Towards Consistent Vision-aided Inertial Navigation

Joel A. Hesch, Dimitrios G. Kottas, Sean L. Bowman, and Stergios I. Roumeliotis

Abstract In this paper, we study estimator inconsistency in Vision-aided Inertial Navigation Systems (VINS) from a standpoint of system observability. We postulate that a leading cause of inconsistency is the gain of spurious information along unobservable directions, resulting in smaller uncertainties, larger estimation errors, and possibly even 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. Our analysis, along with the proposed method for reducing inconsistency, are extensively validated with simulation trials and real-world experiments.

1 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 [5]. For example, 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 two time-separated 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 3D 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 [10] or height invariance in semi-structured en-

Joel A. Hesch, Dimitrios G. Kottas, Sean L. Bowman, and Stergios I. Roumeliotis
Dept. of Comp. Sci. and Eng., University of Minnesota, Minneapolis, MN 55455, USA
e-mail: {joel|dkottas|bowman|stergios}@cs.umn.edu

vironments [30], 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 Micro Aerial Vehicles (MAVs) (e.g., [36, 37]).

Numerous VINS approaches have been presented in the literature, including methods based on the Extended Kalman Filter (EKF) [3, 17, 26], the Unscented Kalman Filter (UKF) [7], and Batch-least Squares (BLS) [32]. Non-parametric estimators, such as the Particle Filter (PF), have also been applied to visual odometry (e.g., [6, 33]). However, these have focused on the simplified problem of estimating the 2D robot pose 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 [26, 37], dealing with delayed measurements [36], increasing the accuracy of feature initialization and estimation [15], and improving the robustness to estimator initialization errors [21].

A fundamental issue that has not yet been fully addressed in the literature is how estimator inconsistency affects VINS. As defined in [1], a state estimator is consistent if the estimation errors are zero-mean and have covariance smaller than or equal to the one calculated by the filter. We analyze the structure of the true and estimated systems, and postulate that a main source of inconsistency is spurious information gained along unobservable directions of the system. Furthermore, we propose a simple, yet powerful, estimator modification that explicitly prohibits this incorrect information gain. We validate our method with Monte-Carlo simulations to show that it has increased consistency and lower errors compared to standard VINS. In addition, we demonstrate the performance of our approach experimentally to show its viability for improving VINS consistency.

The rest of this paper is organized as follows: We begin with an overview of the related work (Sect. 2). In Sect. 3, we describe the system and measurement models, followed by our analysis of VINS inconsistency in Sect. 4. The proposed estimator modification is presented in Sect. 4.1, and subsequently validated both in simulations and experimentally (Sects. 5 and 6). Finally, we provide our concluding remarks and outline our future research directions in Sect. 7.

2 Related Work

Until recently, little attention was paid within the robotics community to the effects that observability properties can have on nonlinear estimator consistency. The work by Huang et al. [11, 12, 13] was the first to identify this connection for several 2D localization problems (i.e., simultaneous localization and mapping, cooperative localization). 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, allowing the estimator to surreptitiously gain spurious information along the direction corresponding to global orientation. This increases the estimation errors while reducing the estimator uncertainty, and leads to inconsistency.

Several authors have studied the observability properties of VINS under a variety of scenarios. For the task of IMU-camera extrinsic calibration, Mirzaei and Roumeliotis [24], as well as Kelly and Sukhatme [16], have analyzed the system observability using Lie derivatives [8] to determine when the IMU-camera transformation is observable. Jones and Soatto [15] studied VINS observability by examining the indistinguishable trajectories of the system [14] under different sensor configurations (i.e., inertial only, vision only, vision and inertial). Finally, Martinelli [22] utilized the concept of continuous symmetries to show that the IMU biases, 3D velocity, and absolute roll and pitch angles, are observable for VINS.

VINS inconsistency was recently addressed by Li and Mourikis [18]. Specifically, they studied the link between the VINS observability properties and estimator inconsistency for the bias-free case, and leveraged the First-Estimates Jacobian (FEJ) methodology of [11] to mitigate inconsistency in Visual-Inertial Odometry (VIO). In contrast to their work, our approach has the advantage that any linearization method can be employed (e.g., computing Jacobians analytically, numerically, or using sample points) by the estimator. Additionally, we show that our approach is flexible enough to be applied in a variety of VINS problems such as VIO or Simultaneous Localization and Mapping (SLAM).

