned}\bar{\mathbf{H}}_{\mathbf{x}} &= \begin{bmatrix}\mathbf{0}_{d\times 24} & \bar{\mathbf{H}}_{\mathbf{w}}\end{bmatrix} \\ &= \begin{bmatrix}\mathbf{0}_{d\times 24} & \begin{bmatrix}\mathbf{Q}_1 & \mathbf{Q}_2\end{bmatrix}\begin{bmatrix}\bar{\mathbf{T}}_{\mathbf{w}} \\ \mathbf{0}_{(d-6(N-1))\times 6(N-1)}\end{bmatrix}\end{bmatrix} \\ &= \begin{bmatrix}\mathbf{Q}_1 & \mathbf{Q}_2\end{bmatrix}\begin{bmatrix}\mathbf{0}_{d\times 24} & \begin{bmatrix}\bar{\mathbf{T}}_{\mathbf{w}} \\ \mathbf{0}_{(d-6(N-1))\times 6(N-1)}\end{bmatrix}\end{bmatrix}\end{aligned}$$

where $\mathbf{Q}_1$ and $\mathbf{Q}_2$ are the unitary matrices of dimension $d \times 6(N-1)$ and $d \times (d - 6(N-1))$, respectively, and $\bar{\mathbf{T}}_{\mathbf{w}}$ is an upper triangular matrix of dimension $6(N-1)$. With this definition, (36) yields

$$\begin{aligned}&\bar{\mathbf{r}} = \begin{bmatrix}\mathbf{Q}_1 & \mathbf{Q}_2\end{bmatrix}\begin{bmatrix}\mathbf{0} & \bar{\mathbf{T}}_{\mathbf{w}} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}\tilde{\mathbf{x}} + \bar{\mathbf{n}} \Rightarrow \\ &\begin{bmatrix}\mathbf{Q}_1^\top \\ \mathbf{Q}_2^\top\end{bmatrix}\bar{\mathbf{r}} = \begin{bmatrix}\mathbf{0} & \bar{\mathbf{T}}_{\mathbf{w}} \\ \mathbf{0} & \mathbf{0}\end{bmatrix}\tilde{\mathbf{x}} + \begin{bmatrix}\mathbf{Q}_1^\top \\ \mathbf{Q}_2^\top\end{bmatrix}\bar{\mathbf{n}}\end{aligned} \tag{37}$$

for which we can discard the lower $d - 6(N-1)$ rows that are only about the measurement noise, but employ the upper $6(N-1)$ rows, instead of (36), as the compressed residual:

$$\breve{\mathbf{r}} = \mathbf{Q}_1^\top\bar{\mathbf{r}} = \begin{bmatrix}\mathbf{0} & \bar{\mathbf{T}}_{\mathbf{w}}\end{bmatrix}\tilde{\mathbf{x}} + \mathbf{Q}_1^\top\bar{\mathbf{n}} = \breve{\mathbf{H}}_{\mathbf{x}}\tilde{\mathbf{x}} + \breve{\mathbf{n}} \tag{38}$$

where $\breve{\mathbf{n}} = \mathbf{Q}_1^\top\bar{\mathbf{n}}$ is the noise vector with covariance matrix $\breve{\mathbf{R}} = \mathbf{Q}_1^\top\bar{\mathbf{R}}\mathbf{Q}_1 = \sigma_{im}^2\mathbf{I}_{6(N-1)}$. In particular, when we have $d \gg 6(N-1)$ this can be done using the Givens rotations, with $O(N^2 d)$ complexity. Based on the above, the standard EKF update is performed according to (Maybeck, 1979)

$$\begin{aligned}\mathbf{K} &= \mathbf{P}\breve{\mathbf{H}}_{\mathbf{x}}^\top(\breve{\mathbf{H}}_{\mathbf{x}}\mathbf{P}\breve{\mathbf{H}}_{\mathbf{x}}^\top + \breve{\mathbf{R}})^{-1} \\ \hat{\mathbf{x}}_{k+1|k+1} &= \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}\breve{\mathbf{r}} \\ \mathbf{P}_{k+1|k+1} &= (\mathbf{I} - \mathbf{K}\breve{\mathbf{H}}_{\mathbf{x}})\mathbf{P}_{k+1|k}(\mathbf{I} - \mathbf{K}\breve{\mathbf{H}}_{\mathbf{x}})^\top + \mathbf{K}\breve{\mathbf{R}}\mathbf{K}^\top\end{aligned}$$

3.3.3. State augmentation. To utilize the most accurate, relative motion information for estimation, we employ stochastic state cloning (Roumeliotis and Burdick, 2002). In particular, the state augmentation is carried out once an EKF update is done, for which a copy of the current relative pose estimate, $\{{}^{k+1}_{k}\hat{\bar{q}}_{k+1|k+1}, {}^{R_k}\hat{\mathbf{p}}_{I_{k+1|k+1}}\}$, is appended to the end of the current sliding-window vector, $\hat{\mathbf{w}}_{k+1|k+1}$. Accordingly, the covariance matrix is augmented as follows:

$$\begin{aligned}&\mathbf{P}_{k+1|k+1} \leftarrow \begin{bmatrix}\mathbf{I}_{24+6(N-1)} \\ \mathbf{J}\end{bmatrix}\mathbf{P}_{k+1|k+1}\begin{bmatrix}\mathbf{I}_{24+6(N-1)} \\ \mathbf{J}\end{bmatrix}^\top, \\ &\mathbf{J} = \begin{bmatrix}\mathbf{0}_{3\times 9} & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_{3\times 9} & \mathbf{0}_{3\times 6(N-1)} \\ \mathbf{0}_{3\times 9} & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_{3\times 9} & \mathbf{0}_{3\times 6(N-1)}\end{bmatrix}\end{aligned} \tag{39}$$

3.4. Composition

Note that in the proposed robocentric formulation every time when the update is finished, we shift the frame of reference of the VINS. At this point, the current IMU frame $\{I_{k+1}\}$ is set as the new frame of reference, $\{R_{k+1}\}$, to replace $\{R_k\}$. The state vector expressed in $\{R_{k+1}\}$ is then obtained as

$$\begin{aligned}&\hat{\mathbf{x}}_{k+1} = \begin{bmatrix}{}^{R_{k+1}}\hat{\mathbf{x}}_{k+1} \\ \hat{\mathbf{w}}_{k+1}\end{bmatrix} = \begin{bmatrix}{}^{R_k}\hat{\mathbf{x}}_{k+1} \boxplus {}^{R_k}\hat{\mathbf{x}}_{I_{k+1}} \\ \hat{\mathbf{w}}_{k+1|k+1}\end{bmatrix} \Rightarrow \\ &\begin{bmatrix}{}^{k+1}_{G}\hat{\bar{q}} \\ {}^{R_{k+1}}\hat{\mathbf{p}}_G \\ {}^{R_{k+1}}\hat{\mathbf{g}} \\ {}^{k+1}_{k+1}\hat{\bar{q}} \\ {}^{R_{k+1}}\hat{\mathbf{p}}_{R_{k+1}} \\ \hat{\mathbf{v}}_{R_{k+1}} \\ \hat{\mathbf{b}}_{g_{k+1}} \\ \hat{\mathbf{b}}_{a_{k+1}} \\ \hat{\mathbf{w}}_{k+1}\end{bmatrix} = \begin{bmatrix}{}^{k+1}_{k}\hat{\bar{q}} \otimes {}^{k}_{G}\hat{\bar{q}} \\ {}^{k+1}_{k}\mathbf{C}_{\hat{\bar{q}}}({}^{R_k}\hat{\mathbf{p}}_G - {}^{R_k}\hat{\mathbf{p}}_{I_{k+1}}) \\ {}^{k+1}_{k}\mathbf{C}_{\hat{\bar{q}}}{}^{R_k}\hat{\mathbf{g}} \\ \bar{q}_0 \\ \mathbf{0}_{3\times 1} \\ \hat{\mathbf{v}}_{I_{k+1}} \\ \hat{\mathbf{b}}_{g_{k+1}} \\ \hat{\mathbf{b}}_{a_{k+1}} \\ \hat{\mathbf{w}}_{k+1|k+1}\end{bmatrix}\end{aligned} \tag{40}$$

where $\bar{q}_0 = [0, 0, 0, 1]^\top$ and $\boxplus$ denotes the state composition operator. For brevity of presentation the full subscripts of the robocentric states have been omitted. Note that in the IMU state the relative pose is *reset* to the origin of $\{R_{k+1}\}$, while the velocity and biases evolving in the local IMU frame are not affected by the change of (local) frame of reference. The covariance composition is correspondingly performed using the transformation:

$$\mathbf{P}_{k+1} = \mathbf{U}_{k+1}\mathbf{P}_{k+1|k+1}\mathbf{U}_{k+1}^\top \tag{41}$$

Algorithm 1 Robocentric Visual–Inertial Odometry

Input: Camera images and IMU measurements
Output: 6-DOF real-time pose estimates
R-VIO: Initialize the state and covariance with respect to the first local frame of reference, $\{R_0\}$ (e.g., $\{I_0\}$), when first available IMU measurement(s) comes in. After that, every time when a camera image is available, do

- **Visual tracking**: extract features from the image, and perform Kanade–Lucas–Tomasi (KLT) tracking with outlier rejection; record the tracking history of the inliers within the current sliding window.
- **Propagation**: propagate state and covariance matrix using preintegration with all the IMU measurements starting from last image time.
- **Update**: for the features whose tracks are complete (i.e., losing track, or reaching the maximum tracking length) compute the inverse-depth measurement model matrices, then do
  - *EKF update*: use the features that have passed the Mahalanobis distance test for EKF update.
  - *State augmentation*: augment state and covariance matrix with the (updated) current relative pose (state and covariance) estimate.
- **Composition**: shift local frame of reference to the current IMU frame, and update global state and covariance matrix using (updated) current relative pose estimate; reset relative pose estimate of IMU state for next image.

$$\mathbf{U}_{k+1} = \frac{\partial \tilde{\mathbf{x}}_{k+1}}{\partial \tilde{\mathbf{x}}_{k+1|k+1}} = \begin{bmatrix} \mathbf{V}_{k+1} & \mathbf{0}_{24\times 6N} \\ \mathbf{0}_{6N\times 24} & \mathbf{I}_{6N} \end{bmatrix} \tag{42}$$

where $\mathbf{V}_{k+1}$ is the Jacobian with respect to the robocentric state in $\{R_k\}$:

$$\mathbf{V}_{k+1} = \frac{\partial^{R_{k+1}} \tilde{\mathbf{x}}_{k+1}}{\partial^{R_k} \tilde{\mathbf{x}}_{k+1|k+1}} = \begin{bmatrix} {}^{k+1}_{k}\mathbf{C}_{\tilde{q}} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & {}^{k+1}_{k}\mathbf{C}_{\hat{q}} & \mathbf{0}_3 & \lfloor {}^{R_{k+1}}\hat{\mathbf{p}}_G \times \rfloor & -{}^{k+1}_{k}\mathbf{C}_{\hat{q}} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & {}^{k+1}_{k}\mathbf{C}_{\hat{q}} & \lfloor {}^{R_{k+1}}\hat{\mathbf{g}} \times \rfloor & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{43}$$

The corresponding covariance of the relative pose in the IMU state is also *reset* to zero; that is, no uncertainty for the robocentric frame of reference itself.

3.5. Initialization

It is important to point out that with the proposed robocentric formulation, the filter initialization becomes simple, because the states are uniformly relative to a local frame of reference and typically start from zero, *without* the need to align the initial pose with a fixed global frame. In particular, in our implementation: (i) the initial global pose and IMU relative pose are both set to $\{\bar{q}_0, \mathbf{0}_{3\times 1}\}$, (ii) the initial local gravity is the average of first available accelerometer measurement(s), $\underline{\mathbf{a}}_m$, before the IMU moves, and (iii) the initial accelerometer bias is obtained by removing standard gravity effects from $\underline{\mathbf{a}}_m$, while the initial gyroscope bias uses the average of the corresponding stationary measurement(s), $\underline{\boldsymbol{\omega}}_m$. In addition, the initial covariance subblock for global pose is set to $10^{-6}\mathbf{I}_6$, while for the local gravity and the biases, the corresponding subblocks are set to $\mathbf{P}_0^g = \Delta T \sigma_a^2 \mathbf{I}_3$, $\mathbf{P}_0^{b_g} = \Delta T \sigma_{wg}^2 \mathbf{I}_3$, and $\mathbf{P}_0^{b_a} = \Delta T \sigma_{wa}^2 \mathbf{I}_3$, respectively, where ΔT denotes the time length of the initialization. In summary, the main steps of the proposed R-VIO are outlined in Algorithm 1.

