Consistency Analysis and Improvement of Vision-aided Inertial Navigation

Joel A. Hesch, *Student Member, IEEE*, Dimitrios G. Kottas, *Student Member, IEEE*,
Sean L. Bowman, *Student Member, IEEE*, and Stergios I. Roumeliotis, *Senior Member, IEEE*

Abstract—In this paper, we study estimator inconsistency in vision-aided inertial navigation systems (VINS) from the standpoint of system's observability. We postulate that a leading cause of inconsistency is the gain of spurious information along unobservable directions, which results in smaller uncertainties, larger estimation errors, and divergence. We develop an observability constrained VINS (OC-VINS), which explicitly enforces the unobservable directions of the system, hence preventing spurious information gain and reducing inconsistency. This framework is applicable to several variants of the VINS problem such as visual simultaneous localization and mapping (V-SLAM), as well as visual-inertial odometry using the multi-state constraint Kalman filter (MSC-KF). Our analysis, along with the proposed method to reduce inconsistency, are extensively validated with simulation trials and real-world experimentation.

Index Terms—Consistency, nonlinear estimation, observability analysis, vision-aided inertial navigation.

I. INTRODUCTION

A VISION-AIDED inertial navigation system (VINS) fuses data from a camera and an inertial measurement unit (IMU) to track the six-degrees-of-freedom (d.o.f.) position and orientation (pose) of a sensing platform. This sensor pair is ideal since it combines complementary sensing capabilities [1]. Specifically, an IMU can accurately track dynamic motions over short time durations, while visual data can be used to estimate the pose displacement (up to scale) between consecutive views. Within the robotics community, VINS has gained popularity as a method to address GPS-denied navigation for several reasons. First, contrary to approaches which utilize wheel odometry, VINS uses inertial sensing that can track general 3-D motions of a vehicle. Hence, it is applicable to a variety of platforms such as aerial vehicles, legged robots, and even humans, which are not constrained to move along planar trajectories. Second, unlike laser-scanner-based methods that rely on the existence of

structural planes [2] or height invariance in semistructured environments [3], using vision as an exteroceptive sensor enables VINS methods to work in unstructured areas such as collapsed buildings or outdoors. Furthermore, both cameras and IMUs are light-weight and have low power-consumption requirements, which has lead to recent advances in onboard estimation for payload-constrained platforms such as micro aerial vehicles (e.g., [4]–[7]).

Numerous VINS approaches have been presented in the literature, including methods based on the extended Kalman filter (EKF) [8]–[11], the unscented Kalman filter [12], and batch least squares (BLS) [13]. Nonparametric estimators, such as the particle filter, have also been applied to visual odometry (e.g., [14], [15]). However, these have focused on the simplified problem of estimating the pose of a vehicle whose motion is constrained to 2-D, since the number of particles required is exponential in the size of the state vector. Existing work has addressed a variety of issues in VINS, such as reducing its computational cost [4], [9], dealing with delayed observations [7], improving fault detection by processing the visual and inertial measurements in a loosely coupled manner [5], increasing the accuracy of feature initialization and estimation [16], and improving the robustness to estimator initialization errors [17].

A fundamental issue that has only received limited attention in the literature is how estimator inconsistency affects VINS. As defined in [18], a state estimator is consistent if the estimation errors are zero-mean and have covariance equal to the one calculated by the filter. Estimator inconsistency can have a devastating effect, particularly in navigation applications, since both the current pose estimate and its uncertainty must be accurate in order to address tasks that depend on the localization solution, such as path planning. For nonlinear systems, several potential sources of inconsistency exist (e.g., motion-model mismatch in target tracking), and great care must be taken when designing an estimator to improve consistency.

In this paper, we report on VINS inconsistency.[1] We focus specifically on estimator inconsistency due to spurious information gain which arises from approximations incurred when applying linear estimation tools to nonlinear problems (i.e., when using linearized estimators such as the EKF). In summary, the main contributions of this work are as follows.

[1]A poster describing the main results of this study appeared in [19]. Moreover, a short version of this paper detailing the OC-VINS framework applied to V-SLAM appeared in [20], while the application to the MSC-KF appeared in [21]. In comparison with [20] and [21], in this paper, we provide an extended theoretic analysis of the problem, including an observability analysis of the linearized VINS system model, and present extensive simulation and experimental results that validate our approach.

1) We analyze the structure of the true and estimated systems and show that for the true system four unobservable directions exist (i.e., three-d.o.f. global translation and one-d.o.f. rotation about the gravity vector), while the system employed for estimation purposes has only three unobservable directions (three-d.o.f. global translation). Moreover, we postulate that a main source of inconsistency in VINS is spurious information gained when orientation information is incorrectly projected along the direction corresponding to rotations about the gravity vector.

2) We propose a simple, yet powerful, estimator modification that explicitly prohibits this incorrect information gain. Our approach is general enough to be applied in multiple VINS domains (e.g., V-SLAM and the MSC-KF [22]) when linearized estimators, such as the EKF, are used.

3) We provide extensive evidence to demonstrate the inconsistency in standard VINS approaches as well as validate our method with Monte-Carlo simulations to show that it improves consistency and reduces estimation errors as compared with standard VINS. In addition, we demonstrate the performance of our approach experimentally using a miniature IMU and a small-size camera.

The rest of this paper is organized as follows: We begin with an overview of the related work in Section II. In Section III, we describe the system and measurement models, followed by our analysis of VINS inconsistency in Section IV. The proposed estimator modification is presented in Section V and, subsequently, validated both in simulations and experimentally in Sections VI and VII. Finally, we provide our concluding remarks and outline our future research directions in Section VIII.

II. RELATED WORK

Until recently, little attention was paid to the effect that the observability properties of a system can have on the consistency of a linearized estimator that is employed to solve a nonlinear estimation problem. The work by Huang *et al.* [23]–[25] was the first to identify this connection for several 2-D localization problems [i.e., simultaneous localization and mapping (SLAM), and cooperative localization (CL)]. The authors showed that, for these problems, a mismatch exists between the number of unobservable directions of the true nonlinear system and the linearized system used for estimation purposes. In particular, the estimated (linearized) system has one fewer unobservable direction than the true system, which allows the estimator to surreptitiously gain spurious information along the direction corresponding to global orientation (yaw). This increases the estimation errors, while erroneously reducing the estimator uncertainty, and leads to inconsistency.

No similar analysis existed that link the VINS observability properties to the estimator inconsistency until recently [19]–[21], [26], despite the fact that several authors have studied the observability properties of VINS for various scenarios. For the task of IMU-camera extrinsic calibration, Mirzaei and Roumeliotis [27], as well as, Kelly and Sukhatme [28], have analyzed the system observability using Lie derivatives [29] to determine when the IMU-camera transformation is observable.

Jones and Soatto [16] studied VINS observability by examining the indistinguishable trajectories of the system [30] under different sensor configurations (i.e., inertial only, vision only, vision and inertial), while Martinelli [31] utilized the concept of continuous symmetries to show that the IMU biases, 3-D velocity, and absolute roll and pitch angles are observable for VINS. Li and Mourikis [26] recently presented an observability analysis of a bias-free linearized VINS model and leveraged the first-estimates Jacobian methodology of [23] to reduce the impact of inconsistency in visual-inertial odometry. In contrast, our observability analysis examines the linearized system comprising the full VINS state (i.e., including IMU biases). Furthermore, our approach to reduce the inconsistency is more general since any linearization method can be employed by the estimator (e.g., computing Jacobians analytically, numerically, or using sample points).

In this paper, we study the observability properties of the ideal linearized VINS model (i.e., the one whose Jacobians are evaluated at the true states) and prove that it has four unobservable d.o.f., corresponding to three-d.o.f. global translations and one-d.o.f. global rotation about the gravity vector. Moreover, we show that when the estimated states are used to evaluate the Jacobians, as is the case for the EKF, the number of unobservable directions is reduced by one. In particular, the global rotation about the gravity vector becomes (erroneously) observable, which allows the estimator to gain spurious information and leads to inconsistency. These results confirm the findings of [16] and [31] using a different approach (i.e., the observability matrix), while additionally specifying the exact mathematical structure of the unobservable directions necessary to assess the EKF's inconsistency.[2] Finally, in order to improve inconsistency, we introduce a modification of the VINS EKF where its estimated Jacobians are updated to ensure that the number of unobservable directions is the same as when using the true Jacobians. In this manner, the global rotation about the gravity vector remains unobservable (as it should), and the consistency of the VINS EKF is significantly improved.

III. VINS ESTIMATOR DESCRIPTION

We begin with an overview of the propagation and measurement models which govern the VINS. In particular, we employ an EKF to fuse the camera and IMU measurements to estimate the state of the system including the pose, velocity, and IMU biases, as well as the 3-D positions of visual landmarks observed by the camera. We operate in a previously unknown environment and utilize two types of visual features in our VINS framework. The first are opportunistic features (OFs) that can be accurately and efficiently tracked across short image sequences (e.g., using KLT [32]) but are not visually distinctive enough to be efficiently recognized when revisiting an area. OFs can be efficiently used to estimate the motion of the camera over short time horizons (i.e., using the MSC-KF), but they are not included in the state vector. The second are distinguishable features (DFs), which

---

[2]The analysis in [31] addresses the special case with one known feature at the origin, which, in contrast with our approach, cannot be used to determine the nullspace directions when multiple features are considered.
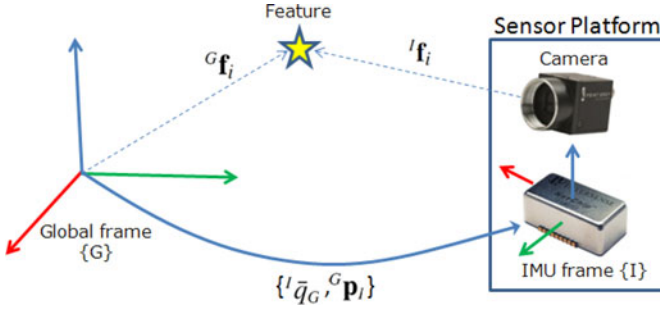
Fig. 1. Sensor platform comprising an IMU and a camera. $\{^I\bar{q}_G, {}^G\mathbf{p}_I\}$ is the quaternion of orientation and position vector pair describing the pose of the sensing frame $\{I\}$ with respect to the global frame $\{G\}$. The ith feature's 3-D coordinates are denoted as ${}^G\mathbf{f}_i$ and ${}^I\mathbf{f}_i$, with respect to $\{G\}$ and $\{I\}$, respectively.

are typically significantly fewer in number, and can be reliably redetected when revisiting an area (e.g., scale-invariant feature transform (SIFT) keys [33]). The 3-D coordinates of the DFs are estimated to construct a map of the area.

A. System State and Propagation Model

The EKF estimates the 3-D IMU pose and linear velocity together with the time-varying IMU biases and a map of visual features (see Fig. 1). In particular, the filter state is the $(16 + 3N) \times 1$ vector.

$$\mathbf{x} = \begin{bmatrix} {}^I\bar{q}_G^T & \mathbf{b}_g^T & {}^G\mathbf{v}_I^T & \mathbf{b}_a^T & {}^G\mathbf{p}_I^T & | & {}^G\mathbf{f}_1^T & \cdots & {}^G\mathbf{f}_N^T \end{bmatrix}^T$$
$$= \begin{bmatrix} \mathbf{x}_s^T & | & \mathbf{x}_f^T \end{bmatrix}^T \tag{1}$$

where $\mathbf{x}_s(t)$ is the 16×1 sensor-platform state, and $\mathbf{x}_f(t)$ is the $3N \times 1$ state of the feature map. The first component of the sensor-platform state is ${}^I\bar{q}_G(t)$ which is the unit quaternion that represents the orientation of the *global frame* $\{G\}$ in the IMU frame $\{I\}$ at time t. The frame $\{I\}$ is attached to the IMU, while $\{G\}$ is a local-vertical reference frame whose origin coincides with the initial IMU position. The sensor-platform state also includes the position and velocity of $\{I\}$ in $\{G\}$, which are denoted by the 3×1 vectors ${}^G\mathbf{p}_I(t)$ and ${}^G\mathbf{v}_I(t)$, respectively. The remaining components are the biases, $\mathbf{b}_g(t)$ and $\mathbf{b}_a(t)$, which affect the gyroscope and accelerometer measurements, which are modeled as random-walk processes driven by the zero-mean, white Gaussian noise $\mathbf{n}_{wg}(t)$ and $\mathbf{n}_{wa}(t)$, respectively.

The map state, $\mathbf{x}_f$, comprises the 3-D coordinates of N DFs, ${}^G\mathbf{f}_i, i = 1, \ldots, N$, and grows as new DFs are observed [34]. In contrast, we do not store OFs in the map. Instead, all OFs are processed and marginalized on-the-fly using the MSC-KF approach [22] (see Section III-B). With the state of the system now defined, we turn our attention to the continuous-time model which governs the state of the system.

1) Continuous-Time Model: The system model that describes the time evolution of the state is (see [35] and [36])

$$^I\dot{\bar{q}}_G(t) = \frac{1}{2}\mathbf{\Omega}(\boldsymbol{\omega}(t)){}^I\bar{q}_G(t) \tag{2}$$

$$\dot{\mathbf{b}}_g(t) = \mathbf{n}_{wg}(t) \tag{3}$$

$$^G\dot{\mathbf{v}}_I(t) = {}^G\mathbf{a}(t) \tag{4}$$

$$\dot{\mathbf{b}}_a(t) = \mathbf{n}_{wa}(t) \tag{5}$$

$$^G\dot{\mathbf{p}}_I(t) = {}^G\mathbf{v}_I(t) \tag{6}$$

$$^G\dot{\mathbf{f}}_i(t) = \mathbf{0}_{3\times1}, \quad i = 1, \ldots, N. \tag{7}$$

In these expressions, $\boldsymbol{\omega}(t) = [\omega_1(t) \ \omega_2(t) \ \omega_3(t)]^T$ is the rotational velocity of the IMU, expressed in $\{I\}$, ${}^G\mathbf{a}$ is the IMU acceleration expressed in $\{G\}$, and

$$\mathbf{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} -\lfloor\boldsymbol{\omega}\times\rfloor & \boldsymbol{\omega} \\ -\boldsymbol{\omega}^T & 0 \end{bmatrix}, \quad \lfloor\boldsymbol{\omega}\times\rfloor \triangleq \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}.$$

The gyroscope and accelerometer measurements, $\boldsymbol{\omega}_m$ and $\mathbf{a}_m$, are modeled as

$$\boldsymbol{\omega}_m(t) = \boldsymbol{\omega}(t) + \mathbf{b}_g(t) + \mathbf{n}_g(t) \tag{8}$$

$$\mathbf{a}_m(t) = \mathbf{C}({}^I\bar{q}_G(t))\left({}^G\mathbf{a}(t) - {}^G\mathbf{g}\right) + \mathbf{b}_a(t) + \mathbf{n}_a(t), \tag{9}$$

where $\mathbf{n}_g$ and $\mathbf{n}_a$ are zero-mean, white Gaussian noise processes, and ${}^G\mathbf{g}$ is the gravitational acceleration. The matrix $\mathbf{C}(\bar{q})$ is the rotation matrix corresponding to $\bar{q}$. The DFs belong to the static scene; thus, their time derivatives are zero [see (7)].

Linearizing at the current estimates and applying the expectation operator on both sides of (2)–(7), we obtain the state estimate propagation model

$$^I\dot{\hat{\bar{q}}}_G(t) = \frac{1}{2}\mathbf{\Omega}(\hat{\boldsymbol{\omega}}(t)){}^I\hat{\bar{q}}_G(t) \tag{10}$$

$$\dot{\hat{\mathbf{b}}}_g(t) = \mathbf{0}_{3\times1} \tag{11}$$

$$^G\dot{\hat{\mathbf{v}}}_I(t) = \mathbf{C}^T({}^I\hat{\bar{q}}_G(t))\hat{\mathbf{a}}(t) + {}^G\mathbf{g} \tag{12}$$

$$\dot{\hat{\mathbf{b}}}_a(t) = \mathbf{0}_{3\times1} \tag{13}$$

$$^G\dot{\hat{\mathbf{p}}}_I(t) = {}^G\hat{\mathbf{v}}_I(t) \tag{14}$$

$$^G\dot{\hat{\mathbf{f}}}_i(t) = \mathbf{0}_{3\times1}, \quad i = 1, \ldots, N \tag{15}$$

where $\hat{\mathbf{a}}(t) = \mathbf{a}_m(t) - \hat{\mathbf{b}}_a(t)$, and $\hat{\boldsymbol{\omega}}(t) = \boldsymbol{\omega}_m(t) - \hat{\mathbf{b}}_g(t)$.