Specifically, we leverage the key result of the existing VINS observability analysis, i.e., that the VINS model has four unobservable degrees of freedom, corresponding to three-d.o.f. global translations and one-d.o.f. global rotation about the gravity vector. Due to linearization errors, the number of unobservable directions is reduced in a standard EKF-based VINS approach, allowing the estimator to gain spurious information and leading to inconsistency. What we present is a significant, nontrivial extension of our previous work on mitigating inconsistency in 2D robot localization problems [12]. This is due in part to the higher-dimensional state of the 3D VINS system as compared to 2D localization (15 elements vs. 3), as well as more complex motion and measurement models. Furthermore, the proposed solution for reducing estimator inconsistency is general, and can be directly applied in a variety of linearized estimation frameworks such as the EKF and UKF.

3 VINS Estimator Description

We begin with an overview of the propagation and measurement models which govern the VINS system. We adopt the EKF as our framework for fusing the camera and IMU measurements to estimate the state of the system including the pose, velocity, and IMU biases, as well as the 3D 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 [20]), but are not visually distinctive. OFs are efficiently used to estimate the motion of the camera, but they are not included in the state vector. The second are Distinguishable Features (DFs), which are typically much fewer in number, and can be reliably redetected when revisiting an area (e.g., SIFT keys [19]). The 3D coordinates of the DFs are estimated to construct a map of the area.

3.1 System State and Propagation Model

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

$$\mathbf{x} = \left[{}^I\bar{q}_G^T \; \mathbf{b}_g^T \; {}^G\mathbf{v}_I^T \; \mathbf{b}_a^T \; {}^G\mathbf{p}_I^T \mid {}^G\mathbf{f}_1^T \cdots {}^G\mathbf{f}_N^T\right]^T = \left[\mathbf{x}_s^T \mid \mathbf{x}_m^T\right]^T , \qquad (1)$$

where $\mathbf{x}_s(t)$ is the 16×1 sensor platform state, and $\mathbf{x}_m(t)$ is the $3N\times 1$ state of the map. The first component of the sensor platform state is ${}^I\bar{q}_G(t)$ which is the unit quaternion representing 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 $\{G\}$, 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 $\mathbf{b}_a(t)$, affecting 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, $\mathbf{x}_m$, comprises N DFs, ${}^G\mathbf{f}_i$, $i=1,\ldots,N$, and grows as new DFs are observed [9]. However, we do not store OFs in the map. Instead, all OFs are processed and marginalized on-the-fly using the MSC-KF approach [25] (see Sect. 3.2). With the state of the system now defined, we turn our attention to the continuous-time kinematic model which governs the time evolution of the system state.

3.1.1 Continuous-time model

The system model describing the time evolution of the state is (see [4, 34]):

$$ {}^I\dot{\bar{q}}_G(t) = \frac{1}{2}\mathit{\Omega}(\omega(t))\,{}^I\bar{q}_G(t) \;\;,\;\; {}^G\dot{\mathbf{p}}_I(t) = {}^G\mathbf{v}_I(t) \;\;,\;\; {}^G\dot{\mathbf{v}}_I(t) = {}^G\mathbf{a}_I(t) \qquad (2)$$

$$ \dot{\mathbf{b}}_g(t) = \mathbf{n}_{wg}(t) \;\;,\;\; \dot{\mathbf{b}}_a(t) = \mathbf{n}_{wa}(t) \;\;,\;\; {}^G\dot{\mathbf{f}}_i(t) = \mathbf{0}_{3\times 1} \;,\; i=1,\ldots,N. \qquad (3)$$

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

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

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

$$\omega_m(t) = \omega(t) + \mathbf{b}_g(t) + \mathbf{n}_g(t) \tag{4}$$

$$\mathbf{a}_m(t) = \mathbf{C}({}^I\bar{q}_G(t))\,({}^G\mathbf{a}(t) - {}^G\mathbf{g}) + \mathbf{b}_a(t) + \mathbf{n}_a(t), \tag{5}$$

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 (3)].

Linearizing at the current estimates and applying the expectation operator on both sides of (2)-(3), we obtain the state estimate propagation model

$${}^I\dot{\hat{\bar{q}}}_G(t) = \frac{1}{2}\Omega(\hat{\omega}(t))\,{}^I\hat{\bar{q}}_G(t) \ , \ \ {}^G\dot{\hat{\mathbf{p}}}_I(t) = {}^G\hat{\mathbf{v}}_I(t) \ , \ \ {}^G\dot{\hat{\mathbf{v}}}_I(t) = \mathbf{C}^T({}^I\hat{\bar{q}}_G(t))\,\hat{\mathbf{a}}(t) + {}^G\mathbf{g} \tag{6}$$