4. Observability analysis

Observability of the system reveals whether the information provided by the sensor measurements is sufficient to estimate the states without ambiguities. In this section, we examine the observability properties of the proposed R-VIO's system model with the case of a single landmark being observed by a mobile sensing platform which performs arbitrary motion, while the conclusion of analysis can easily be generalized to the case of multiple landmarks. Note that a direct analysis of the observability properties of R-VIO could be cumbersome mainly due to the feature marginalization (see (33)) and state cloning, thus we perform observability analysis by using an EKF-SLAM model having the same observability properties as an EKF-VIO model provided the same linearization points used, which has been shown to be a common practice (see Guo and Roumeliotis, 2013; Li and Mourikis, 2013a; Hesch et al., 2014a,b) in the VINS literature.

To this end, the state vector at time-step k including the position of a single landmark L:

$$\mathbf{x}_k = \begin{bmatrix} {}^{R_k}\mathbf{x}_k^\top & {}^{R_k}\mathbf{p}_L^\top \end{bmatrix}^\top \tag{44}$$

and the measurement model (21) (or its inverse-depth form (29)) are employed in what follows. The observability matrix is computed as (Chen et al., 1990)

$$\mathbf{M} = \begin{bmatrix} \mathbf{H}_k \\ \vdots \\ \mathbf{H}_\ell \boldsymbol{\Psi}_{\ell,k} \\ \vdots \\ \mathbf{H}_{k+m} \boldsymbol{\Psi}_{k+m,k} \end{bmatrix} \tag{45}$$

where $\mathbf{\Psi}_{\ell,k}$ is the error-state transition matrix from time-step k to ℓ, and $\mathbf{H}_\ell$ is the measurement Jacobian corresponding to the observation(s) at time-step ℓ. Each block row is evaluated at ${}^{R_k}\hat{\mathbf{p}}_L$ and ${}^{R_k}\hat{\mathbf{x}}_i$, $i=k,\ \ldots,\ell,\ \ldots,k+m$. The nullspace of $\mathbf{M}$ consists of the directions of the state space in which no information is provided by the measurements; that is, the unobservable subspace. It should be noted that since the proposed robocentric EKF includes three steps, propagation, update and composition, and the local frame of reference is changed by the composition step, we study the observability in a complete cycle of: (i) propagation and update, and then (ii) composition. We analytically prove that the proposed R-VIO's system model has an invariant unobservable subspace and does *not* undergo the observability mismatch issue that has been identified to be the main cause of inconsistency (Huang et al., 2010; Li and Mourikis, 2013a; Hesch et al., 2014a,b), thus being able to improve the resulting VINS performance.

4.0.1. Error-state transition matrix. To ensure theoretical analysis, the *analytic*, closed-form error-state transition matrix is computed as follows:

$$\mathbf{\Psi}(\ell,k)=\begin{bmatrix}\mathbf{\Phi}(\ell,k) & \mathbf{0}_{24\times 3}\\ \mathbf{0}_{3\times 24} & \mathbf{I}_3\end{bmatrix} \tag{46}$$

where, instead of (20), $\mathbf{\Phi}(\ell,k)$ is obtained by integrating the following differential equation over the time interval $[t_k,t_\ell]$:

$$\dot{\mathbf{\Phi}}(\ell,k)=\mathbf{F}\mathbf{\Phi}(\ell,k) \tag{47}$$

with initial condition $\mathbf{\Phi}(k,k)=\mathbf{I}_{24}$. While the closed-form solution is given in the following, the interested reader is referred to our companion technical report (Huai and Huang, 2018) for the detailed derivations:

$$\mathbf{\Phi}(\ell,k)=\begin{bmatrix}\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3\\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3\\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3\\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{\Phi}_{44} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{\Phi}_{47} & \mathbf{0}_3\\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{\Phi}_{53} & \mathbf{\Phi}_{54} & \mathbf{I}_3 & \mathbf{\Phi}_{56} & \mathbf{\Phi}_{57} & \mathbf{\Phi}_{58}\\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{\Phi}_{63} & \mathbf{\Phi}_{64} & \mathbf{0}_3 & \mathbf{\Phi}_{66} & \mathbf{\Phi}_{67} & \mathbf{\Phi}_{68}\\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3\\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3\end{bmatrix}$$

$$\mathbf{\Phi}_{44}(\ell,k)={}^{\ell}_{k}\mathbf{C}_{\hat{\bar{q}}} \tag{48}$$

$$\mathbf{\Phi}_{47}(\ell,k)=-{}^{\ell}_{k}\mathbf{C}_{\hat{\bar{q}}}\int_{t_k}^{t_\ell}{}^{\tau}_{k}\mathbf{C}_{\hat{\bar{q}}}^{\top}\,d\tau \tag{49}$$

$$\mathbf{\Phi}_{53}(\ell,k)=-\frac{1}{2}\mathbf{I}_3\Delta t_{k,\ell}^2 \tag{50}$$

$$\mathbf{\Phi}_{54}(\ell,k)=-\left\lfloor\left({}^{R_k}\hat{\mathbf{p}}_{I_\ell}+\frac{1}{2}{}^{R_k}\hat{\mathbf{g}}\Delta t_{k,\ell}^2\right)\times\right\rfloor \tag{51}$$

$$\mathbf{\Phi}_{56}(\ell,k)=\mathbf{I}_3\Delta t_{k,\ell} \tag{52}$$

$$\begin{aligned}\mathbf{\Phi}_{57}(\ell,k)=&\int_{t_k}^{t_\ell}\left\lfloor{}^{\tau}_{k}\mathbf{C}_{\hat{\bar{q}}}^{\top}\hat{\mathbf{v}}_{I_\tau}\times\right\rfloor\int_{t_k}^{\tau}{}^{\mu}_{k}\mathbf{C}_{\hat{\bar{q}}}^{\top}\,d\mu d\tau\\&+\left\lfloor{}^{R_k}\hat{\mathbf{g}}\times\right\rfloor\int_{t_k}^{t_\ell}\int_{t_k}^{\tau}\int_{t_k}^{\mu}{}^{\lambda}_{k}\mathbf{C}_{\hat{\bar{q}}}^{\top}\,d\lambda d\mu d\tau\\&-\int_{t_k}^{t_\ell}\int_{t_k}^{\tau}{}^{\mu}_{k}\mathbf{C}_{\hat{\bar{q}}}^{\top}\left\lfloor\hat{\mathbf{v}}_{I_\mu}\times\right\rfloor d\mu d\tau\end{aligned} \tag{53}$$

$$\mathbf{\Phi}_{58}(\ell,k)=-\int_{t_k}^{t_\ell}\int_{t_k}^{\tau}{}^{\mu}_{k}\mathbf{C}_{\hat{\bar{q}}}^{\top}\,d\mu d\tau \tag{54}$$

$$\mathbf{\Phi}_{63}(\ell,k)=-{}^{\ell}_{k}\mathbf{C}_{\hat{\bar{q}}}\Delta t_{k,\ell} \tag{55}$$

$$\mathbf{\Phi}_{64}(\ell,k)=-{}^{\ell}_{k}\mathbf{C}_{\hat{\bar{q}}}\left\lfloor{}^{R_k}\hat{\mathbf{g}}\times\right\rfloor\Delta t_{k,\ell} \tag{56}$$

$$\mathbf{\Phi}_{66}(\ell,k)=\mathbf{\Phi}_{44}(\ell,k) \tag{57}$$

$$\begin{aligned}\mathbf{\Phi}_{67}(\ell,k)=&{}^{\ell}_{k}\mathbf{C}_{\hat{\bar{q}}}\left\lfloor{}^{R_k}\hat{\mathbf{g}}\times\right\rfloor\int_{t_k}^{t_\ell}\int_{t_k}^{\tau}{}^{\mu}_{k}\mathbf{C}_{\hat{\bar{q}}}^{\top}\,d\mu d\tau\\&-\int_{t_k}^{t_\ell}{}^{\ell}_{\tau}\mathbf{C}_{\hat{\bar{q}}}\left\lfloor\hat{\mathbf{v}}_{I_\tau}\times\right\rfloor d\tau\end{aligned} \tag{58}$$

$$\mathbf{\Phi}_{68}(\ell,k)=-{}^{\ell}_{k}\mathbf{C}_{\hat{\bar{q}}}\int_{t_k}^{t_\ell}{}^{\tau}_{k}\mathbf{C}_{\hat{\bar{q}}}^{\top}\,d\tau \tag{59}$$

where $\Delta t_{k,\ell}=t_\ell-t_k$.

4.0.2. Measurement Jacobian. At time $t_\ell\in[t_k,t_{k+m}]$, the position of landmark L in $\{I_\ell\}$ can be expressed as

$${}^{I_\ell}\mathbf{p}_L={}^{\ell}_{k}\mathbf{C}_{\bar{q}}({}^{R_k}\mathbf{p}_L-{}^{R_k}\mathbf{p}_{I_\ell}) \tag{60}$$

Based on (21), the bearing-only measurement is given by

$$\mathbf{z}_\ell=\frac{1}{z}\begin{bmatrix}x\\ y\end{bmatrix},\quad {}^{I_\ell}\mathbf{p}_L=[\,x\quad y\quad z\,]^{\top} \tag{61}$$

We assume that the IMU and camera frames *coincide* for brevity of presentation, and the corresponding measurement Jacobian is thus in the form of

$$\begin{aligned}&\mathbf{H}_\ell=\mathbf{H}_{\mathrm{p}}{}^{\ell}_{k}\mathbf{C}_{\hat{\bar{q}}}[\,\mathbf{0}_3\quad\mathbf{0}_3\quad\mathbf{0}_3\quad\mathbf{H}_{\boldsymbol{\theta}_\ell}\quad-\mathbf{I}_3\quad\mathbf{0}_{3\times 9}\ \mid\ \mathbf{I}_3\,]\\&\mathbf{H}_{\mathrm{p}}=\frac{1}{\hat{z}}\begin{bmatrix}1 & 0 & -\frac{\hat{x}}{\hat{z}}\\ 0 & 1 & -\frac{\hat{y}}{\hat{z}}\end{bmatrix},\quad \mathbf{H}_{\boldsymbol{\theta}_\ell}=\left\lfloor({}^{R_k}\hat{\mathbf{p}}_L-{}^{R_k}\hat{\mathbf{p}}_{I_\ell})\times\right\rfloor{}^{\ell}_{k}\mathbf{C}_{\hat{\bar{q}}}^{\top}\end{aligned} \tag{62}$$

4.1. Observability of propagation and update

Based on the above Jacobians, we obtain the ℓth block row, $\mathbf{M}_\ell$, of $\mathbf{M}$ as follows (see (46), (48)–(59), and (62)):

$$\begin{aligned}\mathbf{M}_\ell&=\mathbf{H}_\ell\mathbf{\Psi}_{\ell,k}\\&=\mathbf{\Pi}[\,\mathbf{0}_3\quad\mathbf{0}_3\quad\mathbf{\Gamma}_1\quad\mathbf{\Gamma}_2\quad-\mathbf{I}_3\quad\mathbf{\Gamma}_3\quad\mathbf{\Gamma}_4\quad\mathbf{\Gamma}_5\ \mid\ \mathbf{I}_3\,]\end{aligned}$$

where

$$\mathbf{\Pi} = \mathbf{H}_{\mathrm{p}k}^{\ell}\mathbf{C}_{\hat{\bar{q}}} \tag{63}$$

$$\mathbf{\Gamma}_1 = -\mathbf{\Phi}_{53} = \frac{1}{2}\mathbf{I}_3\Delta t_{k,\ell}^2 \tag{64}$$

$$\begin{aligned}\mathbf{\Gamma}_2 &= \left\lfloor ({}^{R_k}\hat{\mathbf{p}}_L - {}^{R_k}\hat{\mathbf{p}}_{I_\ell})\times \right\rfloor {}_k^{\ell}\mathbf{C}_{\hat{\bar{q}}}^{\top}\mathbf{\Phi}_{44} - \mathbf{\Phi}_{54} \\ &= \left\lfloor {}^{R_k}\hat{\mathbf{p}}_L \times \right\rfloor + \frac{1}{2}\left\lfloor {}^{R_k}\hat{\mathbf{g}} \times \right\rfloor \Delta t_{k,\ell}^2\end{aligned} \tag{65}$$

$$\mathbf{\Gamma}_3 = -\mathbf{\Phi}_{56} = -\mathbf{I}_3\Delta t_{k,\ell} \tag{66}$$