The $(15 + 3N) \times 1$ error-state vector is defined as

$$\widetilde{\mathbf{x}} = \begin{bmatrix} {}^I\delta\boldsymbol{\theta}_G^T & \widetilde{\mathbf{b}}_g^T & {}^G\widetilde{\mathbf{v}}_I^T & \widetilde{\mathbf{b}}_a^T & {}^G\widetilde{\mathbf{p}}_I^T & | & {}^G\widetilde{\mathbf{f}}_1^T & \cdots & {}^G\widetilde{\mathbf{f}}_N^T \end{bmatrix}^T$$
$$= \begin{bmatrix} \widetilde{\mathbf{x}}_s^T & | & \widetilde{\mathbf{x}}_f^T \end{bmatrix}^T \tag{16}$$

where $\widetilde{\mathbf{x}}_s(t)$ is the 15×1 error state corresponding to the sensing platform, and $\widetilde{\mathbf{x}}_f(t)$ is the $3N \times 1$ error state of the map. For the IMU position, velocity, biases, and the map, an additive error model is utilized (i.e., $\widetilde{x} = x - \hat{x}$ is the error in the estimate $\hat{x}$ of a quantity x). However, for the quaternion we employ a multiplicative error model [36]. Specifically, the error between the quaternion $\bar{q}$ and its estimate $\hat{\bar{q}}$ is the 3×1 angle-error vector,

$\delta\boldsymbol{\theta}$, implicitly defined by the error quaternion

$$\delta\bar{q} = \bar{q} \otimes \hat{\bar{q}}^{-1} \simeq \begin{bmatrix} \frac{1}{2}\delta\boldsymbol{\theta}^T & 1 \end{bmatrix}^T \quad (17)$$

where $\delta\bar{q}$ describes the small rotation that causes the true and estimated attitudes to coincide. This allows us to represent the attitude uncertainty by the 3×3 covariance matrix $\mathbb{E}[\delta\boldsymbol{\theta}\delta\boldsymbol{\theta}^T]$, which is a minimal representation.

The linearized continuous-time error-state equation is

$$\dot{\widetilde{\mathbf{x}}} = \begin{bmatrix} \mathbf{F}_s & \mathbf{0}_{15 \times 3N} \\ \mathbf{0}_{3N \times 15} & \mathbf{0}_{3N} \end{bmatrix} \widetilde{\mathbf{x}} + \begin{bmatrix} \mathbf{G}_s \\ \mathbf{0}_{3N \times 12} \end{bmatrix} \mathbf{n}$$
$$= \mathbf{F}\widetilde{\mathbf{x}} + \mathbf{G}\mathbf{n} \quad (18)$$

where $\mathbf{0}_{3N}$ denotes the $3N \times 3N$ matrix of zeros. Here, $\mathbf{n}$ is the vector comprising the IMU measurement noise terms, as well as the process noise driving the IMU biases, i.e.,

$$\mathbf{n} = \begin{bmatrix} \mathbf{n}_g^T & \mathbf{n}_{wg}^T & \mathbf{n}_a^T & \mathbf{n}_{wa}^T \end{bmatrix}^T \quad (19)$$

while $\mathbf{F}_s$ is the continuous-time error-state transition matrix corresponding to the sensor-platform state, and $\mathbf{G}_s$ is the continuous-time input noise matrix, i.e.,

$$\mathbf{F}_s = \begin{bmatrix} -\lfloor\hat{\boldsymbol{\omega}}\times\rfloor & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ -\mathbf{C}^T(^I\hat{\bar{q}}_G)\lfloor\hat{\mathbf{a}}\times\rfloor & \mathbf{0}_3 & \mathbf{0}_3 & -\mathbf{C}^T(^I\hat{\bar{q}}_G) & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \quad (20)$$

$$\mathbf{G}_s = \begin{bmatrix} -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & -\mathbf{C}^T(^I\hat{\bar{q}}_G) & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \quad (21)$$

where $\mathbf{I}_3$ is the 3×3 identity matrix. The system noise is modeled as a zero-mean white Gaussian process with autocorrelation $\mathbb{E}[\mathbf{n}(t)\mathbf{n}^T(\tau)] = \mathbf{Q}_c\delta(t - \tau)$, where $\mathbf{Q}_c$ depends on the IMU noise characteristics and is computed off-line [36].

2) Discrete-Time Implementation: The IMU signals $\boldsymbol{\omega}_m$ and $\mathbf{a}_m$ are sampled at a constant rate $1/\delta t$, where $\delta t = t_{k+1} - t_k$. Every time a new IMU measurement is received, the state estimate is propagated using numerical integration of (10)–(15). In order to derive the covariance propagation equation, we compute the discrete-time state transition matrix, $\boldsymbol{\Phi}_{k+1,k}$, from time-step t_k to t_{k+1}, as the solution to the following matrix differential equation:

$$\dot{\boldsymbol{\Phi}}_{k+1,k} = \mathbf{F}\boldsymbol{\Phi}_{k+1,k} \quad (22)$$
$$\text{initial condition} \quad \boldsymbol{\Phi}_{k,k} = \mathbf{I}_{18} \quad (23)$$

which can be calculated analytically as we show in [34], or numerically using Runge–Kutta. We also compute the discrete-time system noise covariance matrix $\mathbf{Q}_k$ as

$$\mathbf{Q}_k = \int_{t_k}^{t_{k+1}} \boldsymbol{\Phi}_{k+1,\tau}\mathbf{G}\mathbf{Q}_c\mathbf{G}^T\boldsymbol{\Phi}_{k+1,\tau}^T \, d\tau \quad (24)$$

and finally, we perform the covariance propagation as

$$\mathbf{P}_{k+1|k} = \boldsymbol{\Phi}_{k+1,k}\mathbf{P}_{k|k}\boldsymbol{\Phi}_{k+1,k}^T + \mathbf{Q}_k. \quad (25)$$

We note that in the aforementioned expression, and throughout the paper, $\mathbf{P}_{i|j}$ and $\hat{\mathbf{x}}_{i|j}$ denote the estimates of the error-state covariance and state, respectively, at time-step i computed using measurements up to time-step j.

B. Measurement Update Model

As the camera-IMU platform moves, the camera observes both opportunistic and distinguishable visual features. These measurements are utilized to concurrently estimate the motion of the sensing platform and the map of DFs. We perform three types of filter updates: 1) DF updates of features already in the map, 2) initialization of DFs not yet in the map, and 3) OF updates. We first describe the feature measurement model, and subsequently, detail how it is employed in each case.

To simplify the discussion, we consider the observation of a single DF point $\mathbf{f}_i$. The camera measures $\mathbf{z}_i$, which is the perspective projection of the 3-D point $^I\mathbf{f}_i$, expressed in the current IMU frame $\{I\}$, onto the image plane,[3] i.e.,

$$\mathbf{z}_i = \frac{1}{z}\begin{bmatrix} x \\ y \end{bmatrix} + \boldsymbol{\eta}_i \quad (26)$$

$$\text{where} \quad \begin{bmatrix} x \\ y \\ z \end{bmatrix} = {}^I\mathbf{f}_i = \mathbf{C}(^I\bar{q}_G)(^G\mathbf{f}_i - {}^G\mathbf{p}_I). \quad (27)$$

The measurement noise, $\boldsymbol{\eta}_i$, is modeled as zero mean, white Gaussian with covariance $\mathbf{R}_i$. The linearized error model is

$$\tilde{\mathbf{z}}_i = \mathbf{z}_i - \hat{\mathbf{z}}_i \simeq \mathbf{H}_i\tilde{\mathbf{x}} + \boldsymbol{\eta}_i \quad (28)$$

where $\hat{\mathbf{z}}$ is the expected measurement computed by evaluating (26)–(27) at the current state estimate, and the measurement Jacobian $\mathbf{H}_i$ is

$$\mathbf{H}_i = \mathbf{H}_c[\mathbf{H}_{\bar{q}} \quad \mathbf{0}_{3 \times 9} \quad \mathbf{H}_\mathbf{p}|\mathbf{0}_3 \cdots \mathbf{H}_{\mathbf{f}_i} \cdots \mathbf{0}_3] \quad (29)$$

with

$$\mathbf{H}_c = \frac{1}{z^2}\begin{bmatrix} z & 0 & -x \\ 0 & z & -y \end{bmatrix} \quad (30)$$

$$\mathbf{H}_{\bar{q}} = \lfloor\mathbf{C}(^I\bar{q}_G)(^G\mathbf{f}_i - {}^G\mathbf{p}_I)\times\rfloor \quad (31)$$

$$\mathbf{H}_\mathbf{p} = -\mathbf{C}(^I\bar{q}_G) \quad (32)$$

$$\mathbf{H}_{\mathbf{f}_i} = \mathbf{C}(^I\bar{q}_G) \quad (33)$$

evaluated at the current state estimate. Here, $\mathbf{H}_c$, is the Jacobian of the camera's perspective projection with respect to $^I\mathbf{f}_i$, while $\mathbf{H}_{\bar{q}}$, $\mathbf{H}_\mathbf{p}$, and $\mathbf{H}_{\mathbf{f}_i}$ are the Jacobians of $^I\mathbf{f}_i$ with respect to $^I\bar{q}_G$, $^G\mathbf{p}_I$, and $^G\mathbf{f}_i$.

---

[3]Without loss of generality, we express the image measurement in normalized pixel coordinates, and consider the camera frame to be coincident with the IMU. In practice, we perform both intrinsic [37] and extrinsic [27] IMU-camera calibration off-line.

This measurement model is utilized in each of the three update methods. For DFs that are already in the map, we directly apply the measurement model (26)–(29) to update the filter. In particular, we compute the measurement residual $\mathbf{r}_i$, along with its covariance $\mathbf{S}_i$, and the Kalman gain $\mathbf{K}_i$, i.e.,

$$\mathbf{r}_i = \mathbf{z}_i - \hat{\mathbf{z}}_i \tag{34}$$

$$\mathbf{S}_i = \mathbf{H}_i \mathbf{P}_{k+1|k} \mathbf{H}_i^T + \mathbf{R}_i \tag{35}$$

$$\mathbf{K}_i = \mathbf{P}_{k+1|k} \mathbf{H}_i^T \mathbf{S}_i^{-1} \tag{36}$$

and update the EKF state and covariance as

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}_i \mathbf{r}_i \tag{37}$$

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{K}_i \mathbf{S}_i \mathbf{K}_i^T. \tag{38}$$

Each time a new (previously unobserved) DF is measured, we initialize it into the state vector. Since a single visual observation does not provide enough information to resolve all three-d.o.f. of a DF's position, we utilize multiple observations to initialize each feature. In order to compute the initial landmark position estimate, uncertainty, and cross-correlation with the current state, we solve a bundle-adjustment problem over a short time window [38], which is described in detail in Appendix A.

In contrast, OFs are not estimated as part of the state. Instead, we employ the MSC-KF to impose an efficient (linear complexity) update, which constrains all of the camera poses from which the feature was observed. For a detailed description of the MSC-KF algorithm, see [11] and [22].

IV. VINS OBSERVABILITY ANALYSIS

In this section, we examine the observability properties of the linearized VINS model in the general case when a single point feature is observed by a sensor platform performing arbitrary motion. Specifically, we first study and analytically determine the four unobservable directions of the *ideal* linearized VINS model (i.e., the system whose Jacobians are evaluated at the true states). Subsequently, we show that the linearized VINS model used by the EKF, whose Jacobians are evaluated using the current state estimates, has only three unobservable directions (i.e., the ones corresponding to global translation), while the one corresponding to global rotation about the gravity vector becomes (erroneously) observable. The key findings of this analysis are then employed in Section V to improve the consistency of the EKF-based VINS.

The observability matrix $\mathbf{M}$ is defined as [39]

$$\mathbf{M}(\mathbf{x}^\star) = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2 \boldsymbol{\Phi}_{2,1} \\ \vdots \\ \mathbf{H}_k \boldsymbol{\Phi}_{k,1} \end{bmatrix} \tag{39}$$

where $\boldsymbol{\Phi}_{k,1} = \boldsymbol{\Phi}_{k,k-1} \dots \boldsymbol{\Phi}_{2,1}$ is the state transition matrix from time-step 1 to k, and $\mathbf{H}_k$, is the measurement Jacobian [see (29)], for the feature observation at time-step k. Note that, since all the Jacobians are evaluated at a particular state $\mathbf{x}^\star = \begin{bmatrix} \mathbf{x}_1^{\star T} & \cdots & \mathbf{x}_k^{\star T} \end{bmatrix}^T$, the observability matrix is also a

function of $\mathbf{x}^\star$. If $\mathbf{M}(\mathbf{x}^\star)$ was full column rank, then the linearized VINS model would be observable. However, as we will show in the following analysis, $\mathbf{M}(\mathbf{x}^\star)$ is rank deficient, and hence, the VINS model is unobservable. More importantly, the number of unobservable directions (right nullspace dimension) differs, depending on the selection of the linearization point.

A. Observability Analysis of the Ideal Linearized VINS Model

In the ideal linearized VINS model, the system and measurement Jacobians are evaluated at the true state (i.e., $\mathbf{x}^\star = \mathbf{x}$). Based on this definition, the first block-row of $\mathbf{M}(\mathbf{x})$ can be written as follows [see (29)]:[4]

$$\mathbf{H}_k = \mathbf{H}_{c,k} \mathbf{C} \left({}^{I_k} \bar{q}_G \right)$$
$$\cdot \begin{bmatrix} \lfloor {}^G \mathbf{f} - {}^G \mathbf{p}_{I_k} \times \rfloor \mathbf{C} \left({}^{I_k} \bar{q}_G \right)^T & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & -\mathbf{I}_3 & \mathbf{I}_3 \end{bmatrix} \tag{40}$$

where ${}^{I_k} \bar{q}_G$ denotes the rotation of $\{G\}$ with respect to frame $\{I_k\}$.

To compute the remaining block rows of the observability matrix, we require $\boldsymbol{\Phi}_{k,1}$, which satisfies the following matrix differential equation [40]:

$$\dot{\boldsymbol{\Phi}}_{k,1} = \mathbf{F} \boldsymbol{\Phi}_{k,1} \tag{41}$$

$$\text{initial condition } \boldsymbol{\Phi}_{1,1} = \mathbf{I}_{18} \tag{42}$$

where $\mathbf{F}$ is defined in (18).

By examining the block elements of (41), we can obtain an analytical solution for the ones necessary for our observability analysis. In particular, the $(2, 1)$ element of $\dot{\boldsymbol{\Phi}}_{k,1}$ is the product of the second block row[5] of $\mathbf{F}$ [i.e., $\mathbf{F}^{(2,:)} = \mathbf{0}_{3 \times 18}$; see (18)] and the first block column of $\boldsymbol{\Phi}_{k,1}$ [see (41)]. Hence, $\dot{\boldsymbol{\Phi}}_{k,1}^{(2,1)} = \mathbf{F}^{(2,:)} \cdot \boldsymbol{\Phi}_{k,1}^{(:,1)} = \mathbf{0}_3$, and recalling the initial condition $\boldsymbol{\Phi}_{1,1}^{(2,1)} = \mathbf{0}_3$ [see (42)], we obtain

$$\boldsymbol{\Phi}_{k,1}^{(2,1)} = \mathbf{0}_3. \tag{43}$$

Following a similar approach, we can easily determine all block elements of $\boldsymbol{\Phi}_{k,1}$ that are either $\mathbf{0}_3$ or $\mathbf{I}_3$, respectively (see [34] for details). Specifically, $\boldsymbol{\Phi}_{k,1}$ has the following structure:

$$\boldsymbol{\Phi}_{k,1} = \begin{bmatrix} \boldsymbol{\Phi}_{k,1}^{(1,1)} & \boldsymbol{\Phi}_{k,1}^{(1,2)} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}_{k,1}^{(3,1)} & \boldsymbol{\Phi}_{k,1}^{(3,2)} & \mathbf{I}_3 & \boldsymbol{\Phi}_{k,1}^{(3,4)} & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}_{k,1}^{(5,1)} & \boldsymbol{\Phi}_{k,1}^{(5,2)} & \delta t_k \mathbf{I}_3 & \boldsymbol{\Phi}_{k,1}^{(5,4)} & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{44}$$

where $\delta t_k = \delta t(k-1)$, is the time difference between time-steps 1 and k.

---

[4]We hereafter focus on the case of a single point feature, i.e., $i = 1$, for the purpose of simplifying the presentation. Extending this analysis to the case of multiple features is straightforward.

[5]The superscript notations $\mathbf{E}^{(i,:)}$ and $\mathbf{E}^{(:,i)}$ refer to the ith block row and block column of matrix $\mathbf{E}$, respectively, while $\mathbf{E}^{(i,j)}$ references the block element (i,j).