$$\dot{\hat{\mathbf{b}}}_g(t) = \mathbf{0}_{3\times 1} \ , \ \ \dot{\hat{\mathbf{b}}}_a(t) = \mathbf{0}_{3\times 1} \ , \ \ {}^G\dot{\hat{\mathbf{f}}}_i\,(t) = \mathbf{0}_{3\times 1}\ , \ i = 1,\ldots,N, \tag{7}$$

where $\hat{\mathbf{a}}(t) = \mathbf{a}_m(t) - \hat{\mathbf{b}}_a(t)$, and $\hat{\omega}(t) = \omega_m(t) - \hat{\mathbf{b}}_g(t)$. The $(15+3N)\times 1$ error-state vector is defined as

$$\widetilde{\mathbf{x}} = \begin{bmatrix} {}^I\delta\theta_G^T \ \widetilde{\mathbf{b}}_g^T \ {}^G\widetilde{\mathbf{v}}_I^T \ \widetilde{\mathbf{b}}_a^T \ {}^G\widetilde{\mathbf{p}}_I^T \mid {}^G\widetilde{\mathbf{f}}_1^T \cdots {}^G\widetilde{\mathbf{f}}_N^T \end{bmatrix}^T = \begin{bmatrix}\widetilde{\mathbf{x}}_s^T \mid \widetilde{\mathbf{x}}_m^T\end{bmatrix}^T, \tag{8}$$

where $\widetilde{\mathbf{x}}_s(t)$ is the 15×1 error state corresponding to the sensing platform, and $\widetilde{\mathbf{x}}_m(t)$ 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 [34]. Specifically, the error between the quaternion $\bar{q}$ and its estimate $\hat{\bar{q}}$ is the 3×1 angle-error vector, $\delta\theta$, implicitly defined by the error quaternion

$$\delta\bar{q} = \bar{q}\otimes\hat{\bar{q}}^{-1} \simeq \begin{bmatrix}\tfrac{1}{2}\delta\theta^T \ 1\end{bmatrix}^T, \tag{9}$$

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

The linearized continuous-time error-state equation is

$$\dot{\widetilde{\mathbf{x}}} = \begin{bmatrix}\mathbf{F}_{s,c} & \mathbf{0}_{15\times 3N}\\ \mathbf{0}_{3N\times 15} & \mathbf{0}_{3N}\end{bmatrix}\widetilde{\mathbf{x}} + \begin{bmatrix}\mathbf{G}_{s,c}\\ \mathbf{0}_{3N\times 12}\end{bmatrix}\mathbf{n} = \mathbf{F}_c\,\widetilde{\mathbf{x}} + \mathbf{G}_c\,\mathbf{n}, \tag{10}$$

where $\mathbf{0}_{3N}$ denotes the $3N\times 3N$ matrix of zeros, $\mathbf{n} = \begin{bmatrix}\mathbf{n}_g^T \ \mathbf{n}_{wg}^T \ \mathbf{n}_a^T \ \mathbf{n}_{wa}^T\end{bmatrix}^T$ is the system noise, $\mathbf{F}_{s,c}$ is the continuous-time error-state transition matrix corresponding to the sensor platform state, and $\mathbf{G}_{s,c}$ is the continuous time input noise matrix, i.e.,

$$\mathbf{F}_{s,c} = \begin{bmatrix} -\lfloor\hat{\omega}\times\rfloor & -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3\\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3\\ -\mathbf{C}^T({}^I\hat{\bar{q}}_G)\lfloor\hat{\mathbf{a}}\times\rfloor & \mathbf{0}_3 & \mathbf{0}_3 & -\mathbf{C}^T({}^I\hat{\bar{q}}_G) & \mathbf{0}_3\\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3\\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \ , \ \ \mathbf{G}_{s,c} = \begin{bmatrix} -\mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3\\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3\\ \mathbf{0}_3 & \mathbf{0}_3 & -\mathbf{C}^T({}^I\hat{\bar{q}}_G) & \mathbf{0}_3\\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3\\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \end{bmatrix} \tag{11}$$

where $\mathbf{0}_3$ is the 3×3 matrix of zeros. The system noise is modelled as a zero-mean white Gaussian process with autocorrelation $\mathbb{E}[\mathbf{n}(t)\mathbf{n}^T(\tau)] = \mathbf{Q}_c\delta(t-\tau)$ which depends on the IMU noise characteristics and is computed off-line [34].