$$\begin{aligned}\mathbf{\Gamma}_4 &= \left\lfloor ({}^{R_k}\hat{\mathbf{p}}_L - {}^{R_k}\hat{\mathbf{p}}_{I_\ell})\times \right\rfloor {}_k^{\ell}\mathbf{C}_{\hat{\bar{q}}}^{\top}\mathbf{\Phi}_{47} - \mathbf{\Phi}_{57} \\ &= -\left\lfloor ({}^{R_k}\hat{\mathbf{p}}_L - {}^{R_k}\hat{\mathbf{p}}_{I_\ell})\times \right\rfloor \int_{t_k}^{t_\ell} {}_k^{\tau}\mathbf{C}_{\hat{\bar{q}}}^{\top}\, d\boldsymbol{\tau} - \mathbf{\Phi}_{57}\end{aligned} \tag{67}$$

$$\mathbf{\Gamma}_5 = -\mathbf{\Phi}_{58} \tag{68}$$

Note that for generic motion (i.e., $\boldsymbol{\omega} \neq \mathbf{0}_{3\times 1}$ and $\mathbf{a} \neq \mathbf{0}_{3\times 1}$), the values of $\mathbf{\Phi}_{57}$ and $\mathbf{\Phi}_{58}$ are time-varying, and thus $\mathbf{\Gamma}_4$ and $\mathbf{\Gamma}_5$ are linearly independent. Moreover, as the value of $\Delta t_{k,\ell}$ is varying with different time intervals, $\mathbf{\Gamma}_1$, $\mathbf{\Gamma}_2$, and $\mathbf{\Gamma}_3$ are general vectors and thus linearly independent. As an immediate result, all these general vectors, $\mathbf{\Gamma}_1$, $\mathbf{\Gamma}_2$, $\mathbf{\Gamma}_3$, $\mathbf{\Gamma}_4$, and $\mathbf{\Gamma}_5$ are linearly independent of $\mathbf{I}_3$. According to that, we do matrix elementary transformations on $\mathbf{M}_\ell$ to facilitate the search for the nullspace:

$$\begin{aligned}\mathbf{M}_\ell &= \mathbf{\Pi}[\,\mathbf{0}_3 \quad \mathbf{0}_3 \quad \mathbf{\Gamma}_1 \quad \mathbf{\Gamma}_2 \quad -\mathbf{I}_3 \quad \mathbf{\Gamma}_3 \quad \mathbf{\Gamma}_4 \quad \mathbf{\Gamma}_5 \;\mid\; \mathbf{I}_3\,] \\ &\sim \mathbf{\Pi}[\,\mathbf{0}_3 \quad \mathbf{0}_3 \quad \mathbf{\Gamma}_1 \quad \mathbf{\Gamma}_2 \quad -\mathbf{I}_3 \quad \mathbf{\Gamma}_3 \quad \mathbf{\Gamma}_4 \quad \mathbf{\Gamma}_5 \;\mid\; \mathbf{0}_3\,]\end{aligned}$$

from which we can find that $\mathbf{M}_\ell$ is rank deficient by 9 (i.e., the dimension of its nullspace is 9). Specifically, $\forall \ell \geqslant k$, we can find that the nullspace of $\mathbf{M}$ consists of the following *nine* directions, as

$$\mathbf{null}(\mathbf{M}) = \underset{\text{col.}}{\text{span}} \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{69}$$

which may be interpreted as follows:

Remark 1. *The proposed robocentric model possesses (9) unobservable directions: the first 6 DOFs are correlated to the orientation (3) and position (3) of the global frame,[2] while the remaining 3 DOFs belong to the same translation (3) simultaneously applied to the sensing platform and the landmark(s), which agrees with our intuition that relative (camera and IMU) measurements do not provide any global information in the VINS estimation, in analogy to the SLAM case (Huang et al., 2010).*

4.2. Observability with composition

After update at time-step ℓ, the estimates of ${}^{R_k}\mathbf{x}_\ell$ and ${}^{R_k}\mathbf{p}_L$ are obtained, we have the following linearized model from time-step k to ℓ, including the composition step, as

$$\tilde{\mathbf{x}}_\ell = \breve{\mathbf{V}}_\ell \mathbf{\Psi}(\ell,k)\tilde{\mathbf{x}}_k = \breve{\mathbf{\Psi}}(\ell,k)\tilde{\mathbf{x}}_k \tag{70}$$

where

$$\begin{aligned}\breve{\mathbf{V}}_\ell &= \begin{bmatrix} \mathbf{V}_\ell & \mathbf{0}_{24\times 3} \\ \mathbf{L}_\ell & \mathbf{N}_\ell \end{bmatrix} = \begin{bmatrix} \mathbf{V}_\ell & \mathbf{0}_{24\times 3} \\ \dfrac{\partial \tilde{\mathbf{x}}_\ell}{\partial {}^{R_k}\tilde{\mathbf{x}}_\ell} & \dfrac{\partial \tilde{\mathbf{x}}_\ell}{\partial {}^{R_k}\tilde{\mathbf{p}}_L} \end{bmatrix}, \\ \mathbf{L}_\ell &= \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \left\lfloor {}^{R_\ell}\hat{\mathbf{p}}_L \times \right\rfloor & -{}_k^{\ell}\mathbf{C}_{\hat{\bar{q}}} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix}, \\ \mathbf{N}_\ell &= {}_k^{\ell}\mathbf{C}_{\hat{\bar{q}}}\end{aligned} \tag{71}$$

For brevity of analysis, only the pertinent entries of $\breve{\mathbf{\Psi}}(\ell,k)$ are shown in the following:

$$\breve{\mathbf{\Psi}}(\ell,k) = \breve{\mathbf{V}}_\ell \mathbf{\Psi}(\ell,k) =$$

$$\begin{bmatrix} \breve{\mathbf{\Psi}}_{11} & \mathbf{0}_3 & \mathbf{0}_3 & \breve{\mathbf{\Psi}}_{14} & \mathbf{0}_3 & \mathbf{0}_3 & \breve{\mathbf{\Psi}}_{17} & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \breve{\mathbf{\Psi}}_{22} & \breve{\mathbf{\Psi}}_{23} & \breve{\mathbf{\Psi}}_{24} & \breve{\mathbf{\Psi}}_{25} & \breve{\mathbf{\Psi}}_{26} & \breve{\mathbf{\Psi}}_{27} & \breve{\mathbf{\Psi}}_{28} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \breve{\mathbf{\Psi}}_{33} & \breve{\mathbf{\Psi}}_{34} & \mathbf{0}_3 & \mathbf{0}_3 & \breve{\mathbf{\Psi}}_{37} & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \breve{\mathbf{\Psi}}_{63} & \breve{\mathbf{\Psi}}_{64} & \mathbf{0}_3 & \breve{\mathbf{\Psi}}_{66} & \breve{\mathbf{\Psi}}_{67} & \breve{\mathbf{\Psi}}_{68} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \breve{\mathbf{\Psi}}_{93} & \breve{\mathbf{\Psi}}_{94} & \breve{\mathbf{\Psi}}_{95} & \breve{\mathbf{\Psi}}_{96} & \breve{\mathbf{\Psi}}_{97} & \breve{\mathbf{\Psi}}_{98} & \breve{\mathbf{\Psi}}_{99} \end{bmatrix}$$

$$\breve{\mathbf{\Psi}}_{93} = -{}_k^{\ell}\mathbf{C}_{\hat{\bar{q}}}\mathbf{\Phi}_{53} \tag{72}$$

$$\breve{\mathbf{\Psi}}_{94} = \left\lfloor {}^{R_\ell}\hat{\mathbf{p}}_L \times \right\rfloor \mathbf{\Phi}_{44} - {}_k^{\ell}\mathbf{C}_{\hat{\bar{q}}}\mathbf{\Phi}_{54} \tag{73}$$

$$\breve{\mathbf{\Psi}}_{95} = -{}_k^{\ell}\mathbf{C}_{\hat{\bar{q}}} \tag{74}$$

$$\breve{\mathbf{\Psi}}_{96} = -{}_k^{\ell}\mathbf{C}_{\hat{\bar{q}}}\mathbf{\Phi}_{56} \tag{75}$$

$$\breve{\mathbf{\Psi}}_{97} = \left\lfloor {}^{R_\ell}\hat{\mathbf{p}}_L \times \right\rfloor \mathbf{\Phi}_{47} - {}_k^{\ell}\mathbf{C}_{\hat{\bar{q}}}\mathbf{\Phi}_{57} \tag{76}$$

$$\breve{\mathbf{\Psi}}_{98} = -{}_k^{\ell}\mathbf{C}_{\hat{\bar{q}}}\mathbf{\Phi}_{58} \tag{77}$$

$$\breve{\mathbf{\Psi}}_{99} = {}_k^{\ell}\mathbf{C}_{\hat{\bar{q}}} \tag{78}$$

Note that the measurement model of (61) becomes linear:

$$\mathbf{z}_\ell = {}^{R_\ell}\mathbf{p}_L, \quad {}^{R_\ell}\mathbf{p}_L = {}_k^{\ell}\mathbf{C}_{\bar{q}}({}^{R_k}\mathbf{p}_L - {}^{R_k}\mathbf{p}_{I_\ell}) \tag{79}$$

and the measurement Jacobian with respect to $\tilde{\mathbf{x}}_\ell$ is

$$\breve{\mathbf{H}}_\ell = [\,\mathbf{0}_{3\times 24} \;\mid\; \mathbf{I}_3\,] \tag{80}$$

Therefore, after composition we have the block row, $\mathbf{M}_\ell$, of $\mathbf{M}$ in the form of

$$\begin{aligned}\mathbf{M}_\ell &= \breve{\mathbf{H}}_\ell \breve{\mathbf{\Psi}}_{\ell,k} \\ &= \begin{bmatrix} \mathbf{0}_3 & \mathbf{0}_3 & \breve{\mathbf{\Psi}}_{93:94} & -{}^{\ell}_{k}\mathbf{C}_{\hat{\bar{q}}} & \breve{\mathbf{\Psi}}_{96:98} \mid {}^{\ell}_{k}\mathbf{C}_{\hat{\bar{q}}} \end{bmatrix}\end{aligned}$$

Note that for generic motion (i.e., $\boldsymbol{\omega} \neq \mathbf{0}_{3\times 1}$ and $\mathbf{a} \neq \mathbf{0}_{3\times 1}$), $\breve{\mathbf{\Psi}}_{93}$, $\breve{\mathbf{\Psi}}_{94}$, $\breve{\mathbf{\Psi}}_{96}$, $\breve{\mathbf{\Psi}}_{97}$, and $\breve{\mathbf{\Psi}}_{98}$ are linearly independent, and obviously the same nullspace as that of the propagation and update (see (69)) can be obtained.

Remark 2. *For the proposed robocentric model, changing local frame of reference (by composition) does not alter the unobservable subspace.*

Up to now, we have completely shown that the proposed robocentric VINS model has an invariant unobservable subspace that is independent of the linearization points under generic motions, which guarantees that the proposed R-VIO has not only *correct* unobservable dimension as in Huang et al. (2010), Li and Mourikis (2013a), Hesch et al. (2014a,b), and Huang et al. (2014), but also *constant* unobservable directions, thus being expected to improve estimation consistency.

4.3. Observability under special motions

Depending on the motion undertaken, the system observability properties might change in some degenerate cases. Identifying and understanding such special motions is essential for improving the VINS performance, especially in practice. For example, one of the most commonly seen cases is the planar motion (the translation is only excited in the x–y plane, and the rotation is only about the z-axis) and the recent analysis on the world-centric VINS (Wu et al., 2017) has pointed out that in this type of motion two more unobservable directions emerge: (i) the global orientation and (ii) the scale. It should be noted that for the proposed robocentric system model the global orientation has already been shown to be unobservable (see (69)). Therefore, in what follows, we study in-depth the observability under special motions by focusing on the scale (un)observability which may directly affect the performance.

4.3.1. Effect of scaling on VINS states. We are first to understand the implications of an underlying scale factor applied to the state vector of the proposed robocentric system, which is to form the basis for identifying the degenerate motions causing the special unobservable directions.