For the purposes of the observability analysis, we only require a selected number of the remaining block elements, while the expressions for all of them are provided in [34]. We begin by computing $\Phi_{k,1}^{(1,1)}$. Proceeding from (41)

$$\dot{\Phi}_{k,1}^{(1,1)} = \mathbf{F}^{(1,:)}\Phi_{k,1}^{(:,1)}$$

$$= \begin{bmatrix} -\lfloor^{I_k}\boldsymbol{\omega}\times\rfloor & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \begin{bmatrix} \Phi_{k,1}^{(1,1)} \\ \mathbf{0}_3 \\ \Phi_{k,1}^{(3,1)} \\ \mathbf{0}_3 \\ \Phi_{k,1}^{(5,1)} \\ \mathbf{0}_3 \end{bmatrix}$$

$$= -\lfloor^{I_k}\boldsymbol{\omega}\times\rfloor\Phi_{k,1}^{(1,1)} \tag{45}$$

with the initial condition $\Phi_{1,1}^{(1,1)} = \mathbf{I}_3$. Thus, the solution for $\Phi_{k,1}^{(1,1)}$ is computed as

$$\Phi_{k,1}^{(1,1)} = \Phi_{1,1}^{(1,1)} \exp\left(\int_{t_1}^{t_k} -\lfloor^{I_\tau}\boldsymbol{\omega}\times\rfloor d\tau\right)$$

$$= \exp\left(-\int_{t_1}^{t_k}\lfloor^{I_\tau}\boldsymbol{\omega}\times\rfloor d\tau\right) = \mathbf{C}(^{I_k}\bar{q}_{I_1}). \tag{46}$$

We follow an analogous approach to compute the other elements pertinent to the observability study (see Appendix B), i.e.,

$$\Phi_{k,1}^{(1,2)} = -\int_{t_1}^{t_k}\mathbf{C}\left(^{I_k}\bar{q}_{I_\tau}\right) d\tau \tag{47}$$

$$\Phi_{k,1}^{(5,1)} = \lfloor^G\mathbf{p}_{I_1} + {}^G\mathbf{v}_{I_1}\delta t_k - \frac{1}{2}{}^G\mathbf{g}\delta t_k^2 - {}^G\mathbf{p}_{I_k}\times\rfloor\mathbf{C}(^G\bar{q}_{I_1}) \tag{48}$$

$$\Phi_{k,1}^{(5,2)} = \int_{t_1}^{t_k}\int_{t_1}^{\theta}\mathbf{C}(^G\bar{q}_{I_s})\lfloor^{I_s}\mathbf{a}\times\rfloor\int_{t_1}^{s}\mathbf{C}(^{I_s}\bar{q}_{I_\tau}) d\tau\, ds\, d\theta \tag{49}$$

$$\Phi_{k,1}^{(5,4)} = -\int_{t_1}^{t_k}\int_{t_1}^{s}\mathbf{C}(^G\bar{q}_{I_\tau}) d\tau\, ds. \tag{50}$$

Using these expressions, we can obtain the kth block row, $\mathbf{M}_k$, of $\mathbf{M}$, for any $k > 1$, as follows [see (39), (40), (44), and (46)–(50)]:

$$\mathbf{M}_k = \mathbf{H}_k\Phi_{k,1}$$
$$= \Gamma_1[\Gamma_2 \quad \Gamma_3 \quad -\delta t_k\mathbf{I}_3 \quad \Gamma_4 \quad -\mathbf{I}_3 \quad \mathbf{I}_3] \tag{51}$$

where

$$\Gamma_1 = \mathbf{H}_{c,k}\mathbf{C}\left(^{I_k}\bar{q}_G\right) \tag{52}$$

$$\Gamma_2 = \lfloor^G\mathbf{f} - {}^G\mathbf{p}_{I_1} - {}^G\mathbf{v}_{I_1}\delta t_k + \frac{1}{2}{}^G\mathbf{g}\delta t_k^2\times\rfloor\mathbf{C}^T\left(^{I_1}\bar{q}_G\right) \tag{53}$$

$$\Gamma_3 = \lfloor^G\mathbf{f} - {}^G\mathbf{p}_{I_k}\times\rfloor\mathbf{C}^T\left(^{I_k}\bar{q}_G\right)\Phi_{k,1}^{(1,2)} - \Phi_{k,1}^{(5,2)} \tag{54}$$

$$\Gamma_4 = -\Phi_{k,1}^{(5,4)}. \tag{55}$$

We note that for generic motions (i.e., $\boldsymbol{\omega} \neq \mathbf{0}_{3\times1}, \mathbf{a} \neq \mathbf{0}_{3\times1}$) $\Phi_{k,1}^{(1,2)}, \Phi_{k,1}^{(5,2)}, \Phi_{k,1}^{(5,4)}$, and thus, Γ_3 and Γ_4 are time varying matrices, whose columns are linearly independent. The structure

of the remaining block elements of $\mathbf{M}_k$ [see (51)] is employed to form a basis of the nullspace of $\mathbf{M}$ analytically.

At this point, we state the main result of our analysis:

Theorem 1: The right nullspace $\mathbf{N}_1$ of the observability matrix $\mathbf{M}(\mathbf{x}^\star)$ [see (39)] of the linearized VINS model

$$\mathbf{M}(\mathbf{x}^\star)\mathbf{N}_1 = \mathbf{0} \tag{56}$$

is spanned by the following four directions:

$$\mathbf{N}_1 = \begin{bmatrix} \mathbf{0}_3 & \mathbf{C}\left(^{I_1}\bar{q}_G\right){}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_{3\times1} \\ \mathbf{0}_3 & -\lfloor^G\mathbf{v}_{I_1}\times\rfloor^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_{3\times1} \\ \mathbf{I}_3 & -\lfloor^G\mathbf{p}_{I_1}\times\rfloor^G\mathbf{g} \\ \mathbf{I}_3 & -\lfloor^G\mathbf{f}\times\rfloor^G\mathbf{g} \end{bmatrix} = \begin{bmatrix} \mathbf{N}_{t,1} & | & \mathbf{N}_{r,1} \end{bmatrix}. \tag{57}$$

Proof: The fact that $\mathbf{N}_1$ is indeed the right nullspace of $\mathbf{M}(\mathbf{x}^\star)$ can be verified by multiplying each of its block rows [see (51)] with $\mathbf{N}_{t,1}$ and $\mathbf{N}_{r,1}$ in (57). Specifically

$$\mathbf{M}_k\mathbf{N}_{t,1} = \Gamma_1[\Gamma_2 \quad \Gamma_3 \quad -\delta t_k\mathbf{I}_3 \quad \Gamma_4 \quad -\mathbf{I}_3 \quad \mathbf{I}_3] \begin{bmatrix} \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{I}_3 \\ \mathbf{I}_3 \end{bmatrix}$$

$$= \Gamma_1(-\mathbf{I}_3 + \mathbf{I}_3) = \mathbf{0}_{2\times3} \tag{58}$$

while

$$\mathbf{M}_k\mathbf{N}_{r,1} = \Gamma_1[\Gamma_2 \quad \Gamma_3 \quad -\delta t_k\mathbf{I}_3 \quad \Gamma_4 \quad -\mathbf{I}_3 \quad \mathbf{I}_3]$$

$$\times \begin{bmatrix} \mathbf{C}\left(^{I_1}\bar{q}_G\right){}^G\mathbf{g} \\ \mathbf{0}_{3\times1} \\ -\lfloor^G\mathbf{v}_{I_1}\times\rfloor^G\mathbf{g} \\ \mathbf{0}_{3\times1} \\ -\lfloor^G\mathbf{p}_{I_1}\times\rfloor^G\mathbf{g} \\ -\lfloor^G\mathbf{f}\times\rfloor^G\mathbf{g} \end{bmatrix}$$

$$= \Gamma_1(\lfloor^G\mathbf{f} - {}^G\mathbf{p}_{I_1} - {}^G\mathbf{v}_{I_1}\delta t_k + \frac{1}{2}{}^G\mathbf{g}\delta t_k^2\times\rfloor^G\mathbf{g}$$
$$+ \lfloor^G\mathbf{v}_{I_1}\delta t_k + {}^G\mathbf{p}_{I_1} - {}^G\mathbf{f}\times\rfloor^G\mathbf{g})$$

$$= \Gamma_1\left(\lfloor^G\mathbf{g}\times\rfloor^G\mathbf{g}\right)\frac{1}{2}\delta t_k^2 = \mathbf{0}_{2\times1}.$$

Since $\mathbf{M}_k\mathbf{N}_{t,1} = \mathbf{0}$ and $\mathbf{M}_k\mathbf{N}_{r,1} = \mathbf{0}$, $\forall k \geq 1$, it follows that $\mathbf{M}\mathbf{N}_1 = \mathbf{0}$. Hence, $\mathbf{N}_1$ belongs to the right nullspace of $\mathbf{M}$. The fact that the right nullspace contains *only* the four directions of $\mathbf{N}_1$ follows from the structure of Γ_3 and Γ_4, which are full rank and time varying (see (54), (55), and [41]). $\square$

Remark 1: The 18×3 block column $\mathbf{N}_{t,1}$ corresponds to global translations, i.e., translating both the sensing platform and the landmark by the same amount.

Remark 2: The 18×1 column $\mathbf{N}_{r,1}$ corresponds to global rotations of the sensing platform and the landmark about the gravity vector.

B. Observability Analysis of the EKF Linearized VINS Model

Ideally, any VINS estimator should employ a linearized system with an unobservable subspace that matches the true unobservable directions (57), both in number and structure. However, when linearizing about the estimated state, $\hat{\mathbf{x}}$, the observability matrix, $\mathbf{M}(\hat{\mathbf{x}})$, gains rank due to errors in the state estimates across time [34].

In particular, the last two block elements of $\mathbf{M}_k$ comprise identity matrices (i.e., are not a function of the linearization point), and thus, the nullspace directions corresponding to translation are preserved, i.e.,

$$\mathbf{M}(\hat{\mathbf{x}})\,\mathbf{N}_{t,1} = \mathbf{0}. \tag{59}$$

In contrast, in general, $\mathbf{M}(\hat{\mathbf{x}})\,\widehat{\mathbf{N}}_{r,1} \neq \mathbf{0}$, since $\widehat{\mathbf{N}}_{r,1}$ as well as $\widehat{\boldsymbol{\Gamma}}_2$, and thus, $\mathbf{M}(\hat{\mathbf{x}})$ depend on the linearization point [see (51), (53), and (57)]. That is, as different state estimates (i.e., the prior and posterior) are utilized when evaluating the system and measurement Jacobians, the directions in which the estimator gains information are altered. This causes the vector corresponding to global rotations, $\widehat{\mathbf{N}}_{r,1}$, not to be in the nullspace of $\mathbf{M}(\hat{\mathbf{x}})$, and as a result the rank of the observability matrix corresponding to the EKF linearized VINS model increases by one. This effect can also be easily verified by numerically evaluating the observability matrix during any experiment. The approach to address this issue is described in the following section.

V. OC-VINS: ALGORITHM DESCRIPTION

In order to address the EKF VINS inconsistency problem, we must ensure that (56) is satisfied for every block row of $\mathbf{M}(\hat{\mathbf{x}})$ when the state estimates are used for computing $\widehat{\mathbf{H}}_k$, and $\widehat{\boldsymbol{\Phi}}_{k,1}, \forall k > 0$, i.e.,

$$\widehat{\mathbf{H}}_k \widehat{\boldsymbol{\Phi}}_{k,1} \widehat{\mathbf{N}}_1 = \mathbf{0}, \quad \forall k > 0. \tag{60}$$

One way to enforce this is by requiring that at each time step, $\widehat{\boldsymbol{\Phi}}_{k+1,k}$ and $\widehat{\mathbf{H}}_k$ satisfy the following constraints:

$$\widehat{\mathbf{N}}_{k+1} = \widehat{\boldsymbol{\Phi}}_{k+1,k} \widehat{\mathbf{N}}_k \tag{61a}$$

$$\widehat{\mathbf{H}}_k \widehat{\mathbf{N}}_k = \mathbf{0}, \quad \forall k > 0 \tag{61b}$$

where $\widehat{\mathbf{N}}_k, k > 0$ is computed analytically (see Section V-A). This can be accomplished by appropriately modifying $\widehat{\boldsymbol{\Phi}}_{k+1,k}$ (see Section V-B) and $\widehat{\mathbf{H}}_k$ (see Section V-C).

In particular, rather than changing the linearization points explicitly (e.g., as in [23]), we maintain the nullspace, $\widehat{\mathbf{N}}_k$, at each time step, and use it to enforce the unobservable directions. This has the benefit of allowing us to linearize with the most accurate state estimates, hence, reducing the linearization error, while still explicitly adhering to the system observability properties.

Algorithm 1 OC-VINS Algorithm Overview

Initialization: Compute initial nullspace from (62)
while running **do**
   Propagation:
      Integrate state equations [see (10)-(15)]
      Compute nullspace at current time-step from (63)
      Compute $\boldsymbol{\Phi}_{k+1,k}$ from (22)
      Modify $\boldsymbol{\Phi}_{k+1,k}$ using (70)-(72)
      Propagate covariance [see (25)]
   Update:
   for all observed features **do**
      Compute measurement Jacobian from (29)
      Modify $\mathbf{H}_k$ using (79)-(83)
      Apply filter update [see (34)-(38)]
   end for
   New landmark initialization:
   for all new DFs observed **do**
      Initialize ${}^G\hat{\mathbf{f}}_i$, using approach in Appendix A
      Create nullspace block, $\widehat{\mathbf{N}}_{\mathbf{f}_i}$, for ${}^G\hat{\mathbf{f}}_i$ [see (63)]
      Augment $\widehat{\mathbf{N}}_k$ with the new sub-block $\widehat{\mathbf{N}}_{\mathbf{f}_i}$
   end for
end while

A. Nullspace Determination

The initial nullspace is analytically defined as [see (57)]

$$\widehat{\mathbf{N}}_1 = \begin{bmatrix} \mathbf{0}_3 & \mathbf{C}\left({}^I\hat{\bar{q}}_{G,1|0}\right){}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_{3\times 1} \\ \mathbf{0}_3 & -\lfloor {}^G\hat{\mathbf{v}}_{I,1|0} \times \rfloor {}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_{3\times 1} \\ \mathbf{I}_3 & -\lfloor {}^G\hat{\mathbf{p}}_{I,1|0} \times \rfloor {}^G\mathbf{g} \\ \mathbf{I}_3 & -\lfloor {}^G\hat{\mathbf{f}}_{1,1|0} \times \rfloor {}^G\mathbf{g} \end{bmatrix}. \tag{62}$$

At subsequent time steps, the nullspace is augmented to include additional block-rows corresponding to each new DF in the filter state, i.e.,

$$\widehat{\mathbf{N}}_k = \begin{bmatrix} \mathbf{0}_3 & \mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_{3\times 1} \\ \mathbf{0}_3 & -\lfloor {}^G\hat{\mathbf{v}}_{I,k|k-1} \times \rfloor {}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_{3\times 1} \\ \mathbf{I}_3 & -\lfloor {}^G\hat{\mathbf{p}}_{I,k|k-1} \times \rfloor {}^G\mathbf{g} \\ \mathbf{I}_3 & -\lfloor {}^G\hat{\mathbf{f}}_{1,k|k-\ell_1} \times \rfloor {}^G\mathbf{g} \\ \vdots & \vdots \\ \mathbf{I}_3 & -\lfloor {}^G\hat{\mathbf{f}}_{N,k|k-\ell_N} \times \rfloor {}^G\mathbf{g} \end{bmatrix} = \begin{bmatrix} \widehat{\mathbf{N}}_{t,k} & | & \widehat{\mathbf{N}}_{r,k} \end{bmatrix} \tag{63}$$

where the block-row $\widehat{\mathbf{N}}_{\mathbf{f}_i} = [\, \mathbf{I}_3 \quad -\lfloor {}^G\hat{\mathbf{f}}_{i,k|k-\ell_i} \times \rfloor {}^G\mathbf{g} \,], i = 1 \ldots N$, corresponds to the ith feature in the map, and is a function of the feature estimate ${}^G\hat{\mathbf{f}}_{i,k|k-\ell_i}$ at the time-step $k-\ell_i$, it was initialized.