3.1.2 Discrete-time implementation

The IMU signals ω_m and $\mathbf{a}_m$ are sampled at a constant rate $1/\delta t$, where $\delta t \triangleq t_{k+1} - t_k$. Every time a new IMU measurement is received, the state estimate is propagated using 4th-order Runge-Kutta numerical integration of (6)–(7). In order to derive the covariance propagation equation, we evaluate the discrete-time state transition matrix, Φ_k, and the discrete-time system noise covariance matrix, $\mathbf{Q}_{d,k}$, as

$$\Phi_k = \Phi(t_{k+1}, t_k) = \exp\left(\int_{t_k}^{t_{k+1}} \mathbf{F}_c(\tau) \mathrm{d}\tau \right) \quad , \quad \mathbf{Q}_{d,k} = \int_{t_k}^{t_{k+1}} \Phi(t_{k+1}, \tau) \mathbf{G}_c \mathbf{Q}_c \mathbf{G}_c^T \Phi^T(t_{k+1}, \tau) \mathrm{d}\tau.$$

The propagated covariance is then computed as $\mathbf{P}_{k+1|k} = \Phi_k \mathbf{P}_{k|k} \Phi_k^T + \mathbf{Q}_{d,k}$.

3.2 Measurement Update Model

As the camera-IMU platform moves, the camera observes both opportunistic and distinguishable visual features. These measurements are exploited to concurrently estimate the motion of the sensing platform and the map of DFs. We distinguish three types of filter updates: (i) DF updates of features already in the map, (ii) initialization of DFs not yet in the map, and (iii) 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 point $\mathbf{f}_i$. The camera measures, $\mathbf{z}_i$, which is the perspective projection of the 3D point, ${}^I\mathbf{f}_i$, expressed in the current IMU frame $\{I\}$, onto the image plane[1], i.e.,

$$\mathbf{z}_i = \frac{1}{z} \begin{bmatrix} x \\ y \end{bmatrix} + \eta_i, \quad \text{where} \quad \begin{bmatrix} x & y & z \end{bmatrix}^T = {}^I\mathbf{f}_i = \mathbf{C}\left({}^I\bar{q}_G\right)\left({}^G\mathbf{f}_i - {}^G\mathbf{p}_I\right). \tag{12}$$

The measurement noise, η_i, is modeled as zero mean, white Gaussian with covariance $\mathbf{R}_i$. The linearized error model is $\tilde{\mathbf{z}}_i = \mathbf{z}_i - \hat{\mathbf{z}}_i \simeq \mathbf{H}_i \tilde{\mathbf{x}} + \eta_i$, where $\hat{\mathbf{z}}$ is the expected measurement computed by evaluating (12) at the current state estimate, and the measurement Jacobian, $\mathbf{H}_i$, is

$$\mathbf{H}_i = \mathbf{H}_{cam} \begin{bmatrix} \mathbf{H}_{\theta_G} & \mathbf{0}_{3\times 9} & \mathbf{H}_{\mathbf{p}_I} \mid \mathbf{0}_3 & \cdots \mathbf{H}_{\mathbf{f}_i} & \cdots & \mathbf{0}_3 \end{bmatrix} \tag{13}$$

$$\mathbf{H}_{cam} = \frac{1}{z^2} \begin{bmatrix} z & 0 & -x \\ 0 & z & -y \end{bmatrix} \quad , \quad \mathbf{H}_{\theta_G} = \left\lfloor \mathbf{C}\left({}^I\bar{q}_G\right)\left({}^G\mathbf{f}_i - {}^G\mathbf{p}_I\right) \times \right\rfloor \quad , \quad \mathbf{H}_{\mathbf{p}_I} = -\mathbf{C}\left({}^I\bar{q}_G\right) \quad , \quad \mathbf{H}_{\mathbf{f}_i} = \mathbf{C}\left({}^I\bar{q}_G\right)$$

Here, $\mathbf{H}_{cam}$, is the Jacobian of the perspective projection with respect to ${}^I\mathbf{f}_i$, while $\mathbf{H}_{\theta_G}$, $\mathbf{H}_{\mathbf{p}_I}$, 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 ${}^G\mathbf{f}_i$, respectively.

[1] 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 and extrinsic camera/IMU calibration off-line [2, 24].