Lemma 1. *For the proposed robocentric model, given the actual state,* $\mathbf{x}$*, and the underlying state,* $\mathbf{x}'$*, that are related through a scale factor, s, there exists the following relation between the corresponding error states (see (44))*:

$$\begin{bmatrix} \delta\boldsymbol{\theta}_G \\ {}^{R_k}\tilde{\mathbf{p}}_G \\ {}^{R_k}\tilde{\mathbf{g}} \\ \delta\boldsymbol{\theta}_I \\ {}^{R_k}\tilde{\mathbf{p}}_I \\ \tilde{\mathbf{v}}_I \\ \tilde{\mathbf{b}}_g \\ \tilde{\mathbf{b}}_a \\ {}^{R_k}\tilde{\mathbf{p}}_L \end{bmatrix} = \begin{bmatrix} \delta\boldsymbol{\theta}'_G \\ {}^{R_k}\tilde{\mathbf{p}}'_G \\ {}^{R_k}\tilde{\mathbf{g}}' \\ \delta\boldsymbol{\theta}'_I \\ {}^{R_k}\tilde{\mathbf{p}}'_I \\ \tilde{\mathbf{v}}'_I \\ \tilde{\mathbf{b}}'_g \\ \tilde{\mathbf{b}}'_a \\ {}^{R_k}\tilde{\mathbf{p}}'_L \end{bmatrix} + (s-1) \begin{bmatrix} \mathbf{0}_{3\times 1} \\ {}^{R_k}\hat{\mathbf{p}}'_G \\ \mathbf{0}_{3\times 1} \\ \mathbf{0}_{3\times 1} \\ {}^{R_k}\hat{\mathbf{p}}'_I \\ \hat{\mathbf{v}}'_I \\ \mathbf{0}_{3\times 1} \\ -{}^{I}\hat{\mathbf{a}}' \\ {}^{R_k}\hat{\mathbf{p}}'_L \end{bmatrix} \Rightarrow \tag{81}$$

$$\tilde{\mathbf{x}} = \tilde{\mathbf{x}}' + (s-1)\mathbf{u}$$

Proof. See Appendix C. □

4.3.2. Special motions for scale unobservability. It becomes clear from (81) that if the proposed robocentric VINS estimation is metrically scaled by a factor of s, then the error state (and, hence, the state) would be changed along the direction of $\mathbf{u}$ by a factor of $(s-1)$. However, as evident from the proof (see Appendix C) we might not distinguish this scale ambiguity from the camera and IMU measurements, which implies that the direction of scale is *unobservable*. The following analysis further identifies the special motions that can cause the scale unobservability for the proposed robocentric system.

Lemma 2. *For the proposed robocentric model, there exist such special motions that can cause scale unobservable: (i) no rotation, with*

$$\Delta t_{k,\ell}\hat{\mathbf{v}}'_{I_\ell} = -\frac{1}{2}\Delta t^2_{k,\ell}\,{}^{\ell}\hat{\mathbf{a}}, \quad \forall \ell \geqslant k \tag{82}$$

(ii) constant local acceleration, with $\hat{\mathbf{v}}'_{I_\tau} \equiv \mathbf{0}$, $\forall \tau \in [t_k, t_\ell]$*; that is, the system is either stationary or purely rotating.*

Proof. See Appendix D. □

As a final remark, it is clear from the above lemma that the scale unobservable direction does exist when: (i) (82) holds true (e.g., may occur during the deceleration motion), or (ii) the sensing platform stays in the same place. However, these two cases can easily be mitigated in the estimation. Specifically, in the case of (i), as it holds when $\Delta t_{k,\ell} \to 0$, we can simply increase $\Delta t_{k,\ell}$ in practice to avoid the scale change. While in the case of (ii), we will confront low parallax, but the inverse-depth measurement model used by the proposed R-VIO (see (29)) forces it mainly to exploit the orientation information from the measurements, thus holding the scale. It should be pointed out that in contrast to the world-centric remedy (Wu et al., 2017), which fused the wheel odometer measurements, the proposed R-VIO does not require any additional sensor to address this scale issue, thus revealing the better adaptability and robustness, as shown later in the real-world experiment.

Table 1. Sensor parameters in simulation.

Parameter	Value
IMU rate	100 [Hz]
Gyroscope noise density (σ_g)	1.1220×10^{-4} [rad/s/$\sqrt{\text{Hz}}$]
Gyroscope random walk (σ_{wg})	5.6323×10^{-6} [rad/s^2/$\sqrt{\text{Hz}}$]
Accelerometer noise density (σ_a)	5.0119×10^{-4} [m/s^2/$\sqrt{\text{Hz}}$]
Accelerometer random walk (σ_{wa})	3.9811×10^{-5} [m/s^3/$\sqrt{\text{Hz}}$]
Camera rate	10 [Hz]
Image noise level (σ_{im})	1.5 [pixels]
Magnitude of gravity	9.8038 [m/s^2]

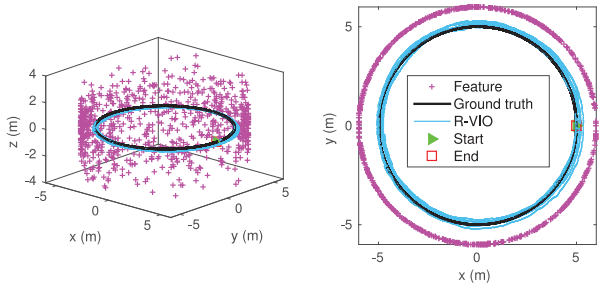


Fig. 2. Simulation scenario: a camera/IMU pair moves along a circular path of radius 5 m (black) at an average speed 1.0 m/s. The camera with 45° field of view observes point features (pink) randomly distributed on a circumscribing cylinder of radius 6 m.

5. Simulation results

In this section, we present the Monte Carlo simulation results that verify the analysis provided in the preceding sections, and demonstrate the performance of the proposed R-VIO in comparison with two world-centric EKF-based counterparts: (i) the standard (Std)-MSCKF (Mourikis and Roumeliotis, 2007), as well as the state-of-the-art (ii) state-transition and observability constrained (STOC)-MSCKF (Huang et al., 2014) that enforces correct system observablility to improve estimation consistency. In particular, two metrics are used for the evaluation: (a) the root-mean-squared error (RMSE), which offers a concise measure of the filter's accuracy, and (b) the normalized estimation error squared (NEES), which provides a standard criterion for evaluating the given filter's consistency (Bar-Shalom et al., 2001). In order to make a fair comparison, we implemented all three filters using the same parameters, such as the number of features used per update, and processing the same data of 50 Monte Carlo trails which were generated with real microelectromechanical system (MEMS) sensors' quality (see Table 1 and Figure 2 for details).

The statistical results over 50 Monte Carlo trails are shown in Figure 3, while Table 2 provides the average RMSE and NEES results for all the filters, which clearly show that the proposed R-VIO as expected outperforms the Std-MSCKF, as well as the STOC-MSCKF in terms of

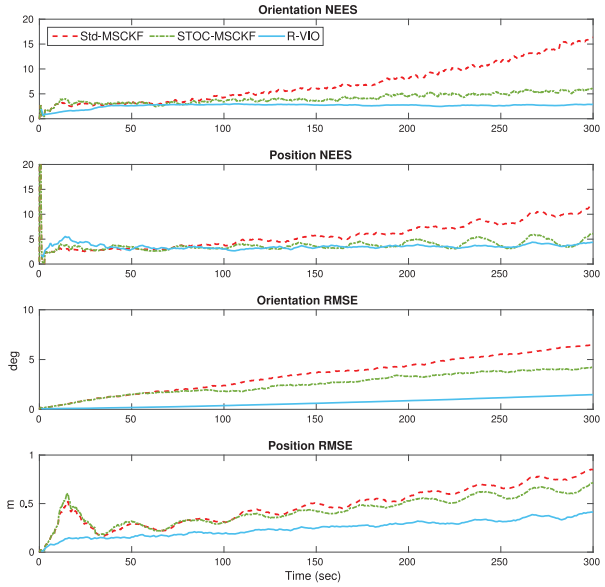


Fig. 3. Simulation results: the statistics of NEES and RMSE of orientation and position over 50 Monte Carlo trials.

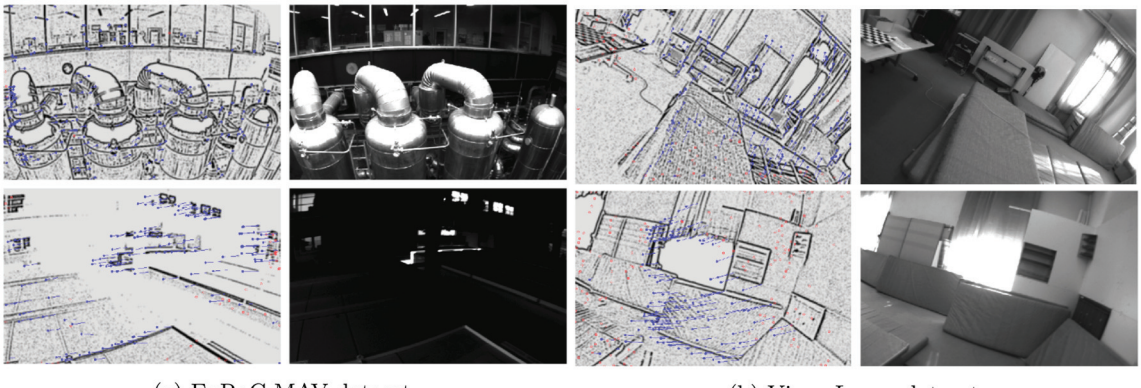
Table 2. Average RMSE and NEES corresponding to Figure 3.

	Orientation [deg]	Position [m]	Orientation NEES	Position NEES
Std-MSCKF	3.470	0.477	7.048	5.810
STOC-MSCKF	2.523	0.430	4.096	3.793
R-VIO	**0.694**	**0.253**	**2.615**	**3.552**

accuracy (smaller RMSE) and consistency (NEES closer to *three*), attributed to the novel reformulation of the system.

6. Experimental results

We further experimentally validate the proposed R-VIO, in both indoor and outdoor environments, using both the public benchmark dataset on MAVs and the real urban driving data collected by our own sensing platform on a car. As described in Algorithm 1, we implemented that in C++ multithread framework.[3] In the *front end*, the visual tracking handler extracts features from the image using the Shi-Tomasi corner detector (Shi and Tomasi, 1994), that are being tracked between consecutive images with the KLT algorithm (Baker and Matthews, 2004). In particular, to deal with the varying lighting conditions in practice, a combined operation of the adaptive thresholding and box blurring was applied for every image before doing the KLT tracking. This effectively mitigates the sharp change of illumination while outlining the environmental structures even in the dark area (see Figure 4), which is experimentally helpful for the feature detection. In addition, to remove the bad tracks from the tracking results, we realized the gyro-



(a) EuRoC MAV dataset.

(b) Vicon Loops dataset.

(c) Urban driving test.

Fig. 4. Visual tracking: the post-processing results (left) and the respective raw images (right). Inliers (blue dots) are tracked between pairwise images with outliers (red dots) being rejected by the RANSAC. In particular, the tracking results have been visualized by the inliers' tracks (blue lines). Note that, this front end is able to process: (i) dusky, (ii) blurred, as well as (iii) overexposed image streams of real-world environments.

aided, two-point random sample consensus (RANSAC) algorithm (Troiani et al., 2014). In the end, the tracking history of all the inliers are stored by a first-in, first-out (FIFO) data structure, in order to be efficiently queried during the estimation.

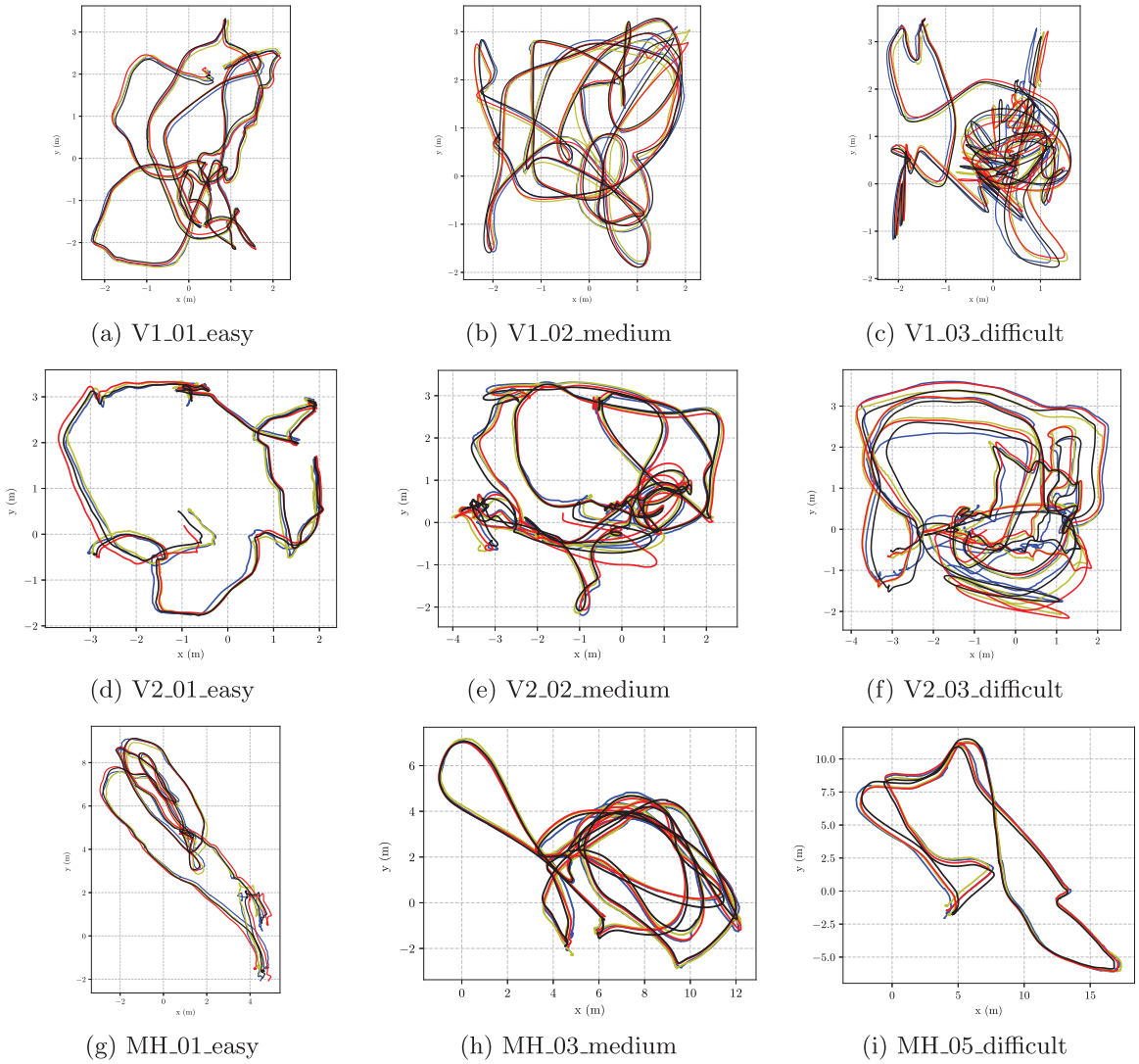
Once the visual tracking is done, the *back end* processes all the visual and inertial measurements using the proposed robocentric EKF. In particular, for the features losing track in the current image, we use all their measurements within the current sliding window for update, while for those reaching the maximum tracking length (e.g., the sliding-window size) we use a subset (e.g., 1/2) of their respective measurements and keep the rest for next update. All the following tests run on an Intel Core i7-4710MQ @ 2.5 GHz laptop in *real time*.