B. Modification of the State Transition Matrix $\mathbf{\Phi}$

During the covariance propagation step, we must ensure that $\widehat{\mathbf{N}}_{k+1} = \widehat{\mathbf{\Phi}}_{k+1,k} \widehat{\mathbf{N}}_k$, or equivalently

$$\widehat{\mathbf{N}}_{t,k+1} = \widehat{\mathbf{\Phi}}_{k+1,k} \widehat{\mathbf{N}}_{t,k} \tag{64}$$

$$\widehat{\mathbf{N}}_{r,k+1} = \widehat{\mathbf{\Phi}}_{k+1,k} \widehat{\mathbf{N}}_{r,k}. \tag{65}$$

We note that the constraint involving $\widehat{\mathbf{N}}_{t,k}$ and $\widehat{\mathbf{N}}_{t,k+1}$ is automatically satisfied due to the structure of $\widehat{\mathbf{\Phi}}_{k+1,k}$ [see (44) and (63)], so we focus on $\widehat{\mathbf{N}}_{r,k}$. Specifically, we rewrite (65) element-wise as follows:[6]

$$
\begin{bmatrix}
\mathbf{C}\left({}^I\hat{\bar{q}}_{G,k+1|k}\right){}^G\mathbf{g} \\
\mathbf{0}_{3\times 1} \\
-\lfloor {}^G\hat{\mathbf{v}}_{I,k+1|k} \times\rfloor {}^G\mathbf{g} \\
\mathbf{0}_{3\times 1} \\
-\lfloor {}^G\hat{\mathbf{p}}_{I,k+1|k} \times\rfloor {}^G\mathbf{g}
\end{bmatrix}
=
$$

$$
\begin{bmatrix}
\widehat{\mathbf{\Phi}}^{(1,1)}_{k+1,1} & \widehat{\mathbf{\Phi}}^{(1,2)}_{k+1,1} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\
\mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\
\widehat{\mathbf{\Phi}}^{(3,1)}_{k+1,1} & \widehat{\mathbf{\Phi}}^{(3,2)}_{k+1,1} & \mathbf{I}_3 & \widehat{\mathbf{\Phi}}^{(3,4)}_{k+1,1} & \mathbf{0}_3 \\
\mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\
\widehat{\mathbf{\Phi}}^{(5,1)}_{k+1,1} & \widehat{\mathbf{\Phi}}^{(5,2)}_{k+1,1} & \delta t\mathbf{I}_3 & \widehat{\mathbf{\Phi}}^{(5,4)}_{k+1,1} & \mathbf{I}_3
\end{bmatrix}
\begin{bmatrix}
\mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g} \\
\mathbf{0}_{3\times 1} \\
-\lfloor {}^G\hat{\mathbf{v}}_{I,k|k-1} \times\rfloor {}^G\mathbf{g} \\
\mathbf{0}_{3\times 1} \\
-\lfloor {}^G\hat{\mathbf{p}}_{I,k|k-1} \times\rfloor {}^G\mathbf{g}
\end{bmatrix}
\tag{66}
$$

and collect the constraints resulting from each block row of the aforementioned vector. Specifically, from the first block row, we have

$$\mathbf{C}\left({}^I\hat{\bar{q}}_{G,k+1|k}\right){}^G\mathbf{g} = \widehat{\mathbf{\Phi}}^{(1,1)}_{k+1,1} \mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g}. \tag{67}$$

As we saw in Section IV-A [see (46)], the 3×3 matrix $\widehat{\mathbf{\Phi}}^{(1,1)}_{k+1,k}$ is a rotation matrix. Let $\widehat{\mathbf{\Phi}}^{(1,1)}_{k+1,k}$ be the rotation matrix obtained by integrating the gyroscopes, which is described by the quaternion $\hat{\bar{q}}$. That is

$$\mathbf{C}(\hat{\bar{q}}) = \widehat{\mathbf{\Phi}}^{(1,1)}_{k+1,k}. \tag{68}$$

We seek a perturbed $\bar{q}^\star$ as a solution, to the optimization problem

$$
\begin{aligned}
\underset{\bar{q}}{\text{minimize}} \quad & J(\bar{q}) = \frac{1}{2}\|\bar{q} - \hat{\bar{q}}\|_2^2 \\
\text{subject to} \quad & \mathbf{C}\left({}^I\hat{\bar{q}}_{G,k+1|k}\right){}^G\mathbf{g} = \mathbf{C}(\bar{q})\mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g} \\
& \bar{q}^T\bar{q} = 1.
\end{aligned}
\tag{69}
$$

As we show in Appendix C, we can compute the solution to (69) in closed-form, and determine the observability constrained

[6]Note that due to the structure of the matrices $\mathbf{\Phi}_{k+1,k}$ [see (44)], $\mathbf{N}_{r,k}$, and $\mathbf{N}_{r,k+1}$ [see (63)], we only need to consider the first five block elements of (65), while the equalities for the remaining ones, i.e., the elements corresponding to the features, are automatically satisfied.

$\widehat{\mathbf{\Phi}}^{(1,1)\star}_{k+1,k}$ as

$$\widehat{\mathbf{\Phi}}^{(1,1)\star}_{k+1,k} = \mathbf{C}(\bar{q}^\star). \tag{70}$$

The requirements for the third- and fifth-block rows are

$$\widehat{\mathbf{\Phi}}^{(3,1)}_{k+1,1}\mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g} = \lfloor {}^G\hat{\mathbf{v}}_{I,k|k-1} - {}^G\hat{\mathbf{v}}_{I,k+1|k} \times\rfloor {}^G\mathbf{g} \tag{71}$$

$$\widehat{\mathbf{\Phi}}^{(5,1)}_{k+1,1}\mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g} = \lfloor \delta t {}^G\hat{\mathbf{v}}_{I,k|k-1} + {}^G\hat{\mathbf{p}}_{I,k|k-1} \times\rfloor {}^G\mathbf{g}$$
$$- \lfloor {}^G\hat{\mathbf{p}}_{I,k+1|k} \times\rfloor {}^G\mathbf{g} \tag{72}$$

both of which are in the form $\mathbf{A}\mathbf{u} = \mathbf{w}$, where $\mathbf{u}$ and $\mathbf{w}$ are nullspace vector elements that are fixed. In order to ensure that (71) and (72) are satisfied, we seek to find a perturbed $\mathbf{A}^\star$, for $\mathbf{A} = \widehat{\mathbf{\Phi}}^{(3,1)}_{k+1,1}$ and $\mathbf{A} = \widehat{\mathbf{\Phi}}^{(5,1)}_{k+1,1}$ that fulfills the constraint. To compute the minimum perturbation, $\mathbf{A}^\star$, of $\mathbf{A}$, we formulate the following minimization problem:

$$
\begin{aligned}
\underset{\mathbf{A}^\star}{\text{minimize}} \quad & \|\mathbf{A}^\star - \mathbf{A}\|_{\mathcal{F}}^2 \\
\text{subject to} \quad & \mathbf{A}^\star\mathbf{u} = \mathbf{w}
\end{aligned}
\tag{73}
$$

where $\|\cdot\|_{\mathcal{F}}$ denotes the Frobenius matrix norm. After employing the method of Lagrange multipliers, and solving the corresponding KKT optimality conditions, the optimal $\mathbf{A}^\star$ that fulfills (73) is

$$\mathbf{A}^\star = \mathbf{A} - (\mathbf{A}\mathbf{u} - \mathbf{w})(\mathbf{u}^T\mathbf{u})^{-1}\mathbf{u}^T. \tag{74}$$

Once we have computed the modified $\widehat{\mathbf{\Phi}}^{(1,1)\star}_{k+1,1}$ from (70), and $\widehat{\mathbf{\Phi}}^{(3,1)\star}_{k+1,1}$ and $\widehat{\mathbf{\Phi}}^{(5,1)\star}_{k+1,1}$ from (73) and (74), we update the corresponding elements of $\widehat{\mathbf{\Phi}}_{k+1,k}$ and proceed with the covariance propagation (see Section III-A).

C. Modification of the Measurement Jacobian $\mathbf{H}$

During each update step, we seek to satisfy $\widehat{\mathbf{H}}_k\widehat{\mathbf{N}}_k = \mathbf{0}$ [see (61b)]. In turn, this means that

$$\widehat{\mathbf{H}}_k\widehat{\mathbf{N}}_{t,k} = \mathbf{0} \tag{75}$$

$$\widehat{\mathbf{H}}_k\widehat{\mathbf{N}}_{r,k} = \mathbf{0} \tag{76}$$

must both hold.

Expressing (75) for a single point, we have [see (29) and (63)]

$$\widehat{\mathbf{H}}_c\begin{bmatrix}\widehat{\mathbf{H}}_{\bar{q}} & \mathbf{0}_{3\times 9} & \widehat{\mathbf{H}}_{\mathbf{p}} & | & \widehat{\mathbf{H}}_{\mathbf{f}}\end{bmatrix}\begin{bmatrix}\mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{0}_3 \\ \mathbf{I}_3 \\ \mathbf{I}_3\end{bmatrix} = \mathbf{0} \tag{77}$$

which is satisfied automatically, since $\widehat{\mathbf{H}}_{\mathbf{p}} = -\widehat{\mathbf{H}}_{\mathbf{f}}$ [see (32) and (33)]. Hence, the nullspace direction corresponding to translation is not violated.

Expanding the second constraint (76), we have

$$\widehat{\mathbf{H}}_c \begin{bmatrix} \widehat{\mathbf{H}}_{\bar{q}} & \mathbf{0}_{3\times 9} & \widehat{\mathbf{H}}_{\mathbf{p}} & | & \widehat{\mathbf{H}}_{\mathbf{f}} \end{bmatrix} \begin{bmatrix} \mathbf{C}\left({}^{I}\hat{\bar{q}}_{G,k|k-1}\right){}^{G}\mathbf{g} \\ \mathbf{0}_{3\times 1} \\ -\lfloor {}^{G}\hat{\mathbf{v}}_{I,k|k-1}\times\rfloor {}^{G}\mathbf{g} \\ \mathbf{0}_{3\times 1} \\ -\lfloor {}^{G}\hat{\mathbf{p}}_{I,k|k-1}\times\rfloor {}^{G}\mathbf{g} \\ -\lfloor {}^{G}\hat{\mathbf{f}}_{k|k-\ell}\times\rfloor {}^{G}\mathbf{g} \end{bmatrix} = \mathbf{0}. \tag{78}$$

Since $\widehat{\mathbf{H}}_{\mathbf{p}} = -\widehat{\mathbf{H}}_{\mathbf{f}}$, (78) is equivalent to satisfying the following relationship:

$$\begin{bmatrix} \widehat{\mathbf{H}}_{c\bar{q}} & \widehat{\mathbf{H}}_{c\mathbf{p}} \end{bmatrix} \begin{bmatrix} \mathbf{C}\left({}^{I}\hat{\bar{q}}_{G,k|k-1}\right){}^{G}\mathbf{g} \\ \lfloor {}^{G}\hat{\mathbf{f}}_{k|k-\ell} - {}^{G}\hat{\mathbf{p}}_{I,k|k-1}\times\rfloor {}^{G}\mathbf{g} \end{bmatrix} = 0 \tag{79}$$

with $\widehat{\mathbf{H}}_{c\bar{q}} = \widehat{\mathbf{H}}_c \widehat{\mathbf{H}}_{\bar{q}}$, and $\widehat{\mathbf{H}}_{c\mathbf{p}} = \widehat{\mathbf{H}}_c \widehat{\mathbf{H}}_{\mathbf{p}}$. Note that (79) is a constraint of the form $\mathbf{A}\mathbf{u} = \mathbf{0}$, where $\mathbf{u}$ is a function of the nullspace elements, and hence, is fixed, while $\mathbf{A}$ comprises block elements of the measurement Jacobian.

We compute the optimal $\mathbf{A}^\star$ that satisfies (79) by formulating an optimization problem similar to (73) for $\mathbf{w} = \mathbf{0}$, whose optimal solution [see (74)] $\mathbf{A}^\star$ is computed as

$$\mathbf{A}^\star = \mathbf{A} - \mathbf{A}\mathbf{u}(\mathbf{u}^T\mathbf{u})^{-1}\mathbf{u}^T. \tag{80}$$

After determining the optimal $\mathbf{A}^\star$, we recover the Jacobian elements as

$$\widehat{\mathbf{H}}_{c\bar{q}}^\star = \mathbf{A}^{(1:2,1:3)\star} \tag{81}$$

$$\widehat{\mathbf{H}}_{c\mathbf{p}}^\star = \mathbf{A}^{(1:2,4:6)\star} \tag{82}$$

$$\widehat{\mathbf{H}}_{c\mathbf{f}}^\star = -\widehat{\mathbf{H}}_{c\mathbf{p}}^\star \tag{83}$$

where the superscripts (i:j, m:n) denote the submatrix spanning rows i to j and columns m to n. Hence, the modified observation matrix is

$$\widehat{\mathbf{H}}_k^\star = \begin{bmatrix} \widehat{\mathbf{H}}_{c\bar{q}}^\star & \mathbf{0}_{2\times 9} & \widehat{\mathbf{H}}_{c\mathbf{p}}^\star & \widehat{\mathbf{H}}_{c\mathbf{f}}^\star \end{bmatrix}. \tag{84}$$

Having computed the modified measurement Jacobian, we proceed with the filter update as described in Section III-B. By following this process, we ensure that the EKF does not gain information along the unobservable directions of the system.

VI. SIMULATIONS

We conducted Monte-Carlo simulations to evaluate the impact of the proposed OC-VINS method on estimator consistency. We applied the proposed methodology to two VINS systems: 1) V-SLAM (see Section VI-A) and 2) the MSC-KF, which performs visual-inertial localization without constructing a map (see Section VI-B).

We employed two error metrics in order to evaluate the consistency and the accuracy of the considered estimators, namely, the normalized estimation error squared (NEES) and the root mean squared error (RMSE) [18]. For each Monte-Carlo simulation,

$i, i = 1, \ldots, M_s$, we compute the pose estimate at each time-step j, comprising the orientation error $\delta\boldsymbol{\theta}_{ij}$ and the position error $\tilde{\mathbf{p}}_{ij}$ [see (16)]. We compute the NEES for each estimator at each time-step j, averaged over all Monte-Carlo simulations as

$$\theta_{\text{NEES}_j} = \frac{1}{M_s}\sum_{i=1}^{M_s} \delta\boldsymbol{\theta}_{ij}^T \mathbf{P}_{\delta\boldsymbol{\theta}_{ij}}^{-1} \delta\boldsymbol{\theta}_{ij} \tag{85}$$

$$\mathbf{p}_{\text{NEES}_j} = \frac{1}{M_s}\sum_{i=1}^{M_s} \tilde{\mathbf{p}}_{ij}^T \mathbf{P}_{\tilde{\mathbf{p}}_{ij}}^{-1} \tilde{\mathbf{p}}_{ij} \tag{86}$$

where $\mathbf{P}_{\delta\boldsymbol{\theta}_{ij}}$ and $\mathbf{P}_{\tilde{\mathbf{p}}_{ij}}$ are the 3×3 covariance matrices corresponding to orientation and position, respectively, which are computed by the filter at the jth time step of the ith simulation. The RMSE at each time step is computed as

$$\theta_{\text{RMSE}_j} = \sqrt{\frac{1}{M_s}\sum_{i=1}^{M_s} \delta\boldsymbol{\theta}_{ij}^T \delta\boldsymbol{\theta}_{ij}} \tag{87}$$

$$\mathbf{p}_{\text{RMSE}_j} = \sqrt{\frac{1}{M_s}\sum_{i=1}^{M_s} \tilde{\mathbf{p}}_{ij}^T \tilde{\mathbf{p}}_{ij}}. \tag{88}$$

These metrics are used in both simulations (see Figs. 2 and 3).

A. Simulation 1: Application of the Proposed Framework to V-SLAM

In this section, we present the results of applying our proposed OC-VINS to V-SLAM, which we term OC-V-SLAM. We compared its performance with the standard V-SLAM (Std-V-SLAM), as well as the ideal V-SLAM that linearizes about the true state.[7] Specifically, we computed the RMSE and NEES over 20 trials in which the camera-IMU platform traversed a circular trajectory of radius 5 m at an average velocity of 60 cm/s. The camera[8] observed visual features distributed on the interior wall of a circumscribing cylinder with radius 6 m and height 2 m [see Fig. 2(c)]. The effect of inconsistency during a single run is depicted in Fig. 2(f), where the error and corresponding 3σ bounds of uncertainty are plotted for the rotation about the gravity vector. As evident, the Std-V-SLAM gains spurious information, hence, reducing its 3σ bounds of uncertainty, while the Ideal-V-SLAM and the OC-V-SLAM do not. The Std-V-SLAM becomes inconsistent on this run as the orientation errors fall outside of the uncertainty bounds, while both the Ideal-V-SLAM and the OC-V-SLAM remain consistent. Fig. 2 also displays the RMSE and NEES plots, in which we observe that the OC-V-SLAM attains orientation accuracy and consistency levels similar to the Ideal-V-SLAM, while significantly outperforming the Std-V-SLAM [see Fig. 2(a) and (b)]. Similarly, the OC-V-SLAM obtains better positioning accuracy compared with the Std-V-SLAM [see Fig. 2(d) and (e)].

---

[7]Since the ideal V-SLAM has access to the true state, it is not realizable in practice, but we include it here as a baseline comparison.