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 (12)-(13) to update the filter. We compute the Kalman gain, $\mathbf{K} = \mathbf{P}_{k+1|k}\mathbf{H}_i{}^T \left(\mathbf{H}_i\mathbf{P}_{k+1|k}\mathbf{H}_i{}^T + \mathbf{R}_i\right)^{-1}$, and the measurement residual $\mathbf{r}_i = \mathbf{z}_i - \hat{\mathbf{z}}_i$. Employing these quantities, we compute the EKF state and covariance update as

$$\hat{\mathbf{x}}_{k+1|k+1} = \hat{\mathbf{x}}_{k+1|k} + \mathbf{K}\mathbf{r}_i \;\;, \;\; \mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{P}_{k+1|k}\mathbf{H}_i{}^T(\mathbf{H}_i\mathbf{P}_{k+1|k}\mathbf{H}_i{}^T + \mathbf{R}_i)^{-1}\mathbf{H}_i\mathbf{P}_{k+1|k} \quad (14)$$

For previously unobserved DFs, we compute an initial estimate, along with covariance and cross-correlations by solving a bundle-adjustment over a short time window [35]. Finally, for OFs, we employ the MSC-KF approach [25] to impose a pose update constraining all the views from which the feature was seen. To accomplish this, we utilize stochastic cloning [29] over a window of m camera poses.

4 Observability-Constrained VINS

Using the VINS system model presented above, we hereafter describe how the system observability properties influence estimator consistency. When using a linearized estimator, such as the EKF, errors in linearization while evaluating the system and measurement Jacobians change the directions in which information is acquired by the estimator. If this information lies along unobservable directions, it leads to larger errors, smaller uncertainties, and inconsistency. We first analyze this issue, and subsequently, present an Observability-Constrained VINS (OC-VINS) that explicitly adheres to the observability properties of VINS.

The Observability Gramian [23] is defined as a function of the linearized measurement model, $\mathbf{H}$, and the discrete-time state transition matrix, $\boldsymbol{\Phi}$, which are in turn functions of the linearization point, $\mathbf{x}$, i.e.,

$$\mathbf{M}(\mathbf{x}) = \begin{bmatrix} \mathbf{H}_1 \\ \mathbf{H}_2\boldsymbol{\Phi}_{2,1} \\ \vdots \\ \mathbf{H}_k\boldsymbol{\Phi}_{k,1} \end{bmatrix} \quad (15)$$

where $\boldsymbol{\Phi}_{k,1} = \boldsymbol{\Phi}_{k-1}\cdots\boldsymbol{\Phi}_1$ is the state transition matrix from time step 1 to k, with $\boldsymbol{\Phi}_1 = \mathbf{I}_{15+3\times N}$. To simplify the discussion, we consider a single landmark in the state vector, and write the first block row as

$$\mathbf{H}_1 = \mathbf{H}_{cam,1}\mathbf{C}\left({}^I\bar{q}_{G,1}\right)\left[\lfloor {}^{\mathbf{G}}\mathbf{f} - {}^{\mathbf{G}}\mathbf{p}_{I,1} \times\rfloor \mathbf{C}\left({}^I\bar{q}_{G,1}\right)^T \;\; \mathbf{0}_3 \;\; \mathbf{0}_3 \;\; \mathbf{0}_3 \;\; -\mathbf{I}_3 \;\; \mathbf{I}_3\right],$$

where ${}^I\bar{q}_{G,1}$, denotes the rotation of $\{G\}$ with respect to frame $\{I\}$ at time step 1, and for the purposes of the observability analysis, all the quantities appearing in the previous expression are the true ones. As shown in [9], the k-th block row, for $k > 1$, is of the form:

$$\mathbf{H}_k\boldsymbol{\Phi}_{k,1} = \mathbf{H}_{cam,k}\mathbf{C}\left({}^I\bar{q}_{G,k}\right)\left[\lfloor {}^{\mathbf{G}}\mathbf{f} - {}^{\mathbf{G}}\mathbf{p}_{I,1} - {}^G\mathbf{v}_{I,1}\delta_{t_{k-1}} + \tfrac{1}{2}{}^G\mathbf{g}\delta^2_{t_{k-1}} \times\rfloor \mathbf{C}\left({}^I\bar{q}_{G,1}\right)^T \;\; \mathbf{D}_k \;\; -\mathbf{I}\delta_{t_{k-1}} \;\; \mathbf{E}_k \;\; -\mathbf{I}_3 \;\; \mathbf{I}_3\right],$$

where $\delta_{t_{k-1}} = (k-1)\delta t$, and $\mathbf{D}_k$ and $\mathbf{E}_k$ are both time-varying matrices. It is straightforward to verify that the right nullspace of $\mathbf{M}(\mathbf{x})$ spans four directions, i.e.,

$$\mathbf{M}(\mathbf{x})\mathbf{N}_1 = \mathbf{0} \ , \ \ \mathbf{N}_1 = \begin{bmatrix} \mathbf{0}_3 & \mathbf{C}\left({}^I\bar{q}_{G,1}\right){}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_{3\times 1} \\ \mathbf{0}_3 & -\lfloor {}^G\mathbf{v}_{I,1}\times\rfloor {}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_{3\times 1} \\ \mathbf{I}_3 & -\lfloor {}^G\mathbf{p}_{I,1}\times\rfloor {}^G\mathbf{g} \\ \mathbf{I}_3 & -\lfloor {}^G\mathbf{f}\times\rfloor {}^G\mathbf{g} \end{bmatrix} = \left[\mathbf{N}_{t,1} \mid \mathbf{N}_{r,1}\right] \tag{16}$$

where $\mathbf{N}_{t,1}$ corresponds to global translations and $\mathbf{N}_{r,1}$ corresponds to global rotations about the gravity vector.