6.1. EuRoC MAV dataset

We tested the proposed R-VIO first using all 11 sequences in the EuRoC MAV dataset (Burri et al., 2016) for which a FireFly helicopter equipped with VI-sensor (an ADIS16448 IMU @ 200 Hz, and dual cameras 752×480 pixels @ 20 Hz) was used for flying data collection. In this test, only the left camera images were used for vision inputs and 200 features were uniformly extracted based on each image. The sliding-window size was set up to 20 (i.e., about 1 s memory for the most recent relative motion). We compared the proposed R-VIO with two state-of-the-art world-centric sliding-window optimization-based VINS, the OKVIS[4] (Leutenegger et al., 2015) and the VINS-Mono[5] (Qin et al., 2018), both of which perform nonlinear batch optimization in the back ends using the Google Ceres optimizer (Agarwal et al., 2010). It should be noted that in order to be a fair comparison between the three systems, we turned off the loop closure of VINS-Mono in the tests. As the 6-DOF pose ground truths are provided for all the sequences, we employed the evaluation tool[6] (Zhang and Scaramuzza, 2018) to generate the results of comparison. Specifically, the results of pose RMSE and the computation time in Table 3 show both the absolute pose accuracy and the computational efficiency of the systems. Figure 5 depicts the estimated trajectories of nine representative sequences, and the corresponding relative pose errors are shown in Figure 6. It is important to note that the proposed R-VIO does not utilize

Table 3. Pose estimation accuracy (RMSE) and computation time (per image) on the EuRoC dataset.

		R-VIO			OKVIS			VINS-Mono		
	Length [m]	Orientation [deg]	Position [m]	Time [ms]	Orientation [deg]	Position [m]	Time [ms]	Orientation [deg]	Position [m]	Time [ms]
V1_01_easy	58.6	**0.106**	**0.080**	**24.6**	0.099	0.083	43.9	0.110	0.089	68.5
V1_02_medium	75.9	**0.033**	**0.108**	**26.5**	0.048	0.192	45.7	0.057	0.110	49.6
V1_03_difficult	79.0	**0.021**	**0.121**	**25.5**	0.069	0.199	46.5	0.106	0.188	35.2
V2_01_easy	36.5	0.038	0.158	**24.0**	0.013	0.154	46.7	**0.036**	**0.088**	35.2
V2_02_medium	83.2	0.030	0.162	**26.3**	0.023	0.216	44.5	**0.074**	**0.160**	46.6
V2_03_difficult	86.1	**0.077**	**0.274**	**23.8**	0.060	0.287	43.1	0.056	0.278	27.3
MH_01_easy	80.6	0.036	0.187	**27.9**	0.027	0.267	44.3	**0.012**	**0.174**	67.1
MH_02_easy	73.5	0.021	0.305	**25.8**	0.040	0.318	43.2	**0.015**	**0.221**	63.6
MH_03_medium	130.9	0.025	0.287	**27.3**	0.058	0.306	41.8	**0.032**	**0.221**	64.5
MH_04_difficult	91.7	0.072	0.761	**25.7**	**0.078**	**0.297**	43.4	0.014	0.371	58.0
MH_05_difficult	97.6	0.013	0.445	**27.3**	0.046	0.502	45.9	**0.008**	**0.351**	60.6
Mean (Sequence 1–6)	—	**0.050**	**0.150**	**25.1**	0.052	0.188	45.0	0.073	0.152	48.2
Mean (Sequence 7–11)	—	0.033	0.397	**26.8**	0.049	0.338	43.7	**0.016**	**0.267**	62.7



(a) V1_01_easy (b) V1_02_medium (c) V1_03_difficult

(d) V2_01_easy (e) V2_02_medium (f) V2_03_difficult

(g) MH_01_easy (h) MH_03_medium (i) MH_05_difficult

Fig. 5. Trajectory estimates on EuRoC dataset: R-VIO (blue), OKVIS (yellow), and VINS-Mono (red) with the ground truth (black).

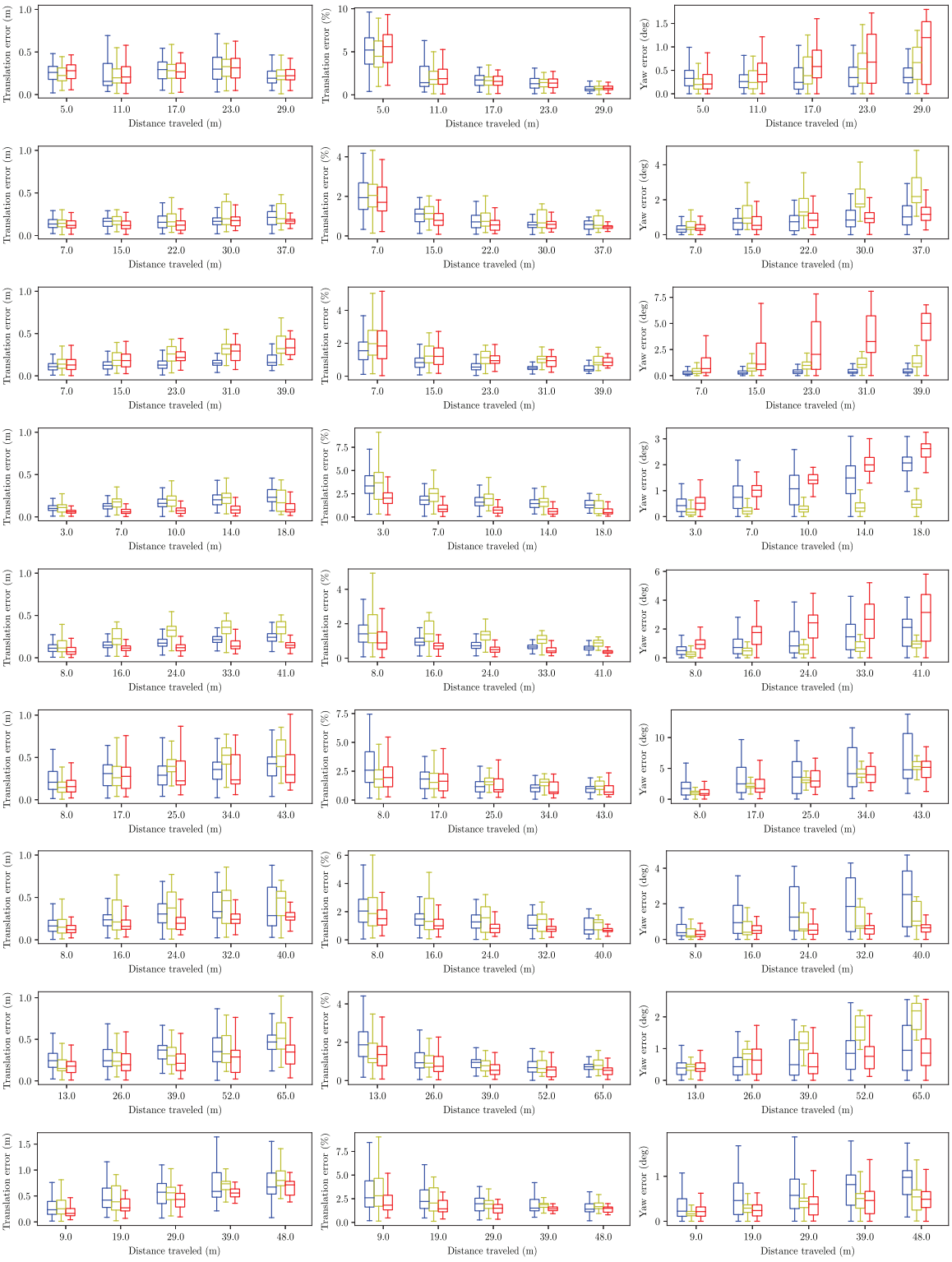
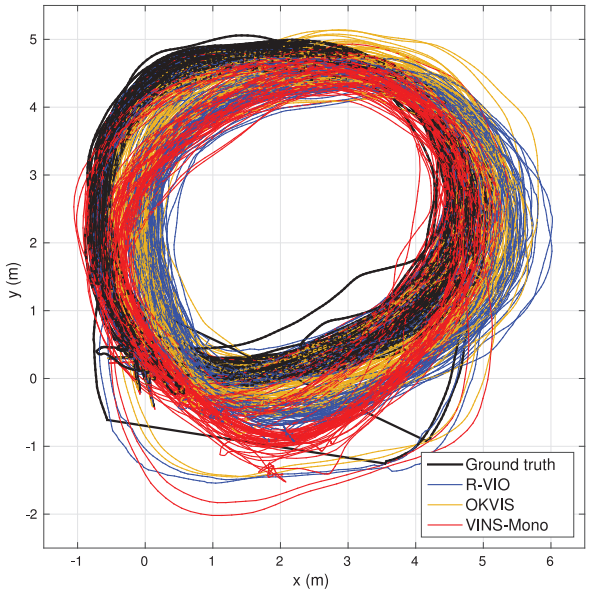
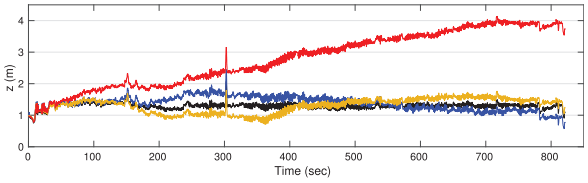


Fig. 6. Relative pose errors corresponding to the trajectories shown in Figure 5(a)–(i). Each row presents the relative translation error (left), the percentage of the relative translation error (middle), and the relative yaw error (right) of R-VIO (blue), OKVIS (yellow), and VINS-Mono (red) with respect to different segments of trajectory of each sequence, respectively.

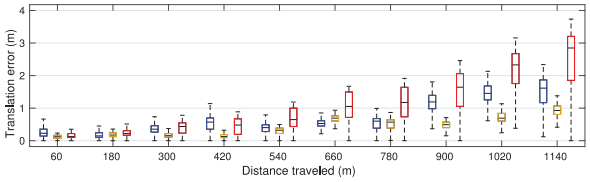
any kind of map, whereas both OKVIS and VINS-Mono do. Nevertheless, it is clear from Table 3 that on this dataset the proposed R-VIO performs very competitively, or even *better* in some sequences, with the OKVIS and VINS-Mono; this, however, is achieved at the *least* computational cost among all three systems considered.



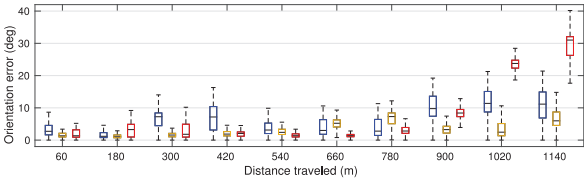
(a) Overhead plot of the estimated trajectories.



(b) Altitude profiles, i.e., z-components of estimator outputs.



(c) Absolute translation error.