[8]The camera had $45°$ field of view, with $\sigma_{px} = 1px$, while the IMU was modeled after MEMS quality sensors.
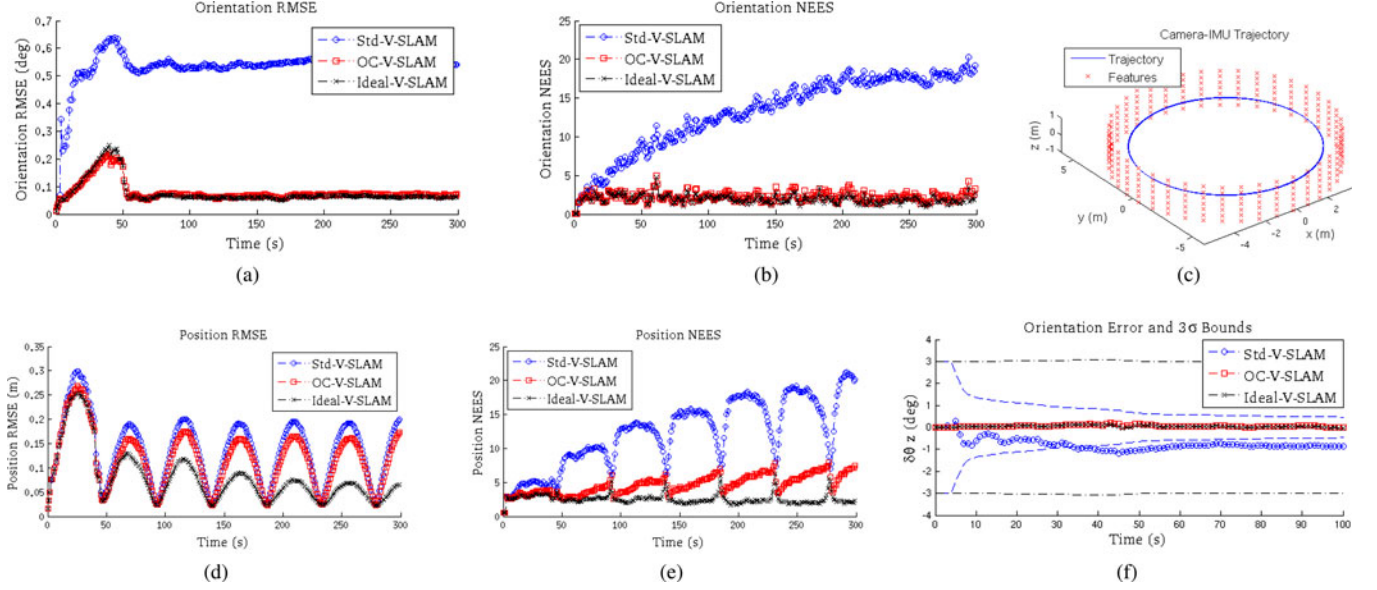
Fig. 2. Simulation 1: The RMSE and NEES errors for orientation (a)–(b) and position (d)–(e) plotted for all three filters, averaged per time step over 20 Monte-Carlo trials. (c) Camera-IMU trajectory and 3-D features. (f) Error and 3σ bounds for the rotation about the gravity vector, plotted for the first 100 s of a representative run.
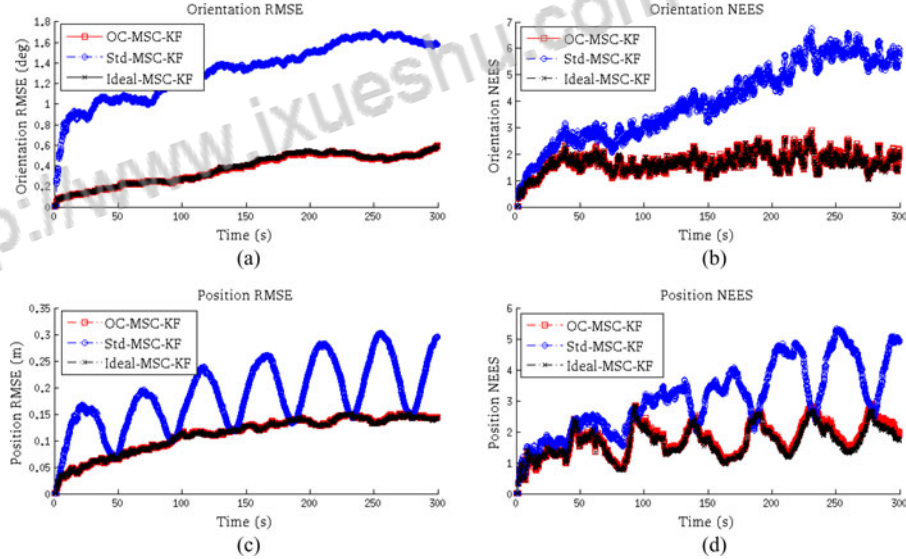

Fig. 3. Simulation 2: The average RMSE and NEES over 30 Monte-Carlo simulation trials for orientation (a)–(b) and position (c)–(d). Note that the OC-MSC-KF attains performance almost indistinguishable to the Ideal-MSC-KF.

B. Simulation 2: Application of the Proposed Framework to MSC-KF

We applied our OC-VINS methodology to the MSC-KF [11], [22], which we term the OC-MSC-KF. In the MSC-KF framework, all the measurements to a given OF are incorporated during a single update step of the filter, after which each OF is marginalized. Hence, in the OC-MSC-KF, we do not maintain the block-rows of the nullspace corresponding to the features [i.e., $\mathbf{N}_{\mathbf{f}_i}, i = 1, \ldots, N$; see (63)]. Instead, we propagate forward in time only the portion of the nullspace corresponding

to the sensor-platform state, and we form the feature nullspace block row for each feature only when it is processed in an update.

We conducted Monte-Carlo simulations to evaluate the consistency of the proposed method applied to the MSC-KF. Specifically, we compared the standard MSC-KF (Std-MSC-KF), with the observability constrained MSC-KF (OC-MSC-KF), which is obtained by applying the methodology described in Section V, as well as the Ideal-MSC-KF, whose Jacobians are linearized at the true states, which we use as a benchmark. We evaluated the RMSE and NEES over 30 trials (see Fig. 3) in which the camera-IMU platform traversed a circular trajectory of radius
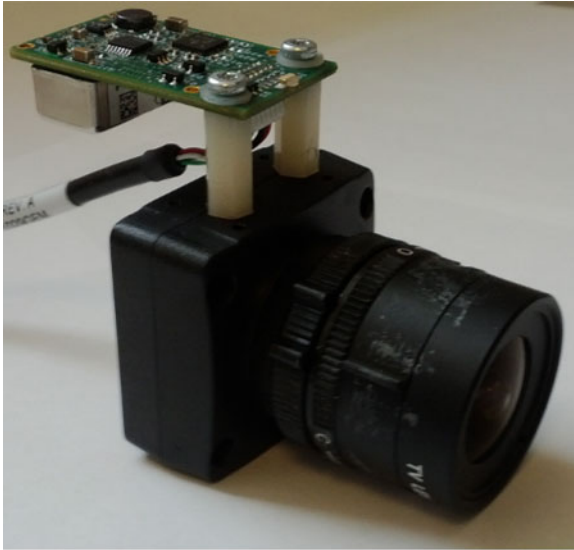
(a)

(b)

Fig. 4.   (a) Experimental testbed comprises a light-weight InterSense NavChip IMU and a PointGrey Chameleon Camera. IMU signals are sampled at a frequency of 100 Hz, while camera images are acquired at 7.5 Hz. The dimensions of the sensing package are approximately 6 cm tall, by 5 cm wide, by 8 cm deep. (b) AscTech Pelican quadrotor on which the camera-IMU package was mounted during the indoor experiments (see Sections VII-B and C).

5 m at an average speed of 60 cm/s and observed 50 randomly distributed features per image. As evident, the OC-MSC-KF outperforms the Std-MSC-KF and attains performance almost indistinguishable from the Ideal-MSC-KF in terms of RMSE and NEES.

VII. Experimental Results

The proposed OC-VINS framework has been validated experimentally and compared with standard VINS approaches. Specifically, we evaluated the performance of OC-V-SLAM (see Section VII-B) and OC-MSC-KF (see Sections VII-C and VII-D) on both indoor and outdoor datasets. In our experimental setup, we utilized a light-weight sensing platform comprised of an InterSense NavChip IMU and a PointGrey Chameleon camera (see Fig. 4). During the indoor experimental tests (see

Sections VII-B and C), the sensing platform was mounted on an Ascending Technologies Pelican quadrotor equipped with a VersaLogic Core 2 Duo single board computer. For the outdoor dataset, the sensing platform was head-mounted on a bicycle helmet (see Section VII-D) and interfaced to a handheld Sony Vaio. We hereafter provide an overview of the system implementation, along with a discussion of the experimental setup and results.

A. Implementation Remarks

The image processing is separated into two components: One to extract and track OFs over short time windows, and one to extract and match DFs to use in V-SLAM.

The OFs are computed from images using the Shi–Tomasi corner detector [42]. After acquiring image k, it is inserted into a sliding-window buffer of m images, $\{k - m + 1, k - m + 2, \ldots, k\}$. We then extract features from the first image in the window and track them pairwise through the window using the KLT tracking algorithm [32]. To remove outliers from the resulting tracks, we use a two-point algorithm to find the essential matrix between successive frames [47]. Specifically, given the filter's estimated rotation (from the gyroscopes' measurements) between image i and j, $^i\hat{\bar{q}}_j$, we estimate the essential matrix from only two feature correspondences. This approach is more robust than the five-point algorithm [43] because it provides two solutions for the essential matrix rather than up to ten. Moreover, it requires only two data points, and thus, it reaches a consensus with fewer hypotheses when used in a RANSAC framework [44].

The DFs are extracted using SIFT descriptors [33]. To identify global features observed from several different images, we first utilize a vocabulary tree (VT) structure for image matching [45]. Specifically, for an image taken at time k, the VT is used to select which image(s) taken at times $1, 2, \ldots, k - 1$ correspond to the same physical scene. Among those images that the VT reports as potential matches, the SIFT descriptors from each of them are compared with those from image k to create tentative feature correspondences. The epipolar constraint is then enforced using RANSAC and Nister's five-point algorithm [43] to eliminate outliers. It is important to note that the images used to construct the VT are not taken along our experimental trajectory, but rather are randomly selected from a set of representative images that were previously collected in similar environments. This is beneficial, since it allows us to build the VT once, offline, and reuse it during experimentation.

B. Experiment 1: Indoor Validation of OC-V-SLAM

In the first experimental trial, we compared the performance of OC-V-SLAM with that of Std-V-SLAM on an indoor trajectory. The sensing platform traveled a total distance of 172.5 m, covering three loops over two floors in Walter Library, University of Minnesota. The quadrotor was returned to its starting location at the end of the trajectory, to provide a quantitative characterization of the achieved accuracy.

Opportunistic features were tracked using a window of $m = 10$ images. Every m camera frames, up to 30 features from all
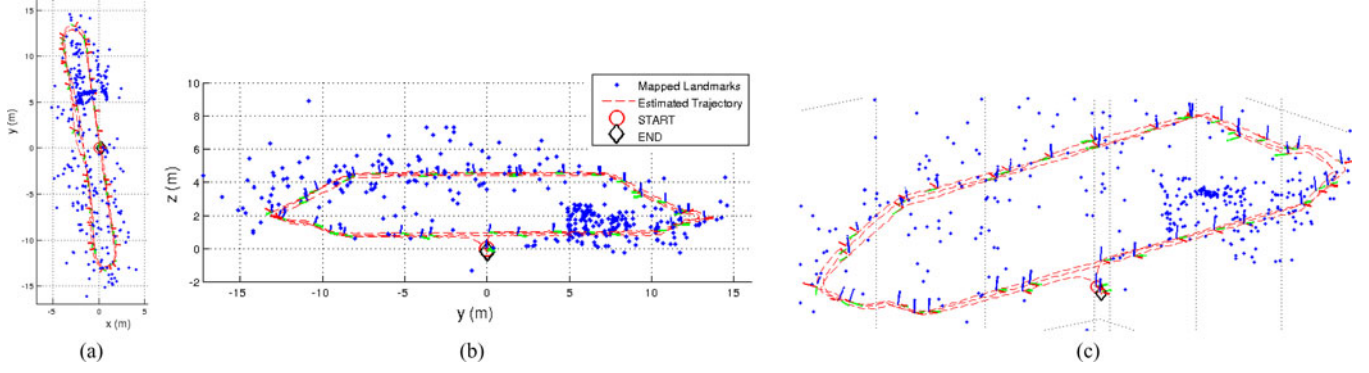
Fig. 5. Experiment 1: Estimated 3-D trajectory over the three traversals of the two floors of the building, along with the estimated positions of the DFs. (a) Projection on the x- and y-axes, (b) projection on the y- and z-axes, and (c) 3-D view of the overall trajectory and the estimated features.
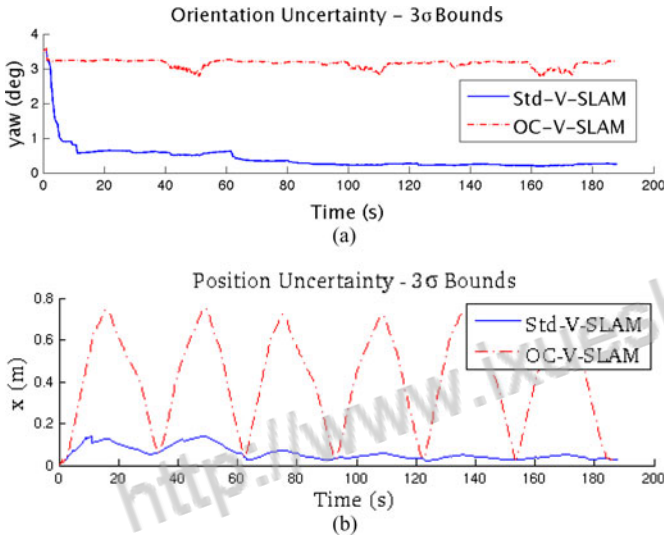

Fig. 6. Experiment 1: Comparison of the estimated 3σ error bounds for attitude and position between Std-V-SLAM and OC-V-SLAM.

available DFs are initialized and the state vector is augmented with their 3-D coordinates. The process of initializing DFs [34] is continued until the occurrence of the first loop closure; from that point on, no new DFs are considered and the filter relies upon the reobservation of previously initialized DFs and the processing of OFs.

For both the Std-V-SLAM and the OC-V-SLAM, the final position error was approximately 34 cm, which is less than 0.2% of the total distance traveled (see Fig. 5). However, the estimated covariances from the Std-V-SLAM are smaller than those from the OC-V-SLAM (see Fig. 6). Furthermore, uncertainty estimates from the Std-V-SLAM decreased in directions that are unobservable (i.e., rotations about the gravity vector); this violates the observability properties of the system and demonstrates that spurious information is injected to the filter. In particular, Fig. 6(a) highlights the difference in estimated yaw uncertainty between the OC-V-SLAM and the Std-V-SLAM. In contrast to the OC-V-SLAM, the Std-V-SLAM covariance rapidly decreases, although no absolute heading information is

provided by the system. Similarly, large differences can be seen in the estimated covariance for the x-axis position estimates [see Fig. 6(b)]. The Std-V-SLAM estimates a much smaller uncertainty than the OC-V-SLAM, supporting the claim that the Std-V-SLAM tends to be inconsistent.

C. Experiment 2: Indoor Validation of the OC-MSC-KF

We validated the proposed OC-MSC-KF on real-world data. The first test comprised a trajectory 50 m in length that covered three loops in an indoor area, after which the testbed was returned to its initial position. At the end of the trajectory, the Std-MSC-KF had a position error of 18.73 cm, while the final error for the OC-MSC-KF was 16.39 cm (approximately 0.38% and 0.33% of the distance traveled, respectively). In order to assess the impact of inconsistency on the orientation estimates of both methods, we used as ground truth the rotation between the first and last images computed independently using BLS and feature point matches. The Std-MSC-KF had final orientation error $\begin{bmatrix} 0.15 & -0.23 & -5.13 \end{bmatrix}$ degrees for roll, pitch, and yaw (rpy), while the rpy errors for the OC-MSC-KF were $\begin{bmatrix} 0.19 & -0.20 & -1.32 \end{bmatrix}$ degrees, respectively. Note that although the roll and pitch estimates for the two filters are of comparable accuracy, the error in the yaw estimate of the OC-MSC-KF is almost four times smaller than that of the Std-MSC-KF.

In addition to achieving higher accuracy, for yaw in particular, the OC-MSC-KF is more conservative since it strictly adheres to the unobservable directions of the system. This is evident in both the position and orientation uncertainties. We plot the y-axis position and yaw angle uncertainties in Fig. 7, as representative results. Most notably, the yaw uncertainty of the OC-MSC-KF increases and reaches approximately $1.13°$ (3σ), while for the Std-MSC-KF, it reduces to $0.87°$ (3σ). This indicates that the Std-MSC-KF gains spurious (nonexistent) heading information, which leads to inconsistency. Finally, in Fig. 8, we show the 3-D trajectory along with an overhead (x–y) view. It is evident that the Std-MSC-KF yaw error impacts the position accuracy, as the Std-MSC-KF trajectory exhibits a rotation with respect to the OC-MSC-KF.
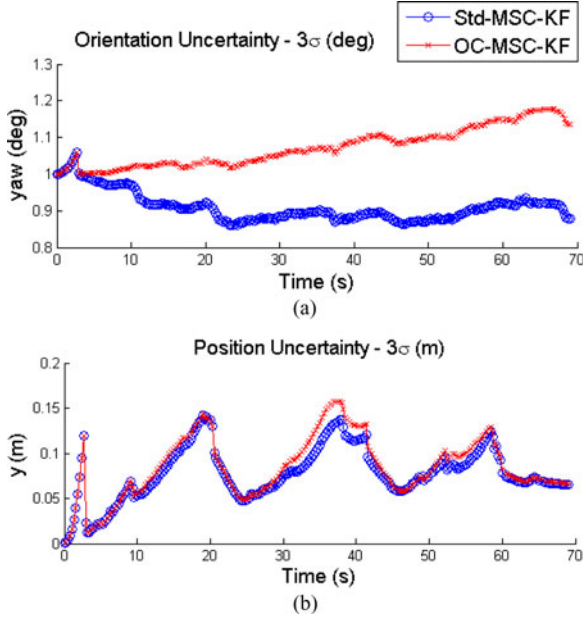
Fig. 7. Experiment 2: (a) Position and (b) orientation uncertainties (3σ bounds) for the yaw angle and the y-axis, which demonstrate that the Std-MSC-KF gains spurious information about its orientation.
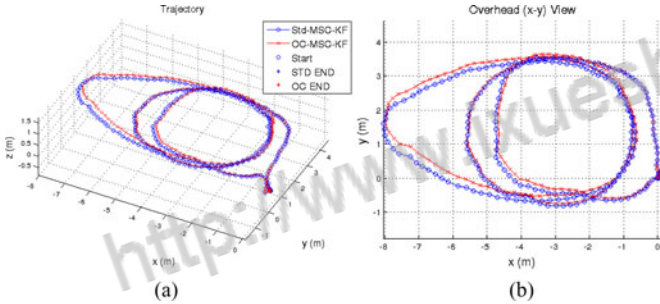

Fig. 8. Experiment 2: (a) 3-D trajectory and (b) corresponding overhead (x–y) view.