Ideally, any estimator we employ should correspond to a system with an unobservable subspace that matches these directions, both in number and structure. However, when linearizing about the estimated state $\hat{\mathbf{x}}$, $\mathbf{M}(\hat{\mathbf{x}})$ gains rank due to errors in the state estimates across time [9]. To address this problem and ensure that (16) is satisfied for every block row of $\mathbf{M}$ when the state estimates are used for computing $\mathbf{H}_\ell$, and $\boldsymbol{\Phi}_{\ell,1}$, $\ell = 1,\ldots,k$, we must ensure that $\mathbf{H}_\ell\boldsymbol{\Phi}_{\ell,1}\mathbf{N}_1 = \mathbf{0}$, $\ell = 1,\ldots,k$.

One way to enforce this is by requiring that at each time step

$$\mathbf{N}_{\ell+1} = \boldsymbol{\Phi}_\ell\mathbf{N}_\ell \ \ , \ \ \mathbf{H}_\ell\mathbf{N}_\ell = \mathbf{0}, \ \ \ell = 1,\ldots,k \tag{17}$$

where $\mathbf{N}_\ell$, $\ell \geq 1$ is computed analytically (see (18) and [9]). This can be accomplished by appropriately modifying $\boldsymbol{\Phi}_\ell$ and $\mathbf{H}_\ell$ following the process described in the next section.

4.1 OC-VINS: Algorithm Description

Hereafter, we present our OC-VINS algorithm which enforces the observability constraints dictated by the VINS system structure. Rather than changing the linearization points explicitly (e.g., as in [11]), we maintain the nullspace, $\mathbf{N}_k$, at each time step, and use it to enforce the unobservable directions. The 15×4 nullspace block, $\mathbf{N}_k^R$, corresponding to the robot state is analytically defined as [9]:

$$\mathbf{N}_1^R = \begin{bmatrix} \mathbf{0}_3 & \mathbf{C}\left({}^I\hat{\bar{q}}_{G,1|1}\right){}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_{3\times 1} \\ \mathbf{0}_3 & -\lfloor {}^G\hat{\mathbf{v}}_{I,1|1}\times\rfloor {}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_{3\times 1} \\ \mathbf{I}_3 & -\lfloor {}^G\hat{\mathbf{p}}_{I,1|1}\times\rfloor {}^G\mathbf{g} \end{bmatrix} \ , \ \ \mathbf{N}_k^R = \begin{bmatrix} \mathbf{0}_3 & \mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_{3\times 1} \\ \mathbf{0}_3 & -\lfloor {}^G\hat{\mathbf{v}}_{I,k|k-1}\times\rfloor {}^G\mathbf{g} \\ \mathbf{0}_3 & \mathbf{0}_{3\times 1} \\ \mathbf{I}_3 & -\lfloor {}^G\hat{\mathbf{p}}_{I,k|k-1}\times\rfloor {}^G\mathbf{g} \end{bmatrix} = \left[\mathbf{N}_{t,k}^R \mid \mathbf{N}_{r,k}^R\right]. \tag{18}$$

The 3×4 nullspace block, $\mathbf{N}_\ell^f$, corresponding to the feature state, is a function of the feature estimate at time t_ℓ when it was initialized, i.e.,

$$\mathbf{N}_k^f = \left[\mathbf{I}_3 \ -\lfloor {}^G\hat{\mathbf{f}}_{\ell|\ell}\times\rfloor {}^G\mathbf{g}\right] \tag{19}$$