(d) Absolute orientation error.

Fig. 7. Vicon Loops dataset evaluation: R-VIO (blue), OKVIS (yellow) and VINS-Mono (red). In particular, (a) and (b) illustrate the system performance in x-y plane and z-axis; (c) and (d) are the error statistics in terms of median, 5th and 95th percentiles.

6.2. Vicon Loops dataset

We further tested the proposed R-VIO on a long-term indoor dataset presented in Leutenegger et al. (2015). This dataset was collected using a handheld VI-sensor, moving mostly in circles at the speed around 2 m/s, which lasted almost 14 minutes and traveled about 1,200 meters in total. Similarly, the images from left camera were used for vision inputs and 200 features were uniformly extracted per image, while this time a shorter sliding window of size 10 was used in order to handle the mostly short tracking lengths of features caused by constantly, highly dynamic motion for stimulating MAV flying. Again, we compared the performance of the proposed R-VIO with both OKVIS and VINS-Mono while using the metric introduced by Leutenegger et al. (2015); that is, to evaluate the absolute pose errors (between ground truth and the estimates) based on different distances traveled. To this end, for each travel interval the orientation and translation errors are calculated with respect to the starting ground-truth pose of each interval. Comprehensively, Figure 7 shows the estimated trajectories and the error statistics based on a set of distances, $\{60, 180, 300, 420, 540, 660, 780, 900, 1{,}020, 1{,}140\}$, noting that the results of OKVIS were generated using the original data provided by Leutenegger et al. (2015). It should also be pointed out that as this dataset was starting from motion (i.e., there was no static phase at the beginning), the proposed R-VIO only used the first *single* IMU measurement for initialization, while the VINS-Mono was initialized based on a visual–inertial alignment procedure using multiple measurements (Qin et al., 2018). As is evident from Figure 7, all three systems drift in both the orientation and the translation as the distance traveled becomes longer, while without loop closure the VINS-Mono accumulates the obviously larger drifts, in both z-component and yaw angle, than the OKVIS and the proposed R-VIO.

6.3. Urban driving test

We have also performed a road test for R-VIO with our own sensing platform (an Xsens Mti-G-710 GNSS, and a FLIR Bumblebee2 stereo pair $1{,}024 \times 768$ pixels @ 15 Hz) on a car by driving on the streets of Newark, DE. The IMU provided measurements at 400 Hz, while the GPS signal was received at 4 Hz as the (position) ground truth. Similarly, only the left camera images were used for vision inputs with 200 features being uniformly extracted from each image. It is important to note that this test is very challenging primarily due to: (i) several traffic lights where we must stop and wait for 15–25 seconds; (ii) frequent stop/

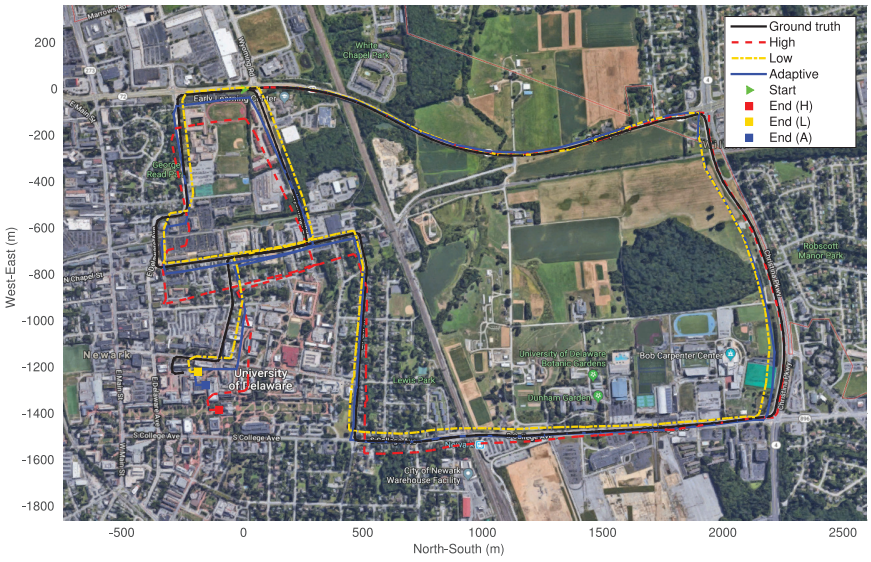


Fig. 8. Trajectory estimates plotted over the map of Newark, DE. The initial position of the vehicle is marked by a green triangle. The black solid line corresponds to the trajectory of ground truth, in addition to the red dashed line to the high update rate (H), the yellow dash-dotted line to the low update rate (L), and the blue solid line to the adaptive update rate (A), with the ends of trajectories being marked by the squares in their respective colors.

Table 4. Estimation accuracy (RMSE) in the urban driving test.

	Length / Duration	Maximum speed	Position RMSE		
			x [m]	y [m]	z [m]
H	9.8 km / 15 min	86 km/h	30.934	68.561	8.418
L	— / —	—	33.984	15.883	10.426
A	— / —	—	**24.222**	**18.901**	**7.689**

Fig. 9. Snapshots during the urban driving test. From left to right and top to bottom: (i) at the start, (ii) at a turning with yield sign, (iii) at a pedestrian traffic light, (iv) at a regular traffic light, (v) at a free intersection with stop sign, and (vi) in the campus area.

yield signs before which we must stop or decelerate; (iii) dynamic scenes including running vehicles and pedestrians in the vicinity; (iv) strong lens flare when driving facing the sun; and (v) high speeds of vehicle when driving on some particular road sections (see Figure 9).

As we discussed earlier, both (i) and (ii) are the degenerate scenes that may cause the scale to be unobservable for the proposed VIO model. The usage of inverse depth-based measurement model partially solved the scale drift during the static phase, while for the deceleration phase we tested three update rates: high (15 Hz), low (7.5 Hz), and adaptive (switching between the high and low). For actual implementation, we employed a heuristic switching logic for the adaptive mode which is to be optimized in our future work; that is, the R-VIO changes the frequency of update from 15 to 7.5 Hz after detecting the speed descending ($\mathbf{\Delta v} < \mathbf{0}$) for at least 2 seconds, and changes it back to 15 Hz once detecting the speed ascending ($\mathbf{\Delta v} > \mathbf{0}$) for at least 2 seconds. The results are summarized in Table 4, while Figure 8 depicts the estimated trajectories of all three update rates. We should note that when using the high update rate, the R-VIO captured the motion of vehicle better than using the low update rate, given an instance that after the first turning our vehicle sped up to 86 km/h and for that the trajectory under high update rate fitted the ground truth better. At the second turning, a series of deceleration events occurred due to the busy traffic at the intersection, as a consequence, the drifts in scale biased trajectory estimation afterwards. In contrast, by lowering down the update rate the R-VIO could compensate for the scale drifts and made the estimated

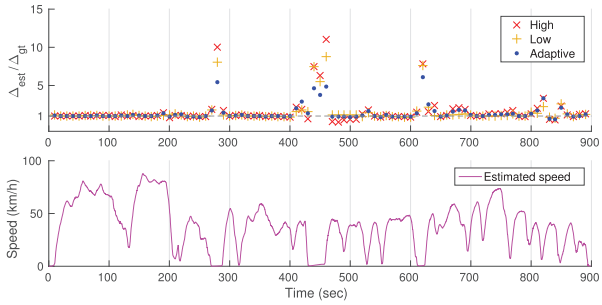


Fig. 10. Statistics of the scale of estimate versus vehicle speed for the three update rates.

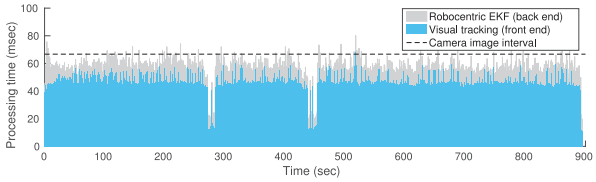


Fig. 11. Time consumption per image (at high update rate): the average processing rate of R-VIO is 20.5 Hz as compared with the image rate of 15 Hz, which satisfies real-time requirement.

trajectory closer to the ground truth. Thus, as expected, the proposed adaptive mode was able to benefit from both of the aforementioned advantages. In addition to the inference, we further confirm it through a test which the differences of translation between consecutive poses of R-VIO's estimates, Δ_{est}, and of the ground truth, Δ_{gt}, for each update rate are accumulated for every 10 seconds, accordingly the scale can be estimated by $\frac{\Delta_{est}}{\Delta_{gt}}$. The results referring to the estimated speed are shown in Figure 10, from which we can find that the distinct scale variance ($\geqslant 2$) only emerged when either a sharp deceleration occurred (e.g., 600–610 s) or vehicle was stopped (e.g., 430–455 s); which agrees with our analysis provided in the preceding section and shows that the drifts in scale can be compensated for by using the proposed adaptive method. Among the three modes, the adaptive one performs the best in terms of median absolute deviation (MAD), a robust measure of the variability of a univariate sample, of scale 0.079, and 0.114 and 0.099 for high and low update rates, respectively, with respect to the ground truth.

Note also that, as the local gravity is jointly estimated, the z-axis drifts are much smaller than the x–y position errors. In the test a sliding window of size 20 was used and the average processing time of pipeline was 48.7 ms per image including 40.8 ms spent on the visual tracking and RANSAC outlier rejection and the other 7.9 ms on robocentric EKF (see Figure 11). In this challenging driving scenario, without utilizing any kind of map, the proposed R-VIO achieves the average position RMSEs of 0.77% (high update rate), 0.40% (low update rate), and 0.32% (adaptive update rate) of the total 9.8 km driving distance.

7. Conclusion and future work

In this paper, we have reformulated the VINS with respect to a moving local frame and developed a lightweight, high-precision, robocentric visual–inertial odometry algorithm, termed R-VIO. With this novel reformulation, we have analytically shown that the proposed R-VIO does not suffer from the observability mismatch issue under generic motions that is encountered by the world-centric counterparts, while even in the degenerate motion cases the observability issue can be easily compensated for without using additional sensor information, thus providing better consistency, accuracy, and robustness. Extensive Monte Carlo simulation as well as real-world experiments using different sensing platforms and navigating in different environments are performed to thoroughly validate our theoretical analysis and show that the proposed R-VIO is versatile and robust to different types of motions and environments, and is capable of providing long-term, high-precision 3D motion tracking in real time. In the future, we will integrate efficient loop closure and online mapping into the current robocentric system in order to bound localization errors, as well as performing online calibration of intrinsic and extrinsic sensor parameters to further improve performance.

Acknowledgments

The authors would like to thank Professor Stefan Leutenegger from Imperial College London and Simon Lynen from ASL/ETH Zurich (at that time) for sharing the Vicon Loops dataset and OKVIS data presented in Leutenegger et al. (2015), and give special thanks to our colleague Patrick Geneva from RPNG/UD for his invaluable help in the urban driving test.

Funding

This research was partially supported by the University of Delaware (UD) College of Engineering, UD Cybersecurity Initiative, the Delaware NASA/EPSCoR Seed Grant, the NSF (grant number IIS-1566129), the DTRA (grant number HDTRA1-16-1-0039), and Google Daydream.

ORCID iD

Zheng Huai https://orcid.org/0000-0002-9840-1843

Notes

1. Throughout this paper, $k, k+1, \ldots$ indicate the image time steps, while $\tau-1, \tau, \ldots$ are the IMU time steps between every two consecutive images. The subscript $\ell|i$ refers to the estimate of a quantity at time-step ℓ, after all measurements up to time-step i have been processed. Here $\hat{x}$ is used to denote the estimate of a random variable x, with $\tilde{x} = x - \hat{x}$ the additive error in this estimate, and $\mathbf{I}_n$ and $\mathbf{0}_n$ are the $n \times n$ identity and zero matrices, respectively. Finally, the left superscript represents the reference frame with respect to which the vector is expressed.

2. Note that the global frame {G} is not necessarily to be gravity-aligned and can be reset to any pose at any time.
3. The open-source code is available at https://github.com/rpng/R-VIO.
4. See https://github.com/ethz-asl/okvis_ros
5. See https://github.com/HKUST-Aerial-Robotics/VINS-Mono
6. See https://github.com/uzh-rpg/rpg_trajectory_evaluation

References

Agarwal S and Mierle K and others (2010) Ceres Solver. Available at: http://ceres-solver.org.

Baker S and Matthews I (2004) Lucas–Kanade 20 years on: A unifying framework. *International Journal of Computer Vision* 56(3): 221–255.

Bar-Shalom Y, Li XR and Kirubarajan T (2001) *Estimation with Applications to Tracking and Navigation*. New York: John Wiley & Sons.

Bloesch M, Burri M, Omari S, Hutter M and Siegwart R (2017) Iterated extended Kalman filter based visual-inertial odometry using direct photometric feedback. *The International Journal of Robotics Research* 36(10): 1053–1072.