D. Experiment 3: Outdoor Validation of the OC-MSC-KF

In our final experimental trial, we tested the OC-MSC-KF on a large outdoor dataset (approximately 1.5 km in length). Fig. 9(a) depicts the OC-MSC-KF (red) and the Std-MSC-KF (blue) trajectory estimates, along with position markers from a low-grade onboard GPS receiver (green). In order to assess the accuracy of both filters, the estimates are overlaid on an overhead image taken from Google-Earth.

Fig. 9(b) depicts a zoomed-in plot of the starting location (center) for both filters, along with the final position estimates. In order to evaluate the accuracy of the proposed method, the sensing platform was returned to its starting location at the end of the trajectory. The OC-MSC-KF obtains a final position error of 4.38 m (approximately 0.3% of the distance travelled), while the Std-MSC-KF obtains a final position error of 10.97 m. This represents an improvement in performance of approximately 60%.

The filters' performance is also illustrated visually in Fig. 9(c) which shows a zoomed-in plot of the turn-around point. The OC-MSC-KF estimates remain on the light-brown portion of the

ground (which is the sidewalk), which coincides with the true trajectory. In contrast, the Std-MSC-KF estimates drift over the dark triangles in the image, which are wading pools filled with water. This shifting of the trajectory represents a slight rotation around the vertical axis, indicating a violation of the rotation nullspace direction $\mathbf{N}_r$.

Fig. 10 depicts the uncertainty in yaw (corresponding to rotations about the gravity vector) and along the x-axis (perpendicular to the direction of motion). It is clear that the Std-MSC-KF reduces its uncertainty in its heading direction, indicating that the filter gains spurious information, while for the OC-MSC-KF, the uncertainty in yaw continuously increases, as it should in the absence of absolute heading information.

VIII. CONCLUSION AND FUTURE WORK

In this paper, we studied one of the root causes of inconsistency for linearized estimators applied to VINS. We were motivated by the fact that in navigation applications, it is of paramount importance to have not only a precise pose estimate at each time instant but an accurate assessment of the corresponding pose estimate uncertainty as well.

Since both the accuracy and attributed uncertainty of an estimate are affected by the directions in which information can be injected into the system, we began by analytically studying the observability properties of VINS. We corroborated earlier findings that the true VINS model has four unobservable directions (corresponding to three d.o.f. global translations and one d.o.f. rotation about the gravity vector) and, for the first time, derived expressions for describing them analytically. Moreover, we showed that for standard filtering approaches, such as the EKF, linearization errors can alter the system's observability properties, both in the structure and number of the unobservable directions, and cause the estimator to acquire spurious information about its state. This, in effect, erroneously reduces the estimator uncertainty, making it overconfident, and causes drift along the unobservable directions of the system.

Based on our analysis, we proposed an estimator modification that reduces inconsistency by ensuring the VINS observability properties are upheld within the framework of existing linearized estimators. Specifically, we proposed simple rules to modify the Jacobians of the system and measurement functions, which explicitly enforce that no information would be gained along the unobservable directions. This strategy required maintaining the nullspace of the system's observability matrix, which we provide in closed form as a function of the state estimates.

In order to validate the accuracy and consistency of the proposed OC-VINS framework, we evaluated the performance of our method on simulated and real-world data. Specifically, in simulation, we studied the NEES and RMSE for three approaches: 1) the standard VINS method that linearizes at the current state estimate, 2) the ideal VINS method that linearizes at the true state, and 3) the OC-VINS which ensures the observability properties of the system are respected. We observed that the standard VINS approach tends to become inconsistent, and in particular, it reduces its uncertainty about global yaw (i.e., rotation about the gravity vector) over time. In contrast,

(a)

(b)

(c)

Fig. 9. Experiment 3: (a) Outdoor experimental trajectory covering 1.5 km across the University of Minnesota campus. The red (blue) line denotes the OC-MSC-KF (Std-MSC-KF) estimated trajectory. The green circles denote a low-quality GPS-based estimate of the position across the trajectory. (b) Zoom-in view of the beginning/end of the run. Both filters start with the same initial pose estimate; however, the error for the Std-MSC-KF at the end of the run is 10.97 m, while for the OC-MSC-KF, the final error is 4.38 m (an improvement of almost 60%). Furthermore, the final error for the OC-MSC-KF is approximately 0.3% of the distance traveled. (c) Zoomed-in view of the turn-around point. The Std-MSC-KF trajectory is shifted compared with the OC-MSC-KF, which remains on the path (light-brown region).
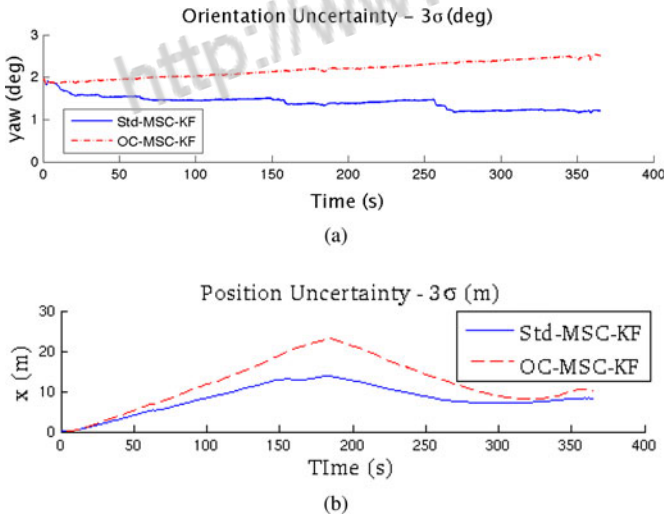

Fig. 10. Experiment 3: (a) Orientation uncertainty about the vertical axis (z). Since rotations about gravity are unobservable, the Std-MSC-KF should not gain any information in this direction. However, as evident from this plot, the Std-MSC-KF uncertainty reduces, indicating inconsistency. For the OC-MSC-KF, the uncertainty does not decrease, indicating that the OC-MSC-KF respects the unobservable system directions. (b) Position uncertainty along the x-axis (perpendicular to the direction of motion) for the Std-MSC-KF and OC-MSC-KF, respectively. The OC-MSC-KF maintains more conservative estimates for position, indicating that the Std-MSC-KF may be inconsistent.

both the ideal VINS method, and the proposed OC-VINS, respect the observability properties of the system and do not gain information about global yaw. We corroborated this analysis through our experimental trials, in which we demonstrated that

the proposed approach has improved the consistency performance compared with standard VINS methods. In particular, it achieves smaller pose-estimate error and maintains more conservative uncertainty bounds primarily due to the fact that it respects the observability properties of the system and does not acquire information along the direction of rotation about gravity.

In our future work, we are interested in investigating additional sources of the estimator inconsistency. In particular, we plan to focus on the case when the true pdf of the state has multiple modes, while we attempt to approximate it as unimodal.

APPENDIX A

As the camera-IMU platform moves into new environments, new features must be added into the map. This entails intersecting the bearing measurements from multiple camera observations to obtain an initial estimate of each new feature's 3-D location, as well as computing the initial covariance and cross-correlation between the new landmark estimate and the state. We solve this as a minimization problem over a parameter vector $\mathbf{x} = \begin{bmatrix} \mathbf{x}_s^T & | & \mathbf{f}^T \end{bmatrix}^T$, where $\mathbf{x}_s = \begin{bmatrix} \mathbf{x}_{s,1}^T & \cdots & \mathbf{x}_{s,m}^T \end{bmatrix}^T$ is the vector of m camera poses from which the new landmark, $\mathbf{f}$, was observed. Specifically, we minimize the following cost function:

$$C\left(\mathbf{x}\right) = \frac{1}{2}\{(\mathbf{x} - \hat{\mathbf{x}})^T \begin{bmatrix} \mathbf{P}_{ss}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \end{bmatrix} (\mathbf{x} - \hat{\mathbf{x}})$$

$$+ \sum_{i=1}^{m} \left(\mathbf{z}_i - \mathbf{h}_i\left(\mathbf{x}\right)\right)^T \mathbf{R}_i^{-1} \left(\mathbf{z}_i - \mathbf{h}_i\left(\mathbf{x}\right)\right)\} \quad (89)$$

where $\mathbf{P}_{ss}^{-1}$ is the information matrix (prior) of the state estimates across all poses obtained from the filter.[9] The fact that we have no initial information about the feature location is denoted by the block (2,2) element of the prior information being equal to zero. The m measurements $\mathbf{z}_i, i = 1 \ldots m$, are the perspective projection observations of the point $\mathbf{f}$ [see (27)], which are corrupted by zero-mean white Gaussian noise with covariance $\mathbf{R}_i = \sigma_\eta^2 \mathbf{I}_2$.

We obtain an initial guess for the landmark location using any intersection method, and then, we iteratively minimize (89). At each iteration, we need to solve the following linear system of equations:

$$\begin{bmatrix} \mathbf{P}_{ss}^{-1} + \mathbf{H}_s^T \mathbf{R}^{-1} \mathbf{H}_s & \mathbf{H}_s^T \mathbf{R}^{-1} \mathbf{H}_f \\ \mathbf{H}_f^T \mathbf{R}^{-1} \mathbf{H}_s & \mathbf{H}_f^T \mathbf{R}^{-1} \mathbf{H}_f \end{bmatrix} \begin{bmatrix} \tilde{\mathbf{x}}_s \\ \tilde{\mathbf{f}} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_s^T \mathbf{R}^{-1} \\ \mathbf{H}_f^T \mathbf{R}^{-1} \end{bmatrix} \tilde{\mathbf{z}}$$

$$\Leftrightarrow \begin{bmatrix} \mathbf{A} & \mathbf{U} \\ \mathbf{U}^T & \mathbf{C} \end{bmatrix} \tilde{\mathbf{x}} = \begin{bmatrix} \mathbf{D} \\ \mathbf{E} \end{bmatrix} \tilde{\mathbf{z}} \quad (90)$$

where $\tilde{\mathbf{x}}_s$ and $\tilde{\mathbf{f}}$ are the unknown correction terms for which we are solving, and $\tilde{\mathbf{z}}$ is the vector of stacked measurement residuals, i.e.,

$$\tilde{\mathbf{z}} = \mathbf{z} - \mathbf{h}(\mathbf{x})$$

$$= \begin{bmatrix} \mathbf{z}_1^T & \cdots & \mathbf{z}_m^T \end{bmatrix}^T - \begin{bmatrix} \mathbf{h}_1(\mathbf{x})^T & \cdots & \mathbf{h}_m(\mathbf{x})^T \end{bmatrix}^T \quad (91)$$

$\mathbf{R} = \mathrm{diag}(\mathbf{R}_1, \ldots, \mathbf{R}_m)$, and $\mathbf{H}_f = \nabla_{\mathbf{f}} \mathbf{h}(\mathbf{x})$ and $\mathbf{H}_s = \nabla_{\mathbf{x}_s} \mathbf{h}(\mathbf{x})$ are the Jacobians of the stacked measurement vector, $\mathbf{h}(\mathbf{x})$, with respect to the feature position and camera poses, respectively [see (29)].

Applying the Sherman–Morrison–Woodbury matrix identity, we solve the system by inverting the matrix on the left-hand side of (90) as

$$\begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{B}^T & \mathbf{C} \end{bmatrix}^{-1} = \begin{bmatrix} \boldsymbol{\Upsilon}_1 & \boldsymbol{\Upsilon}_2 \\ \boldsymbol{\Upsilon}_2^T & \boldsymbol{\Upsilon}_4 \end{bmatrix} \quad (92)$$

where

$$\boldsymbol{\Upsilon}_1 = \left(\mathbf{A} - \mathbf{B} \mathbf{C}^{-1} \mathbf{B}^T \right)^{-1}$$
$$= \mathbf{P}_{ss} - \mathbf{P}_{ss} \mathbf{H}_s^T$$
$$\cdot \{ \mathbf{M}^{-1} - \mathbf{M}^{-1} \mathbf{H}_f \left(\mathbf{H}_f^T \mathbf{M}^{-1} \mathbf{H}_f \right)^{-1} \mathbf{H}_f^T \mathbf{M}^{-1} \} \mathbf{H}_s \mathbf{P}_{ss} \quad (93)$$

$$\boldsymbol{\Upsilon}_2 = -\left(\mathbf{A} - \mathbf{B} \mathbf{C}^{-1} \mathbf{B}^T \right)^{-1} \mathbf{B} \mathbf{C}^{-1}$$
$$= -\mathbf{P}_{ss} \mathbf{H}_s^T \mathbf{M}^{-1} \mathbf{H}_f \left(\mathbf{H}_f^T \mathbf{M}^{-1} \mathbf{H}_f \right)^{-1} \quad (94)$$

$$\boldsymbol{\Upsilon}_4 = \mathbf{C}^{-1} \mathbf{B}^T \left(\mathbf{A} - \mathbf{B} \mathbf{C}^{-1} \mathbf{B}^T \right)^{-1} \mathbf{B} \mathbf{C}^{-1} + \mathbf{C}^{-1}$$
$$= \left(\mathbf{H}_f^T \mathbf{M}^{-1} \mathbf{H}_f \right)^{-1} \quad (95)$$

and, $\mathbf{M} = \mathbf{H}_s \mathbf{P}_{ss} \mathbf{H}_s^T + \mathbf{R}$.

[9]We employ stochastic cloning over m time steps to ensure that the cross-correlations between the camera poses are properly accounted for [46].

During each iteration, the parameter vector is updated as

$$\mathbf{x}^\oplus = \mathbf{x}^\ominus + \begin{bmatrix} \boldsymbol{\Upsilon}_1 & \boldsymbol{\Upsilon}_2 \\ \boldsymbol{\Upsilon}_2^T & \boldsymbol{\Upsilon}_4 \end{bmatrix} \begin{bmatrix} \mathbf{D} \\ \mathbf{E} \end{bmatrix} \tilde{\mathbf{z}}. \quad (96)$$

After the minimization process converges, we compute the posterior covariance of the new state (including the initialized feature) as

$$\mathbf{P}^\oplus = \begin{bmatrix} \boldsymbol{\Upsilon}_1 & \boldsymbol{\Upsilon}_2 \\ \boldsymbol{\Upsilon}_2^T & \boldsymbol{\Upsilon}_4 \end{bmatrix} \quad (97)$$

where each element is defined in (93)–(94).

APPENDIX B

In this Appendix, we explicitly derive the block elements of $\boldsymbol{\Phi}$ employed in (47)–(50) in Section IV.

1. Structure of $\boldsymbol{\Phi}_{k,1}$

As we show in [34], $\boldsymbol{\Phi}_{k,1}$ has a specific structure comprising block identity and zero matrices [see (44)], along with the following subblocks that we show in analytic form.