4.1.1 Modification of the state transition matrix Φ

During the propagation step, we must ensure that $\mathbf{N}^R_{k+1} = \Phi^R_k \mathbf{N}^R_k$, where Φ^R_k is the first 15×15 sub-block of Φ_k corresponding to the robot state. We note that the constraint on $\mathbf{N}^R_{t,k}$ is automatically satisfied by the structure of Φ^R_k [see (20) and [9]], so we focus on $\mathbf{N}^R_{r,k}$. We rewrite this equation element-wise as

$$\mathbf{N}^R_{r,k+1} = \Phi^R_k \mathbf{N}^R_{r,k} \quad \rightarrow \quad \begin{bmatrix} \mathbf{C}\left({}^I\hat{\bar{q}}_{G,k+1|k}\right){}^G\mathbf{g} \\ \mathbf{0}_{3\times 1} \\ -\lfloor {}^G\hat{\mathbf{v}}_{I,k+1|k} \times \rfloor {}^G\mathbf{g} \\ \mathbf{0}_{3\times 1} \\ -\lfloor {}^G\hat{\mathbf{p}}_{I,k+1|k} \times \rfloor {}^G\mathbf{g} \end{bmatrix} = \begin{bmatrix} \Phi_{11} & \Phi_{12} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \Phi_{31} & \Phi_{32} & \mathbf{I}_3 & \Phi_{34} & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \Phi_{51} & \Phi_{52} & \delta t \mathbf{I}_3 & \Phi_{54} & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g} \\ \mathbf{0}_{3\times 1} \\ -\lfloor {}^G\hat{\mathbf{v}}_{I,k|k-1} \times \rfloor {}^G\mathbf{g} \\ \mathbf{0}_{3\times 1} \\ -\lfloor {}^G\hat{\mathbf{p}}_{I,k|k-1} \times \rfloor {}^G\mathbf{g} \end{bmatrix}. \tag{20}$$

From the first block row we have that

$$\mathbf{C}\left({}^I\hat{\bar{q}}_{G,k+1|k}\right){}^G\mathbf{g} = \Phi_{11}\mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g} \quad \Rightarrow \Phi_{11} = \mathbf{C}\left({}^{I,k+1|k}\hat{\bar{q}}_{I,k|k-1}\right). \tag{21}$$

The requirements for the third and fifth block rows are

$$\Phi_{31}\mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g} = \lfloor {}^G\hat{\mathbf{v}}_{I,k|k-1} \times \rfloor {}^G\mathbf{g} - \lfloor {}^G\hat{\mathbf{v}}_{I,k+1|k} \times \rfloor {}^G\mathbf{g} \tag{22}$$

$$\Phi_{51}\mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g} = \delta t \lfloor {}^G\hat{\mathbf{v}}_{I,k|k-1} \times \rfloor {}^G\mathbf{g} + \lfloor {}^G\hat{\mathbf{p}}_{I,k|k-1} \times \rfloor {}^G\mathbf{g} - \lfloor {}^G\hat{\mathbf{p}}_{I,k+1|k} \times \rfloor {}^G\mathbf{g} \tag{23}$$

both of which are in the form $\mathbf{Au} = \mathbf{w}$, where $\mathbf{u}$ and $\mathbf{w}$ are nullspace elements that are fixed, and we seek to find a perturbed $\mathbf{A}^*$, for $\mathbf{A} = \Phi_{31}$ and $\mathbf{A} = \Phi_{51}$ that fulfills the constraint. To compute the minimum perturbation, $\mathbf{A}^*$, of $\mathbf{A}$, we formulate the following minimization problem

$$\min_{\mathbf{A}^*} ||\mathbf{A}^* - \mathbf{A}||^2_{\mathscr{F}} \;\; , \;\; \text{s.t. } \mathbf{A}^*\mathbf{u} = \mathbf{w} \tag{24}$$

where $||\cdot||_{\mathscr{F}}$ denotes the Frobenius matrix norm. After employing the method of Lagrange multipliers, and solving the corresponding KKT optimality conditions, the optimal $\mathbf{A}^*$ that fulfills (24) is $\mathbf{A}^* = \mathbf{A} - (\mathbf{Au} - \mathbf{w})(\mathbf{u}^T\mathbf{u})^{-1}\mathbf{u}^T$.

We compute the modified Φ_{11} from (21), and Φ_{31} and Φ_{51} from (24) and construct the constrained discrete-time state transition matrix. We then proceed with covariance propagation (see Sect. 3.1).