Bloesch M, Omari S, Hutter M and Siegwart R (2015) Robust visual inertial odometry using a direct EKF-based approach. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*. Hamburg, Germany: IEEE, pp. 298–304.

Breckenridge WG (1979) Quaternions proposed standard conventions. Technical report, NASA Jet Propulsion Laboratory.

Burri M, Nikolic J, Gohl P, et al. (2016) The EuRoC micro aerial vehicle datasets. *The International Journal of Robotics Research* 35(10): 1157–1163.

Castellanos JA, Martinez-Cantin R, Tardós JD and Neira J (2007) Robocentric map joining: Improving the consistency of EKF-SLAM. *Robotics and Autonomous Systems* 55(1): 21–29.

Chen Z, Jiang K and Hung JC (1990) Local observability matrix and its application to observability analyses. In: *The 16th Annual Conference of IEEE Industrial Electronic Society*, Pacific Grove, CA, pp. 100–103.

Civera J, Davison AJ and Montiel JM (2008) Inverse depth parametrization for monocular SLAM. *IEEE Transactions on Robotics* 24(5): 932–945.

Civera J, Grasa OG, Davison AJ and Montiel J (2009) 1-point RANSAC for EKF-based structure from motion. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, pp. 3498–3504.

Eckenhoff K, Geneva P and Huang G (2016) High-accuracy preintegration for visual–inertial navigation. In: *International Workshop on the Algorithmic Foundations of Robotics*, San Francisco, CA.

Forster C, Carlone L, Dellaert F and Scaramuzza D (2015) IMU preintegration on manifold for efficient visual–inertial maximum-a-posteriori estimation. In: *Robotics: Science and Systems*, Rome, Italy.

Golub GH and Van Loan CF (2012) *Matrix Computations*, Vol. 3. Baltimore, MD: JHU Press.

Guo C and Roumeliotis S (2013) IMU-RGBD camera 3D pose estimation and extrinsic calibration: Observability analysis and consistency improvement. In: *IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, pp. 2935–2942.

Hesch J, Kottas D, Bowman S and Roumeliotis S (2014a) Camera–IMU-based localization: Observability analysis and consistency improvement. *The International Journal of Robotics Research* 33(1): 182–201.

Hesch JA, Kottas DG, Bowman SL and Roumeliotis SI (2014b) Consistency analysis and improvement of vision-aided inertial navigation. *IEEE Transactions on Robotics* 30(1): 158–176.

Huai Z and Huang G (2018) Robocentric visual-inertial odometry. *Technical report, RPNG, University of Delaware*. Available at: http://udel.edu/~ghuang/papers/tr_rvio_ijrr.pdf.

Huang G, Kaess M and Leonard JJ (2014) Towards consistent visual–inertial navigation. In: *IEEE International Conference on Robotics and Automation*, Hong Kong, China, pp. 4926–4933.

Huang G, Mourikis A and Roumeliotis S (2009) A first-estimates Jacobian EKF for improving SLAM consistency. In: *Experimental Robotics*. New York: Springer, pp. 373–382.

Huang GP, Mourikis AI and Roumeliotis SI (2010) Observability-based rules for designing consistent EKF SLAM estimators. *The International Journal of Robotics Research* 29(5): 502–528.

Indelman V, Williams S, Kaess M and Dellaert F (2013) Information fusion in navigation systems via factor graph based incremental smoothing. *Robotics and Autonomous Systems* 61(8): 721–738.

Jones ES and Soatto S (2011) Visual–inertial navigation, mapping and localization: A scalable real-time causal approach. *The International Journal of Robotics Research* 30(4): 407–430.

Kelly J and Sukhatme GS (2011) Visual–inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration. *The International Journal of Robotics Research* 30(1): 56–79.

Kneip L, Chli M and Siegwart RY (2011) Robust real-time visual odometry with a single camera and an IMU. In: *British Machine Vision Conference*.

Leutenegger S, Lynen S, Bosse M, Siegwart R and Furgale P (2015) Keyframe-based visual–inertial odometry using nonlinear optimization. *The International Journal of Robotics Research* 34(3): 314–334.

Li M and Mourikis AI (2013a) High-precision, consistent EKF-based visual–inertial odometry. *The International Journal of Robotics Research* 32(6): 690–711.

Li M and Mourikis AI (2013b) Optimization-based estimator design for vision-aided inertial navigation. In: *Robotics: Science and Systems*, Berlin, Germany, pp. 241–248.

Lupton T and Sukkarieh S (2012) Visual–inertial-aided navigation for high-dynamic motion in built environments without initial conditions. *IEEE Transactions on Robotics* 28(1): 61–76.

Maybeck PS (1979) *Stochastic Models, Estimation, and Control*, Vol. 1. London: Academic Press.

Mourikis AI and Roumeliotis SI (2007) A multi-state constraint Kalman filter for vision-aided inertial navigation. In: *IEEE International Conference on Robotics and Automation*, Rome, Italy, pp. 3565–3572.

Mourikis AI, Trawny N, Roumeliotis SI, Johnson AE, Ansar A and Matthies L (2009) Vision-aided inertial navigation for spacecraft entry, descent, and landing. *IEEE Transactions on Robotics* 25(2): 264–280.

Mur-Artal R and Tardós JD (2017) Visual–inertial monocular SLAM with map reuse. *IEEE Robotics and Automation Letters* 2(2): 796–803.

Qin T, Li P and Shen S (2018) Vins-mono: A robust and versatile monocular visual–inertial state estimator. *IEEE Transactions on Robotics* 34(4): 1004–1020.

Roumeliotis SI and Burdick JW (2002) Stochastic cloning: A generalized framework for processing relative state measurements. In: *IEEE International Conference on Robotics and Automation*, Washington, DC, pp. 1788–1795.

Shi J and Tomasi C (1994) Good features to track. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Seattle, WA, pp. 593–600.

Tardós JD, Neira J, Newman PM and Leonard JJ (2002) Robust mapping and localization in indoor environments using sonar data. The *International Journal of Robotics Research* 21(4): 311–330.

Trawny N and Roumeliotis SI (2005) Indirect Kalman filter for 3D attitude estimation. Technical report, Department of Computer Science and Engineering, University of Minnesota.

Triggs B, McLauchlan PF, Hartley RI and Fitzgibbon AW (1999) Bundle adjustment: A modern synthesis. In: *International Workshop on Vision Algorithms*, Corfu, Greece, pp. 298–372.

Troiani C, Martinelli A, Laugier C and Scaramuzza D (2014) 2-point-based outlier rejection for camera–IMU systems with applications to micro aerial vehicles. In: *IEEE International Conference on Robotics and Automation*, Hong Kong, China, pp. 5530–5536.

Usenko V, Engel J, Stückler J and Cremers D (2016) Direct visual–inertial odometry with stereo cameras. In: *IEEE International Conference on Robotics and Automation*, Stockholm, Sweden, pp. 1885–1892.

Weiss S and Siegwart R (2011) Real-time metric state estimation for modular vision–inertial systems. In: *IEEE International Conference on Robotics and Automation*, Shanghai, China, pp. 4531–4537.

Wu KJ, Guo CX, Georgiou G and Roumeliotis SI (2017) VINS on wheels. In: *IEEE International Conference on Robotics and Automation*, Singapore, pp. 5155–5162.

Zhang T, Wu K, Song J, Huang S and Dissanayake G (2017) Convergence and consistency analysis for a 3-D invariant-EKF SLAM. *IEEE Robotics and Automation Letters* 2(2): 733–740.

Zhang Z and Scaramuzza D (2018) A tutorial on quantitative trajectory evaluation for visual(–inertial) odometry. In: *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Madrid, Spain, pp. 7244–7251.

Appendix A. Index to multimedia extensions

Archives of IJRR multimedia extensions published prior to 2014 can be found at http://www.ijrr.org, after 2014 all videos are available on the IJRR YouTube channel at http://www.youtube.com/user/ijrrmultimedia

Table of Multimedia Extensions

Extension	Media type	Description
1	Video	The demo of the urban driving test (R-VIO is in the adaptive mode).

Appendix B: BA using inverse-depth parameterized landmark

Assuming a single landmark, L, which has been observed from a set of consecutive robocentric frames in the sliding window, the set of corresponding camera frames is denoted by $\mathcal{C} = \{1, 2, \ldots\}$. To compute an inverse-depth estimate of L, that is, $\hat{\boldsymbol{\lambda}} = [\hat{\phi}, \hat{\psi}, \hat{\rho}]^\top$, we use the proposed inverse-depth measurement model (see (29)) ($i \in \mathcal{C}$):

$$\mathbf{z}_i = \frac{1}{h_{i,3}(\boldsymbol{\lambda})} \begin{bmatrix} h_{i,1}(\boldsymbol{\lambda}) \\ h_{i,2}(\boldsymbol{\lambda}) \end{bmatrix} + \mathbf{n}_i := \mathbf{z}_i(\boldsymbol{\lambda}) + \mathbf{n}_i \tag{83}$$

where $\mathbf{n}_i \sim \mathcal{N}(\mathbf{0}, \boldsymbol{\Lambda}_i^{-1})$ is the image noise, while the relative pose, $\mathbf{w}$, is assumed *known*. Given the measurements of L, $\left\{ \mathbf{z}_i = \left[\frac{u_i}{f}, \frac{v_i}{f}\right]^\top,\ i \in \mathcal{C} \right\}$ where f is the focal length, u_i and v_i are the undistorted pixel coordinates originated from the center of image i, we can formulate a BA problem for $\boldsymbol{\lambda}$, as

$$\begin{aligned} \boldsymbol{\lambda}^* &= \arg\min_{\boldsymbol{\lambda}} \sum_{i \in \mathcal{C}} \| \mathbf{z}_i - \mathbf{z}_i(\boldsymbol{\lambda}) \|_{\boldsymbol{\Lambda}_i} \\ &= \arg\min_{\boldsymbol{\lambda}} \sum_{i \in \mathcal{C}} \| \boldsymbol{\epsilon}_i(\boldsymbol{\lambda}) \|_{\boldsymbol{\Lambda}_i} \end{aligned} \tag{84}$$

where $\| \cdot \|_{\boldsymbol{\Lambda}_i}$ denotes the $\boldsymbol{\Lambda}_i$-weighted energy norm, and we define $\boldsymbol{\epsilon}_i$ as the residual associated to $\mathbf{z}_i$. This problem can be solved iteratively via Gauss–Newton approximation about an initial estimate of $\boldsymbol{\lambda}$, as

$$\begin{aligned} \delta\boldsymbol{\lambda}^* &= \arg\min_{\delta\boldsymbol{\lambda}} \sum_{i \in \mathcal{C}} \| \boldsymbol{\epsilon}_i(\hat{\boldsymbol{\lambda}} + \delta\boldsymbol{\lambda}) \|_{\boldsymbol{\Lambda}_i} \\ &\simeq \arg\min_{\delta\boldsymbol{\lambda}} \sum_{i \in \mathcal{C}} \| \boldsymbol{\epsilon}_i(\hat{\boldsymbol{\lambda}}) + \mathbf{H}_i \delta\boldsymbol{\lambda} \|_{\boldsymbol{\Lambda}_i} \end{aligned} \tag{85}$$

For the initial value of $\hat{\boldsymbol{\lambda}}$, we obtain $[\hat{\phi}, \hat{\psi}]^\top$ by directly using the first pixel measurement, $\mathbf{z}_1$, with the following equation:

$$\begin{bmatrix} \hat{\phi} \\ \hat{\psi} \end{bmatrix} = \begin{bmatrix} \arctan\left(\frac{v_1}{f}, \sqrt{\left(\frac{u_1}{f}\right)^2 + 1}\right) \\ \arctan\left(\frac{u_1}{f}, 1\right) \end{bmatrix} \tag{86}$$