2. Analytic Expression of $\boldsymbol{\Phi}_{k,1}^{(1,2)}$

We begin by computing $\boldsymbol{\Phi}_{k,1}^{(1,2)}$ from its definition [see (18), (20), (41), and (42)] as follows:

$$\dot{\boldsymbol{\Phi}}_{k,1}^{(1,2)} = \mathbf{F}^{(1,:)} \boldsymbol{\Phi}_{k,1}^{(:,2)}$$
$$= -\lfloor^{I_k} \hat{\omega} \times \rfloor \boldsymbol{\Phi}_{k,1}^{(1,2)} - \mathbf{I}_3$$
$$\Leftrightarrow \dot{\boldsymbol{\Phi}}_{k,1}^{(1,2)} + \lfloor^{I_k} \hat{\omega} \times \rfloor \boldsymbol{\Phi}_{k,1}^{(1,2)} = -\mathbf{I}_3. \quad (98)$$

To solve (98), we multiply it from the left, by $\exp(\int_{t_1}^{t_k} \lfloor^{I_\tau} \hat{\boldsymbol{\omega}} \times \rfloor \mathrm{d}\tau)$ to get

$$\exp \left(\int_{t_1}^{t_k} \lfloor^{I_\tau} \hat{\boldsymbol{\omega}} \times \rfloor \mathrm{d}\tau \right) \dot{\boldsymbol{\Phi}}_{k,1}^{(1,2)}$$
$$+ \exp \left(\int_{t_1}^{t_k} \lfloor^{I_\tau} \hat{\boldsymbol{\omega}} \times \rfloor \mathrm{d}\tau \right) \lfloor^{I_k} \hat{\omega} \times \rfloor \boldsymbol{\Phi}_{k,1}^{(1,2)}$$
$$= -\exp \left(\int_{t_1}^{t_k} \lfloor^{I_\tau} \hat{\boldsymbol{\omega}} \times \rfloor \mathrm{d}\tau \right) \implies$$
$$\frac{\mathrm{d}}{\mathrm{d}t} \left[\exp \left(\int_{t_1}^{t_k} \lfloor^{I_\tau} \hat{\boldsymbol{\omega}} \times \rfloor \mathrm{d}\tau \right) \boldsymbol{\Phi}_{k,1}^{(1,2)} \right]$$
$$= -\exp \left(\int_{t_1}^{t_k} \lfloor^{I_\tau} \hat{\boldsymbol{\omega}} \times \rfloor \mathrm{d}\tau \right) \implies$$
$$\frac{\mathrm{d}}{\mathrm{d}t} \left[\mathbf{C}^T (^{I_k} q_{I_1}) \boldsymbol{\Phi}_{k,1}^{(1,2)} \right] = -\mathbf{C}^T (^{I_k} q_{I_1}). \quad (99)$$

From (99) and its initial condition $\boldsymbol{\Phi}_{1,1}^{(1,2)} = \mathbf{0}_3$

$$\boldsymbol{\Phi}_{k,1}^{(1,2)} = -\mathbf{C}(^{I_k} q_{I_1}) \int_{t_1}^{t_k} \mathbf{C}^T (^{I_\tau} q_{I_1}) \mathrm{d}\tau \quad (100)$$

$$= -\int_{t_1}^{t_k} \mathbf{C}(^{I_k} q_{I_1}) \mathbf{C}^T (^{I_\tau} q_{I_1}) \mathrm{d}\tau \quad (101)$$

$$= -\int_{t_1}^{t_k} \mathbf{C}^T(^{I_\tau}q_{I_k})\mathrm{d}\tau. \tag{102}$$

3. Analytic Expression of $\boldsymbol{\Phi}_{k,1}^{(5,1)}$

In order to compute the element $\boldsymbol{\Phi}_{k,1}^{(5,1)}$ analytically, we will require an expression for $\boldsymbol{\Phi}_{k,1}^{(3,1)}$; hence, we begin by computing $\boldsymbol{\Phi}_{k,1}^{(3,1)}$ as follows:

$$
\begin{aligned}
\dot{\boldsymbol{\Phi}}_{k,1}^{(3,1)} &= \mathbf{F}^{(3,:)}\boldsymbol{\Phi}_{k,1}^{(:,1)} \\
&= -\mathbf{C}^T(^{I_k}q_G)\lfloor^{I_k}\hat{\mathbf{a}}\times\rfloor\boldsymbol{\Phi}_{k,1}^{(1,1)} \\
&= -\lfloor\mathbf{C}^T(^{I_k}q_G)^{I_k}\hat{\mathbf{a}}\times\rfloor\mathbf{C}^T(^{I_k}q_G)\boldsymbol{\Phi}_{k,1}^{(1,1)} \\
&= -\lfloor\mathbf{C}^T(^{I_k}q_G)^{I_k}\hat{\mathbf{a}}\times\rfloor\mathbf{C}^T(^{I_k}q_G)\mathbf{C}(^{I_k}q_{I_1}) \\
&= -\lfloor^G\hat{\mathbf{a}}\times\rfloor\mathbf{C}(^Gq_{I_1}) \tag{103}
\end{aligned}
$$

where $^G\hat{\mathbf{a}} := {}^G\mathbf{a} + {}^G\mathbf{g}$ [34], and the initial condition is $\boldsymbol{\Phi}_{1,1}^{(3,1)} = \mathbf{0}_3$. Thus

$$
\begin{aligned}
\boldsymbol{\Phi}_{k,1}^{(3,1)} &= -\int_{t_1}^{t_k}\lfloor^G\hat{\mathbf{a}}(\tau)\times\rfloor\mathbf{C}(^Gq_{I_1})\mathrm{d}\tau \\
&= -\left(\int_{t_1}^{t_k}\lfloor^G\hat{\mathbf{a}}(\tau)\times\rfloor\mathrm{d}\tau\right)\mathbf{C}(^Gq_{I_1}) \\
&= -\left(\int_{t_1}^{t_k}\lfloor^G\mathbf{a}(\tau) + {}^G\mathbf{g}\times\rfloor\mathrm{d}\tau\right)\mathbf{C}(^Gq_{I_1}) \\
&= -\lfloor(^G\mathbf{v}_{I_k} - {}^G\mathbf{v}_{I_1}) + {}^G\mathbf{g}\delta t_k\times\rfloor\mathbf{C}(^Gq_{I_1}) \tag{104}
\end{aligned}
$$

where $\delta t_k = (k-1)\delta t$.

We now turn our attention to $\boldsymbol{\Phi}_{k,1}^{(5,1)}$. Using the definition in (41) and the structure of $\boldsymbol{\Phi}_{k,1}$ [see (44)], we have that

$$\dot{\boldsymbol{\Phi}}_{k,1}^{(5,1)} = \mathbf{F}^{(5,:)}\boldsymbol{\Phi}_{k,1}^{(:,1)} = \boldsymbol{\Phi}_{k,1}^{(3,1)} \tag{105}$$

with initial condition $\boldsymbol{\Phi}_{k,1}^{(5,1)} = \mathbf{0}_3$. Thus

$$
\begin{aligned}
\boldsymbol{\Phi}_{k,1}^{(5,1)} &= -\left(\int_{t_1}^{t_k}\lfloor(^G\mathbf{v}_{I_\tau} - {}^G\mathbf{v}_{I_1}) + {}^G\mathbf{g}\,(\tau - t_1)\times\rfloor\mathrm{d}\tau\right)\mathbf{C}(^Gq_{I_1}) \\
&= -\left(\lfloor^G\mathbf{p}_{I_k} - {}^G\mathbf{p}_{I_1} - {}^G\mathbf{v}_{I_1}\delta t_k + \frac{1}{2}{}^G\mathbf{g}\,\delta t_k^2\times\rfloor\right)\mathbf{C}(^Gq_{I_1}) \\
&= \lfloor^G\mathbf{p}_{I_1} + {}^G\mathbf{v}_{I_1}\delta t_k - \frac{1}{2}{}^G\mathbf{g}\,\delta t_k^2 - {}^G\mathbf{p}_{I_k}\times\rfloor\mathbf{C}(^Gq_{I_1}). \tag{106}
\end{aligned}
$$

4. Analytic Expression for $\boldsymbol{\Phi}_{k,1}^{(5,2)}$

In order to compute an analytic expression for $\boldsymbol{\Phi}_{k,1}^{(5,2)}$, we will first need to determine the element $\boldsymbol{\Phi}_{k,1}^{(3,2)}$. Specifically

$$
\begin{aligned}
\dot{\boldsymbol{\Phi}}_{k,1}^{(3,2)} &= \mathbf{F}^{(3,:)}\boldsymbol{\Phi}_{k,1}^{(:,2)} \\
&= -\mathbf{C}^T(^{I_k}q_G)\lfloor^{I_k}\hat{\mathbf{a}}\times\rfloor\boldsymbol{\Phi}_{k,1}^{(1,2)} \\
&= \mathbf{C}^T(^{I_k}q_G)\lfloor^{I_k}\hat{\mathbf{a}}\times\rfloor\int_{t_1}^{t_k}\mathbf{C}^T(^{I_\tau}q_{I_k})\mathrm{d}\tau. \tag{107}
\end{aligned}
$$

Using this expression and employing the initial condition $\boldsymbol{\Phi}_{k,1}^{(3,2)} = \mathbf{0}_3$, we obtain

$$\boldsymbol{\Phi}_{k,1}^{(3,2)} = \int_{t_1}^{t_k}\mathbf{C}^T(^{I_s}q_G)\lfloor^{I_s}\hat{\mathbf{a}}\times\rfloor\int_{t_1}^s\mathbf{C}^T(^{I_\tau}q_{I_s})\mathrm{d}\tau\mathrm{d}s. \tag{108}$$

Now that we have an expression for $\boldsymbol{\Phi}_{k,1}^{(3,2)}$, we proceed with the derivation of $\boldsymbol{\Phi}_{k,1}^{(5,2)}$, i.e.,

$$
\begin{aligned}
\dot{\boldsymbol{\Phi}}_{k,1}^{(5,2)} &= \mathbf{F}^{(5,:)}\boldsymbol{\Phi}_{k,1}^{(:,2)} = \boldsymbol{\Phi}_{k,1}^{(3,2)} \\
&= \int_{t_1}^{t_k}\mathbf{C}^T(^{I_s}q_G)\lfloor^{I_s}\hat{\mathbf{a}}\times\rfloor\int_{t_1}^s\mathbf{C}^T(^{I_\tau}q_{I_s})\mathrm{d}\tau\mathrm{d}s \tag{109}
\end{aligned}
$$

with initial condition $\boldsymbol{\Phi}_{k,1}^{(5,2)} = \mathbf{0}_3$. Thus

$$\boldsymbol{\Phi}_{k,1}^{(5,2)} = \int_{t_1}^{t_k}\int_{t_1}^\theta\mathbf{C}^T(^{I_s}q_G)\lfloor^{I_s}\hat{\mathbf{a}}\times\rfloor\int_{t_1}^s\mathbf{C}^T(^{I_\tau}q_{I_s})\mathrm{d}\tau\mathrm{d}s\mathrm{d}\theta. \tag{110}$$

5. Analytic Expression for $\boldsymbol{\Phi}_{k,1}^{(5,4)}$

In order to compute $\boldsymbol{\Phi}_{k,1}^{(5,4)}$, we must first determine $\boldsymbol{\Phi}_{k,1}^{(3,4)}$:

$$\dot{\boldsymbol{\Phi}}_{k,1}^{(3,4)} = \mathbf{F}^{(3,:)}\boldsymbol{\Phi}_{k,1}^{(:,4)} = -\mathbf{C}^T(^{I_k}q_G) \tag{111}$$

with initial condition $\boldsymbol{\Phi}_{1,1}^{(3,4)} = \mathbf{0}_3$. Employing this relationship, we obtain

$$\boldsymbol{\Phi}_{k,1}^{(3,4)} = -\int_{t_1}^{t_k}\mathbf{C}^T(^{I_\tau}q_G)\mathrm{d}\tau. \tag{112}$$

We proceed now with the computation of $\boldsymbol{\Phi}_{k,1}^{(5,4)}$, i.e.

$$
\begin{aligned}
\dot{\boldsymbol{\Phi}}_{k,1}^{(5,4)} &= \mathbf{F}^{(5,:)}\boldsymbol{\Phi}_{k,1}^{(:,4)} = \boldsymbol{\Phi}_{k,1}^{(3,4)} \\
&= -\int_{t_1}^{t_k}\mathbf{C}^T(^{I_\tau}q_G)\mathrm{d}\tau \tag{113}
\end{aligned}
$$

with initial condition $\boldsymbol{\Phi}_{k,1}^{(5,4)} = \mathbf{0}_3$. Thus

$$\boldsymbol{\Phi}_{k,1}^{(5,4)} = -\int_{t_1}^{t_k}\int_{t_1}^s\mathbf{C}^T(^{I_\tau}q_G)\mathrm{d}\tau\mathrm{d}s. \tag{114}$$

APPENDIX C

We seek $\bar{q}^\star$ as a solution, to the optimization problem

$$\underset{\bar{q}}{\text{minimize}} \quad J(\bar{q}) = \frac{1}{2}\|\bar{q} - \hat{\bar{q}}\|_2^2$$

$$\text{subject to} \quad \mathbf{C}\left(^I\hat{\bar{q}}_{G,k+1|k}\right){}^G\mathbf{g} = \mathbf{C}(\bar{q})\mathbf{C}\left(^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g}$$

$$\bar{q}^T\bar{q} = 1. \tag{115}$$

Although, $\bar{q} \in \mathbb{R}^4$, we can find a more compact representation through the feasible set of (115). Using quaternion algebra,[10]

---

[10]See [36] for the definition of the quaternion multiplication operation $\otimes$ and its associated matrices $\mathcal{L}, \mathcal{R}$.

we can write the first constraint equation as

$$\mathbf{C}\left({}^{I}\hat{\bar{q}}_{G,k+1|k}\right){}^{G}\mathbf{g} = \mathbf{C}(\bar{q})\mathbf{C}\left({}^{I}\hat{\bar{q}}_{G,k|k-1}\right){}^{G}\mathbf{g} \quad \Leftrightarrow$$

$$\begin{bmatrix} \mathbf{C}({}^{I}\hat{\bar{q}}_{G,k+1|k}){}^{G}\mathbf{g} \\ 1 \end{bmatrix} = \bar{q} \otimes \begin{bmatrix} \mathbf{C}({}^{I}\hat{\bar{q}}_{G,k|k-1}){}^{G}\mathbf{g} \\ 1 \end{bmatrix} \otimes \bar{q}^{-1} \quad (116)$$

where we have expressed the 3-D vector $\mathbf{C}({}^{I}\hat{\bar{q}}_{G,k|k-1}){}^{G}\mathbf{g}$ as a quaternion. Multiplying (116) from the right by $\bar{q}$, we obtain

$$\begin{bmatrix} \mathbf{C}({}^{I}\hat{\bar{q}}_{G,k+1|k}){}^{G}\mathbf{g} \\ 1 \end{bmatrix} \otimes \bar{q} = \bar{q} \otimes \begin{bmatrix} \mathbf{C}({}^{I}\hat{\bar{q}}_{G,k|k-1}){}^{G}\mathbf{g} \\ 1 \end{bmatrix} \quad \Leftrightarrow$$

$$\mathcal{L}\left(\begin{bmatrix} \mathbf{C}({}^{I}\hat{\bar{q}}_{G,k+1|k}){}^{G}\mathbf{g} \\ 1 \end{bmatrix}\right)\bar{q} = \mathcal{R}\left(\begin{bmatrix} \mathbf{C}({}^{I}\hat{\bar{q}}_{G,k|k-1}){}^{G}\mathbf{g} \\ 1 \end{bmatrix}\right)\bar{q}$$

$$(117)$$

where the matrix functions $\mathcal{L}\left(\cdot\right)$ and $\mathcal{R}\left(\cdot\right)$ denote the left- and right-hand side quaternion multiplication matrices [36, pg. 3, (8) and (10)]. Expanding (117) and collecting all the terms on the left-hand side, we obtain

$$\left(\begin{bmatrix} \mathbf{I}_3 - \lfloor\mathbf{C}({}^{I}\hat{\bar{q}}_{G,k+1|k}){}^{G}\mathbf{g}\times\rfloor & \mathbf{C}({}^{I}\hat{\bar{q}}_{G,k+1|k}){}^{G}\mathbf{g} \\ -(\mathbf{C}({}^{I}\hat{\bar{q}}_{G,k+1|k}){}^{G}\mathbf{g})^T & 1 \end{bmatrix}\right.$$

$$\left. - \begin{bmatrix} \mathbf{I}_3 + \lfloor\mathbf{C}({}^{I}\hat{\bar{q}}_{G,k|k-1}){}^{G}\mathbf{g}\times\rfloor & \mathbf{C}({}^{I}\hat{\bar{q}}_{G,k|k-1}){}^{G}\mathbf{g} \\ -(\mathbf{C}({}^{I}\hat{\bar{q}}_{G,k|k-1}){}^{G}\mathbf{g})^T & 1 \end{bmatrix}\right)\bar{q} = 0.$$

$$(118)$$

In order to simplify our notation as we solve for $\bar{q}$ in this expression, we define the following quantities:

$$\bar{q} = \begin{bmatrix} \bar{q}^{(1:3)} \\ \bar{q}^{(4)} \end{bmatrix} \quad (119)$$

$$\mathbf{u}_1 = \mathbf{C}({}^{I}\hat{\bar{q}}_{G,k+1|k}){}^{G}\mathbf{g} \quad (120)$$

$$\mathbf{u}_2 = \mathbf{C}({}^{I}\hat{\bar{q}}_{G,k|k-1}){}^{G}\mathbf{g}. \quad (121)$$

Using this notation, we can write (118) as

$$\underbrace{\begin{bmatrix} -\lfloor\mathbf{u}_1 + \mathbf{u}_2\times\rfloor & (\mathbf{u}_1 - \mathbf{u}_2) \\ -(\mathbf{u}_1 - \mathbf{u}_2)^T & 0 \end{bmatrix}}_{\mathcal{S}} \begin{bmatrix} \bar{q}^{(1:3)} \\ \bar{q}^{(4)} \end{bmatrix} = \mathbf{0}_{4\times 1}. \quad (122)$$

The matrix $\mathcal{S}$ is of rank two, and hence, we can express $\bar{q}$ as a linear combination of any two vectors that span its nullspace. To see this fact, we first note that $\mathcal{S}$ is at least rank two since its last two columns are independent. To show it is exactly two, we need only to find two independent nonzero vectors that belong to its right nullspace. Two such vectors are

$$\xi_1 = \begin{bmatrix} \mathbf{u}_1 + \mathbf{u}_2 \\ 0 \end{bmatrix}, \xi_2 = \begin{bmatrix} \lfloor\mathbf{u}_1 - \mathbf{u}_2\times\rfloor(\mathbf{u}_1 + \mathbf{u}_2) \\ (\mathbf{u}_1 + \mathbf{u}_2)^T(\mathbf{u}_1 + \mathbf{u}_2) \end{bmatrix}. \quad (123)$$

Furthermore, it can be shown by direct computation that $\xi_1^T\xi_2 = 0$. Hence, ξ_1 and ξ_2 are independent.