However, the initial value of $\hat{\rho}$ can be empirically chosen, for which we use 0 to first put the estimate at infinity, then let it converge by performing iterations. The Jacobian of residual, $\mathbf{H}_i = \frac{\partial \boldsymbol{\epsilon}_i(\hat{\boldsymbol{\lambda}} + \delta\boldsymbol{\lambda})}{\partial \delta\boldsymbol{\lambda}}$, evaluated at $\hat{\boldsymbol{\lambda}}$ can be computed following the chain rule, as

$$\mathbf{H}_i = \frac{\partial \boldsymbol{\epsilon}_i}{\partial \mathbf{h}_i} \frac{\partial \mathbf{h}_i}{\partial \boldsymbol{\lambda}} \frac{\partial \boldsymbol{\lambda}}{\partial \delta\boldsymbol{\lambda}} = \frac{\partial \boldsymbol{\epsilon}_i}{\partial \mathbf{h}_i} \frac{\partial \mathbf{h}_i}{\partial \boldsymbol{\lambda}}$$

where

$$\frac{\partial \boldsymbol{\epsilon}_i}{\partial \mathbf{h}_i} = \frac{1}{\hat{h}_{i,3}} \begin{bmatrix} 1 & 0 & -\frac{\hat{h}_{i,1}}{\hat{h}_{i,3}} \\ 0 & 1 & -\frac{\hat{h}_{i,2}}{\hat{h}_{i,3}} \end{bmatrix},$$
$$\frac{\partial \mathbf{h}_i}{\partial \boldsymbol{\lambda}} = \begin{bmatrix} \frac{\partial \mathbf{h}_i}{\partial [\phi, \psi]^\top} & \frac{\partial \mathbf{h}_i}{\partial \rho} \end{bmatrix}$$
$$= \begin{bmatrix} {}^{i}_{1}\bar{\mathbf{C}}_{\hat{\bar{q}}} \begin{bmatrix} -\sin\hat{\phi}\sin\hat{\psi} & \cos\hat{\phi}\cos\hat{\psi} \\ \cos\hat{\phi} & 0 \\ -\sin\hat{\phi}\cos\hat{\psi} & -\cos\hat{\phi}\sin\hat{\psi} \end{bmatrix} & {}^{i}\hat{\mathbf{p}}_1 \end{bmatrix} \tag{87}$$

Every iteration we have an optimal inverse-depth correction, $\delta\boldsymbol{\lambda}^*$, to update current estimate, $\hat{\boldsymbol{\lambda}}$, in the form of

$$\delta\boldsymbol{\lambda}^* = \left(\sum_{i\in\mathcal{C}} \mathbf{H}_i^\top \boldsymbol{\Lambda}_i \mathbf{H}_i\right)^{-1} \left(\sum_{i\in\mathcal{C}} \mathbf{H}_i^\top \boldsymbol{\Lambda}_i \boldsymbol{\epsilon}_i\right), \tag{88}$$
$$\hat{\boldsymbol{\lambda}} \leftarrow \hat{\boldsymbol{\lambda}} + \delta\boldsymbol{\lambda}^*$$

Once $\delta\boldsymbol{\lambda}^*$ converges (e.g., less than a threshold, 10^{-6}), we find the optimal inverse-depth estimate: $\boldsymbol{\lambda}^* = \hat{\boldsymbol{\lambda}}$.

Appendix C: Proof of Lemma 1

Consider the case that the VINS estimation process is up to a scale factor of s (i.e., to recover the actual state, $\mathbf{x}$, the underlying state, $\mathbf{x}'$, has to be "scaled up" metrically). This results in the following expressions of the VINS states, in which the relative translation and landmark's position with respect to $\{R_k\}$ (see (44)) can be written as

$${}^{R_k}\mathbf{p}_G = s\,{}^{R_k}\mathbf{p}'_G \tag{89}$$
$${}^{R_k}\mathbf{p}_I = s\,{}^{R_k}\mathbf{p}'_I \tag{90}$$
$${}^{R_k}\mathbf{p}_L = s\,{}^{R_k}\mathbf{p}'_L \tag{91}$$

where ${}^{R_k}\mathbf{p}'_G$, ${}^{R_k}\mathbf{p}'_I$, and ${}^{R_k}\mathbf{p}'_L$ are the values of underlying states. Note that, the analysis presented in this proof holds true for any $t \in [t_k, t_{k+m}]$, hence we omit the time indexes for brevity of presentation. As the scale s only corresponds to the translation, the scale change does not affect the rotation. Therefore, we have the angular velocity:

$${}^{k}_{G}\mathbf{C}_{\bar{q}} = {}^{k}_{G}\mathbf{C}'_{\bar{q}}, \quad {}^{I}_{k}\mathbf{C}_{\bar{q}} = {}^{I}_{k}\mathbf{C}'_{\bar{q}} \Rightarrow$$
$$\boldsymbol{\omega} = \boldsymbol{\omega}' \tag{92}$$

With those equations, the IMU velocity and acceleration can be obtained by taking the time derivative of (90), as

$${}^{I}_{k}\mathbf{C}_{\bar{q}}\,{}^{R_k}\mathbf{v}_I = s\,{}^{I}_{k}\mathbf{C}'_{\bar{q}}\,{}^{R_k}\mathbf{v}'_I \Rightarrow$$
$$\mathbf{v}_I = s\mathbf{v}'_I, \quad {}^{I}\mathbf{a} = s\,{}^{I}\mathbf{a}' \tag{93}$$

In particular, ${}^{R_k}\mathbf{g}$ is a state having known magnitude, thus is not affected by the scaling:

$${}^{R_k}\mathbf{g} = {}^{R_k}\mathbf{g}' \tag{94}$$

Then, the gravity effects to the IMU frame can be derived as

$${}^{I}_{k}\mathbf{C}_{\bar{q}}\,{}^{R_k}\mathbf{g} = {}^{I}_{k}\mathbf{C}'_{\bar{q}}\,{}^{R_k}\mathbf{g} \Rightarrow$$
$${}^{I}\mathbf{g} = {}^{I}\mathbf{g}' \tag{95}$$

If such scale is unobservable, then the measurements from the camera and IMU should remain the same. First, for the camera measurement of L (see (61)), we have

$$\begin{bmatrix} x \\ y \\ z \end{bmatrix} = {}^{I}\mathbf{p}_L = {}^{I}_{k}\mathbf{C}_{\bar{q}}({}^{R_k}\mathbf{p}_L - {}^{R_k}\mathbf{p}_I)$$
$$= s\,{}^{I}_{k}\mathbf{C}'_{\bar{q}}({}^{R_k}\mathbf{p}'_L - {}^{R_k}\mathbf{p}'_I) = s\,{}^{I}\mathbf{p}'_L = s\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} \Rightarrow \tag{96}$$
$$\mathbf{z}_I = \frac{1}{z}\begin{bmatrix} x \\ y \end{bmatrix} = \frac{1}{sz'}\begin{bmatrix} sx' \\ sy' \end{bmatrix} = \frac{1}{z'}\begin{bmatrix} x' \\ y' \end{bmatrix} = \mathbf{z}'_I$$

where the camera measurement does not change because the scale has no effect on such perspective projection. Then, for the IMU measurements we first examine the angular velocity measured by the gyroscope (see (7)), as

$$\boldsymbol{\omega}_m = \boldsymbol{\omega} + \mathbf{b}_g = \boldsymbol{\omega}' + \mathbf{b}'_g \Rightarrow$$
$$\mathbf{b}_g = \mathbf{b}'_g \tag{97}$$

Similarly, for the linear acceleration measurements from the accelerometer (see (8)), we have

$$\mathbf{a}_m = {}^{I}\mathbf{a} + {}^{I}\mathbf{g} + \mathbf{b}_a = {}^{I}\mathbf{a}' + {}^{I}\mathbf{g}' + \mathbf{b}'_a \Rightarrow$$
$$\mathbf{b}_a = \mathbf{b}'_a - (s-1)\,{}^{I}\mathbf{a}' \tag{98}$$

Note that $\mathbf{b}_a$ cannot be simply represented as an s multiple of $\mathbf{b}'_a$, because it is a random walk process (see (6)). Thus, based on (89), (90), (91), (92), (93), (94), (97), and (98), it is not difficult to validate the corresponding error-state relation as shown in (81).

Appendix D: Proof of Lemma 2

Based on the observability matrix (see (45)), the ℓth block row, $\mathbf{M}'_\ell$, of observability matrix $\mathbf{M}'$ evaluating at ${}^{R_k}\hat{\mathbf{x}}'_\ell$ and ${}^{R_k}\hat{\mathbf{p}}'_L$ has the following structure:

$$\mathbf{M}'_\ell = \boldsymbol{\Pi}'\begin{bmatrix} \mathbf{0}_{3\times 6} & \boldsymbol{\Gamma}'_1 & \boldsymbol{\Gamma}'_2 & -\mathbf{I}_3 & \boldsymbol{\Gamma}'_3 & \boldsymbol{\Gamma}'_4 & \boldsymbol{\Gamma}'_5 & | & \mathbf{I}_3 \end{bmatrix}$$

The scale direction, $\mathbf{u}$, is unobservable (see (81)), if and only if $\mathbf{M}'_\ell\mathbf{u} = \mathbf{0}$, $\forall \ell \geqslant k$; thus, we have

$$\boldsymbol{\Pi}'(-{}^{R_k}\hat{\mathbf{p}}'_{I_\ell} + \boldsymbol{\Gamma}'_3\hat{\mathbf{v}}'_{I_\ell} - \boldsymbol{\Gamma}'^{\,\ell}_5\hat{\mathbf{a}}' + {}^{R_k}\hat{\mathbf{p}}'_L) = \mathbf{0} \tag{99}$$

where

$$\mathbf{\Pi}'({}^{R_k}\hat{\mathbf{p}}'_L - {}^{R_k}\hat{\mathbf{p}}'_{I_\ell}) = \mathbf{H}'_{\mathrm{p}}\,{}^{I_\ell}\hat{\mathbf{p}}'_L = \mathbf{0}$$

because ${}^{I_\ell}\hat{\mathbf{p}}'_L$ is in the right nullspace of $\mathbf{H}'_{\mathrm{p}}$ (see (61)–(62)). Then, what is left to show is

$$\mathbf{\Pi}'(\mathbf{\Gamma}'_3\hat{\mathbf{v}}'_{I_\ell} - \mathbf{\Gamma}'_5{}^{\ell}\hat{\mathbf{a}}') = \mathbf{0}$$

where

$$\mathbf{\Gamma}'_3\hat{\mathbf{v}}'_{I_\ell} - \mathbf{\Gamma}'_5{}^{\ell}\hat{\mathbf{a}}' = -\Delta t_{k,\ell}\hat{\mathbf{v}}'_{I_\ell} - \int_{t_k}^{t_\ell}\int_{t_k}^{\tau} {}^{\mu}_{k}\mathbf{C}^{\top}_{\hat{\bar{q}}}\, d\mu d\tau\, {}^{\ell}\hat{\mathbf{a}}'$$

To this end, we examine two special cases: (i) if no rotation (i.e., $\boldsymbol{\omega} = \mathbf{0}$, $\forall \tau \in [t_k, t_\ell]$), then we have

$$\begin{aligned}\mathbf{\Gamma}'_3\hat{\mathbf{v}}'_{I_\ell} - \mathbf{\Gamma}'_5{}^{\ell}\hat{\mathbf{a}}' &= -\Delta t_{k,\ell}\hat{\mathbf{v}}'_{I_\ell} - \int_{t_k}^{t_\ell}\int_{t_k}^{\tau}\mathbf{I}_3\, d\mu d\tau\, {}^{\ell}\hat{\mathbf{a}}' \\ &= -\Delta t_{k,\ell}\hat{\mathbf{v}}'_{I_\ell} - \frac{1}{2}\Delta t^2_{k,\ell}\,{}^{\ell}\hat{\mathbf{a}}'\end{aligned} \tag{100}$$

and (ii) if constant local acceleration (i.e., ${}^{\tau}\mathbf{a}' \equiv {}^{k}\mathbf{a}'$, $\forall \tau \in [t_k, t_\ell]$), then we have

$$\begin{aligned}\mathbf{\Gamma}'_3\hat{\mathbf{v}}'_{I_\ell} - \mathbf{\Gamma}'_5{}^{\ell}\hat{\mathbf{a}}' &= -\Delta t_{k,\ell}\hat{\mathbf{v}}'_{I_\ell} - \int_{t_k}^{t_\ell}\int_{t_k}^{\tau} {}^{\mu}_{k}\mathbf{C}^{\top}_{\hat{\bar{q}}}\,{}^{\mu}\hat{\mathbf{a}}'\, d\mu d\tau \\ &= -\Delta t_{k,\ell}\hat{\mathbf{v}}'_{I_\ell} - \int_{t_k}^{t_\ell}\int_{t_k}^{\tau} {}^{R_k}\hat{\mathbf{a}}'(\mu)\, d\mu d\tau \\ &= -\Delta t_{k,\ell}\hat{\mathbf{v}}'_{I_\ell} - \int_{t_k}^{t_\ell} ({}^{R_k}\hat{\mathbf{v}}'_{I_\tau} - {}^{R_k}\hat{\mathbf{v}}'_{I_k}) d\tau \\ &= -\Delta t_{k,\ell}\hat{\mathbf{v}}'_{I_\ell} - {}^{R_k}\hat{\mathbf{p}}'_{I_\ell} + \Delta t_{k,\ell}\hat{\mathbf{v}}'_{I_k}\end{aligned} \tag{101}$$

To ensure that (99) holds, both (100) and (101) should be equal to $\mathbf{0}$, and the conclusion of Lemma 2 is immediate.