We specify the unit vectors $\xi_1^o = \frac{1}{\|\xi_1\|_2}\xi_1, \xi_2^o = \frac{1}{\|\xi_2\|_2}\xi_2$, and we note that any $\bar{q}$ that satisfies $\mathcal{S}\bar{q} = 0$, is of the following form:

$$\bar{q} = x\xi_1^o + y\xi_2^o, \ \forall(x,y) \in \mathbb{R}, \ \text{s.t. } x^2 + y^2 = 1. \quad (124)$$

Based on this expression, we return to the original cost function, and substitute the relationship for $\bar{q}$, i.e.,

$$J(\bar{q}) = J(x,y) = \frac{1}{2}\|x\xi_1^o + y\xi_2^o - \hat{\bar{q}}\|_2^2 \quad (125)$$

$$= 1 - x\hat{\bar{q}}^T\xi_1^o - y\hat{\bar{q}}^T\xi_2^o. \quad (126)$$

Hence, solving the optimization problem (115) is equivalent to solving

$$\underset{(x,y)\in\mathbb{R}}{\text{minimize}} \quad J(x,y) = 1 - x\hat{\bar{q}}^T\xi_1^o - y\hat{\bar{q}}^T\xi_2^o \quad (127)$$

$$\text{subject to } x^2 + y^2 = 1 \quad (128)$$

which can be computed in closed form as

$$x^\star = \frac{\hat{\bar{q}}^T\xi_1^o}{\sqrt{(\hat{\bar{q}}^T\xi_1^o)^2 + (\hat{\bar{q}}^T\xi_2^o)^2}} \quad (129)$$

$$y^\star = \frac{\hat{\bar{q}}^T\xi_2^o}{\sqrt{(\hat{\bar{q}}^T\xi_1^o)^2 + (\hat{\bar{q}}^T\xi_2^o)^2}}. \quad (130)$$

Finally, we recover the optimal quaternion as

$$\bar{q}^\star = x^\star\xi_1^o + y^\star\xi_2^o. \quad (131)$$

REFERENCES

[1] P. Corke, J. Lobo, and J. Dias, "An introduction to inertial and visual sensing," *Int. J. Robot. Res.*, vol. 26, no. 6, pp. 519–535, Jun. 2007.

[2] J. A. Hesch, F. M. Mirzaei, G. L. Mariottini, and S. I. Roumeliotis, "A Laser-aided Inertial Navigation System (L-INS) for human localization in unknown indoor environments," in *Proc. IEEE Int. Conf. Robot. Autom.*, Anchorage, AK, USA, May 3–8, 2010, pp. 5376–5382.

[3] S. Shen, N. Michael, and V. Kumar, "Autonomous multi-floor indoor navigation with a computationally constrained MAV," in *Proc. IEEE Int. Conf. Robot. Autom.*, Shanghai, China, May 9–13, 2011, pp. 20–25.

[4] B. Williams, N. Hudson, B. Tweddle, R. Brockers, and L. Matthies, "Feature and pose constrained visual aided inertial navigation for computationally constrained aerial vehicles," in *Proc. IEEE Int. Conf. Robot. Autom.*, Shanghai, China, May 9–13, 2011, pp. 431–438.

[5] S. Weiss and R. Siegwart, "Real-time metric state estimation for modular vision-inertial systems," in *Proc. IEEE Int. Conf. Robot. Autom.*, Shanghai, China, May 9–13, 2011, pp. 4531–4537.

[6] S. Weiss, D. Scaramuzza, and R. Siegwart, "Monocular-SLAM-based navigation for autonomous micro helicopters in GPS-denied environments," *J. Field Robot.*, vol. 28, no. 6, pp. 854–874, Nov./Dec. 2011.

[7] S. Weiss, M. W. Achtelik, M. Chli, and R. Siegwart, "Versatile distributed pose estimation and sensor self-calibration for an autonomous MAV," in *Proc. IEEE Int. Conf. Robot. Autom.*, St. Paul, MN, USA, May 14–18, 2012, pp. 957–964.

[8] J. Kim and S. Sukkarieh, "Real-time implementation of airborne inertial-SLAM," *Robot. Auton. Syst.*, vol. 55, no. 1, pp. 62–71, Jan. 2007.

[9] A. I. Mourikis and S. I. Roumeliotis, "A dual-layer estimator architecture for long-term localization," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. Workshop*, Anchorage, AK, USA, Jun. 2008, pp. 1–8.

[10] M. Bryson and S. Sukkarieh, "Observability analysis and active control for airborne SLAM," *IEEE Trans. Aerosp. Electron. Syst.*, vol. 44, no. 1, pp. 261–280, Jan. 2008.

[11] A. I. Mourikis, N. Trawny, S. I. Roumeliotis, A. E. Johnson, A. Ansar, and L. Matthies, "Vision-aided inertial navigation for spacecraft entry, descent, and landing," *IEEE Trans. Robot.*, vol. 25, no. 2, pp. 264–280, Apr. 2009.

[12] S. Ebcin and M. Veth, "Tightly-coupled image-aided inertial navigation using the unscented Kalman filter," Air Force Inst. Technol., Dayton, OH, USA, Tech. Rep., 2007.

[13] D. W. Strelow, "Motion estimation from image and inertial measurements" Ph.D. dissertation, Carnegie Mellon Univ., Pittsburgh, PA, USA, Nov. 2004.

[14] J. Durrie, T. Gerritsen, E. W. Frew, and S. Pledgie, "Vision-aided inertial navigation on an uncertain map using a particle filter," in *Proc. IEEE Int. Conf. Robot. Autom.*, Kobe, Japan, May 12–17, 2009, pp. 4189–4194.

[15] T. Yap Jr., M. Li, A. I. Mourikis, and C. R. Shelton, "A particle filter for monocular vision-aided odometry," in *Proc. IEEE Int. Conf. Robot. Autom.*, Shanghai, China, May 9–13, 2011, pp. 5663–5669.

[16] E. S. Jones and S. Soatto, "Visual-inertial navigation, mapping and localization: A scalable real-time causal approach," *Int. J. Robot. Res.*, vol. 30, no. 4, pp. 407–430, Apr. 2011.

[17] T. Lupton and S. Sukkarieh, "Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions," *IEEE Trans. Robot.*, vol. 28, no. 1, pp. 61–76, Feb. 2012.

[18] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation With Applications to Tracking and Navigation*. New York, NY, USA: Wiley, 2001.

[19] D. G. Kottas, J. A. Hesch, S. L. Bowman, and S. I. Roumeliotis, "On the consistency of vision-aided inertial navigation," in *Proc. Nat. Acad. Eng.: German-Amer. Front. Eng. Symp.*, Potsdam, Germany, Mar. 28–31, 2012.

[20] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis, "Towards consistent vision-aided inertial navigation," in *Proc. Int. Workshop Algo. Found. Robot.*, Cambridge, MA, USA, Jun. 13–15, 2012.

[21] D. G. Kottas, J. A. Hesch, S. L. Bowman, and S. I. Roumeliotis, "On the consistency of vision-aided inertial navigation," presented at the Int. Symp. Exp. Robot., Quebec City, QC, Canada, Jun. 17–21, 2012.

[22] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proc. IEEE Int. Conf. Robot. Autom.*, Rome, Italy, Apr. 10–14, 2007, pp. 3565–3572.

[23] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "A first-estimates Jacobian EKF for improving SLAM consistency," in *Proc. Int. Symp. Exp. Robot.*, Athens, Greece, Jul. 14–17, 2008, pp. 373–382.

[24] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "Observability-based rules for designing consistent EKF SLAM estimators," *Int. J. Robot. Res.*, vol. 29, no. 5, pp. 502–528, Apr. 2010.

[25] G. P. Huang, N. Trawny, A. I. Mourikis, and S. I. Roumeliotis, "Observability-based consistent EKF estimators for multi-robot cooperative localization," *Auton. Robots*, vol. 30, no. 1, pp. 99–122, Jan. 2011.

[26] M. Li and A. I. Mourikis, "Improving the accuracy of EKF-based visual-inertial odometry," in *Proc. IEEE Int. Conf. Robot. Autom.*, St. Paul, MN, USA, May 14–18, 2012, pp. 828–835.

[27] F. M. Mirzaei and S. I. Roumeliotis, "A Kalman filter-based algorithm for IMU-camera calibration: Observability analysis and performance evaluation," *IEEE Trans. Robot.*, vol. 24, no. 5, pp. 1143–1156, Oct. 2008.

[28] J. Kelly and G. S. Sukhatme, "Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration," *Int. J. Robot. Res.*, vol. 30, no. 1, pp. 56–79, Jan. 2011.

[29] R. Hermann and A. Krener, "Nonlinear controllability and observability," *IEEE Trans. Automat. Control*, vol. 22, no. 5, pp. 728–740, Oct. 1977.

[30] A. Isidori, *Nonlinear Control Systems*. New York, NY, USA: Springer-Verlag, 1989.

[31] A. Martinelli, "Vision and IMU data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination," *IEEE Trans. Robot.*, vol. 28, no. 1, pp. 44–60, Feb. 2012.

[32] B. Lucas and T. Kanade, "An iterative image registration technique with an application to stereo vision," in *Proc. Int. Joint Conf. Artif. Intell.*, Vancouver, B.C., Canada, Aug. 24–28, 1981, pp. 674–679.

[33] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. J. Comput. Vis.*, vol. 60, no. 2, pp. 91–110, Nov. 2004.

[34] J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis. (2012, Feb.). "Observability-constrained vision-aided inertial navigation," Univ. Minnesota, Dept. Comp. Sci. Eng., MARS Lab., Tech. Rep. 2012-001, [Online]. Available: http://www-users.cs.umn.edu/~joel/_files/Joel_Hesch_TR12.pdf

[35] A. Chatfield, *Fundamentals of High Accuracy Inertial Navigation*. Washington, DC, USA: Amer. Inst. Aero. Astro., 1997.

[36] N. Trawny and S. I. Roumeliotis, "Indirect Kalman filter for 3D attitude estimation," Dept. Comp. Sci. Eng., MARS Lab., Univ. Minnesota, Minneapolis, MN, USA, Tech. Rep. 2005-002, Mar. 2005

[37] J.-Y. Bouguet. (2006). "Camera calibration toolbox for MATLAB," [Online]. Available:http://www.vision.caltech.edu/bouguetj/calibdoc/

[38] B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon, "Bundle adjustment—A modern synthesis," in *Vision Algorithms: Theory and Practice*, ser. Lecture Notes in Computer Science, B. Triggs, A. Zisserman, and R. Szeliski, Eds.. vol. 1883, New York, NY, USA: Springer-Verlag, 2000, pp. 298–372.

[39] Z. Chen, K. Jiang, and J. C. Hung, "Local observability matrix and its application to observability analyses," in *Proc. Annu. Conf. IEEE Ind. Electron. Soc.*, Pacific Grove, CA, USA, Nov. 27–30, 1990, pp. 100–103.

[40] P. S. Maybeck, *Stochastic Models, Estimation, and Control*. vol. I, New York, NY, USA: Academic, 1979.

[41] J. A. Hesch, "Consistency analysis and improvement for vision-aided inertial navigation" Ph.D. dissertation, Univ. Minnesota, Minneapolis, MN, USA, Jan. 2013.

[42] J. Shi and C. Tomasi, "Good features to track," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Washington, DC, USA, Jun. 27–Jul. 2, 1994, pp. 593–600.

[43] D. Nistér, "An efficient solution to the five-point relative pose problem," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, Madison, WI, USA, Jun. 16–22, 2003, pp. 195–202.

[44] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*. Cambridge, U.K.: Cambridge Univ. Press, 2000, ISBN: 0521623049.

[45] D. Nistér and H. Stewénius, "Scalable recognition with a vocabulary tree," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, New York, NY, USA, Jun. 17–22, 2006, pp. 2161–2168.

[46] S. I. Roumeliotis and J. W. Burdick, "Stochastic cloning: A generalized framework for processing relative state measurements," in *Proc. IEEE Int. Conf. Robot. Autom.*, Washington, DC, USA, May 11–15, 2002, pp. 1788–1795.

[47] L Kneip, M Chli, and R Siegwart, "Robust real-time visual odometry with a single camera and an IMU," in *Proc. British Mach. Vis. Conf. (BMVC)*, Dundee, Scotland, Aug. 2011.

Joel A. Hesch (S'13) received the B.Eng. degree in computer engineering in 2004 and the M.S. degree in computer science in 2008 from the College of Science and Engineering, University of Minnesota (UMN), Minneapolis, MN, USA, where he is currently working toward the Ph.D. degree with the Department of Computer Science and Engineering. His research interests include localization and navigation of autonomous vehicles and sensor fusion.

Mr. Hesch received the 2011 UMN Doctoral Dissertation Fellowship. In 2010, he was selected as a Honeywell International Innovator Scholar. He also received the National Institutes of Health Neuro-physical-computational Sciences Training Fellowship (2007–2009) in Computational Neuroscience. In addition, he received the 2007, CSE Excellence in Research Award.

Dimitrios G. Kottas (S'12) received the Diploma degree in electrical and computer engineering from the University of Patras, Patra, Greece, in 2010. He is currently working toward the Ph.D. degree with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN, USA.

His research interests include sensor calibration and vision-aided inertial navigation.
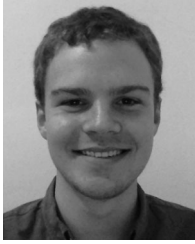
Sean L. Bowman (S'12) received the B.Comp.E. degree in 2012 from the College of Science and Engineering, University of Minnesota, Minneapolis, MN, USA. He is currently working toward the Ph.D. degree with the GRASP Lab., University of Pennsylvania, Philadelphia, PA, USA.

Stergios I. Roumeliotis (SM'12) received the Diploma degree in electrical engineering from the National Technical University of Athens, Athens, Greece, in 1995 and the M.S. and Ph.D. degrees in electrical engineering from the University of Southern California, CA, USA, in 1999 and 2000, respectively. From 2000 to 2002, he was a Postdoctoral Fellow with the California Institute of Technology, Pasadena, CA. Between 2002 and 2013, he was first an Assistant and then an Associate Professor with the Department of Computer Science and Engineering, University of Minnesota, Minneapolis, MN, USA, where he is currently a Professor. Since 2009, he has been an Associate Director for Research, of the Digital Technology Center, University of Minnesota. His research interests include distributed estimation under processing and communication constraints, active sensing for reconfigurable networks of sensors, and vision-aided inertial navigation for space, aerial, and ground vehicles, as well as mobile devices.

Dr. Roumeliotis received the Guillermo E. Borja Award (2009), the National Science Foundation (NSF) Presidential Early Career Award for Scientists and Engineers (PECASE) (2008), the NSF CAREER award (2006), the McKnight Land-Grant Professorship award (2006-08), the ICRA Best Reviewer Award (2006) and he is the corecipient of the One NASA Peer Award (2006) and the One NASA Center Best Award (2006). Papers he has coauthored have received the King-Sun Fu Best Paper Award from the IEEE TRANSACTIONS ON ROBOTICS (2009), the Robotics Society of Japan Best Journal Paper Award (2007), the ICASSP Best Student Paper award (2006), the NASA Tech Briefs Award (2004), and three of them were Finalists for the RSS Best Paper Award (2009), the ICRA Best Student Paper Award (2009), and the IROS Best Paper Award (2006). He served as an Associate Editor for the IEEE TRANSACTIONS ON ROBOTICS between 2006 and 2010.

阅读此文的还阅读了：

1. Vision-Aided Inertial Navigation for Flight Control

2. 面向LBS的机器视觉协助的智能手机惯性导航研究

3. 一种视觉辅助的惯性导航系统动基座初始对准方法

4. Dispersion sensitivity analysis & consistency improvement of APFSDS

5. 惯性导航技术浅析

6. Heading stabilization for aided inertial navigation systems

7. 基于多状态约束的视觉/惯性组合导航算法研究

8. Inertial Navigation System Aided by GPS and Selective Frequency Contents of Vector Measurements

9. 惯性/视觉组合导航系统的研究

10. Integrated Estimation and Control Approach for Vision Aided Inertial Navigation

11. LASER- Aided Inertial Navigation for Autonomous Indoor Flight

12. Improved ICCP Algorithm and Its Application in Gravity Matching Aided Inertial Navigation System

13. Gravity Gradient Aided Inertial Navigation Reference Map Obtained by New Parker's Forward Method

14. 基于GPS惯导视觉的测姿研究

15. Vertical position aided inertial navigation system

16. 鲁棒惯性地形辅助导航算法研究

17. INERTIAL SENSOR AIDED HEADING AND POSITIONING FOR GNSS VEHICLE NAVIGATION

18. Consistency Analysis and Improvement of Vision-aided Inertial Navigation

19. An algorithm for terrain-aided inertial navigation based on nonlinear optimal filtering

20. Design and Implementation of Homography-based Vision-aided Inertial Navigation of UAVs

21. Aided inertial navigation system

22. 地磁匹配辅助惯性导航技术研究

23. Gravity aided inertial navigation system

24. INERTIAL SENSOR AIDED HEADING AND POSITIONING FOR GNSS VEHICLE NAVIGATION

25. Clustering and Discrimination Analysis of Telemetric Pulse Numbers of Inertial Navigation System