4.1.2 Modification of H

During each update step, we seek to satisfy $\mathbf{H}_k\mathbf{N}_k = \mathbf{0}$. Based on (13), we can write this relationship *per feature* as

$$\mathbf{H}_{cam}\left[\mathbf{H}_{\theta_G}\ \mathbf{0}_{3\times 9}\ \mathbf{H}_{\mathbf{p}_I}\ \middle|\ \mathbf{H}_{\mathbf{f}}\right]\begin{bmatrix}\mathbf{0}_3 & \mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g}\\ \mathbf{0}_3 & \mathbf{0}_{3\times 1}\\ \mathbf{0}_3 & -\lfloor{}^G\hat{\mathbf{v}}_{I,k|k-1}\times\rfloor{}^G\mathbf{g}\\ \mathbf{0}_3 & \mathbf{0}_{3\times 1}\\ \mathbf{I}_3 & -\lfloor{}^G\hat{\mathbf{p}}_{I,k|k-1}\times\rfloor{}^G\mathbf{g}\\ \mathbf{I}_3 & -\lfloor{}^G\hat{\mathbf{f}}_{\ell|\ell}\times\rfloor{}^G\mathbf{g}\end{bmatrix}=\mathbf{0}. \tag{25}$$

The first block column of (25) requires that $\mathbf{H}_{\mathbf{f}} = -\mathbf{H}_{\mathbf{p}_I}$. Hence, we rewrite the second block column of (25) as

$$\mathbf{H}_{cam}\left[\mathbf{H}_{\theta_G}\ \mathbf{H}_{\mathbf{p}_I}\right]\begin{bmatrix}\mathbf{C}\left({}^I\hat{\bar{q}}_{G,k|k-1}\right){}^G\mathbf{g}\\ \left(\lfloor{}^G\hat{\mathbf{f}}_{\ell|\ell}\times\rfloor-\lfloor{}^G\hat{\mathbf{p}}_{I,k|k-1}\times\rfloor\right){}^G\mathbf{g}\end{bmatrix}=\mathbf{0}. \tag{26}$$

This is a constraint of the form $\mathbf{A}\mathbf{u} = \mathbf{0}$, where $\mathbf{u}$ is a fixed quantity determined by elements in the nullspace, and $\mathbf{A}$ comprises elements of the measurement Jacobian. We compute the optimal $\mathbf{A}^*$ that satisfies this relationship using the solution to (24). After computing the optimal $\mathbf{A}^*$, we recover the Jacobian as

$$\mathbf{H}_{cam}\mathbf{H}_{\theta_G} = \mathbf{A}'_{1:2,1:3}\ ,\ \ \mathbf{H}_{cam}\mathbf{H}_{\mathbf{p}_I} = \mathbf{A}'_{1:2,4:6}\ ,\ \ \mathbf{H}_{cam}\mathbf{H}_{\mathbf{f}} = -\mathbf{A}'_{1:2,4:6} \tag{27}$$

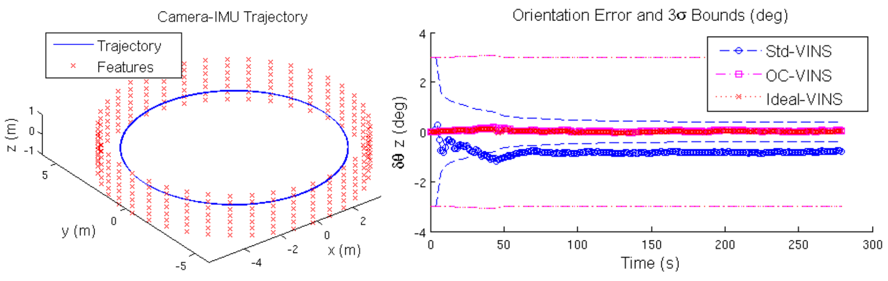
where the subscripts (i:j, m:n) denote the submatrix spanning rows i to j, and columns m to n. After computing the modified measurement Jacobian, we proceed with the filter update as described in Sect. 3.2.

5 Simulations

We conducted Monte-Carlo simulations to evaluate the impact of the proposed Observability-Constrained VINS (OC-VINS) method on estimator consistency. We compared its performance to the standard VINS (Std-VINS), as well as the ideal VINS that linearizes about the true state[2]. Specifically, we computed the Root Mean Squared Error (RMSE) and Normalized Estimation Error Squared (NEES) over 100 trials in which the camera-IMU platform traversed a circular trajectory of radius 5 m at an average velocity of 60 cm/s.[3] The camera observed visual features distributed on the interior wall of a circumscribing cylinder with radius 6 m and height 2 m (see Fig. 1a). The effect of inconsistency during a single run is depicted in Fig. 1b. The error and corresponding 3σ bounds of uncertainty are plotted for the rotation about the gravity vector. It is clear that the Std-VINS gains spurious information, hence reducing its 3σ bounds of uncertainty, while the Ideal-VINS and the OC-VINS do not. The Std-VINS becomes inconsistent on this run as the orientation errors fall outside of the uncertainty bounds, while both the Ideal-VINS and the OC-VINS remain consistent. Figure 2 displays the RMSE and NEES, in which we observe that the OC-VINS obtains orientation accuracy and consistency levels similar to the

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

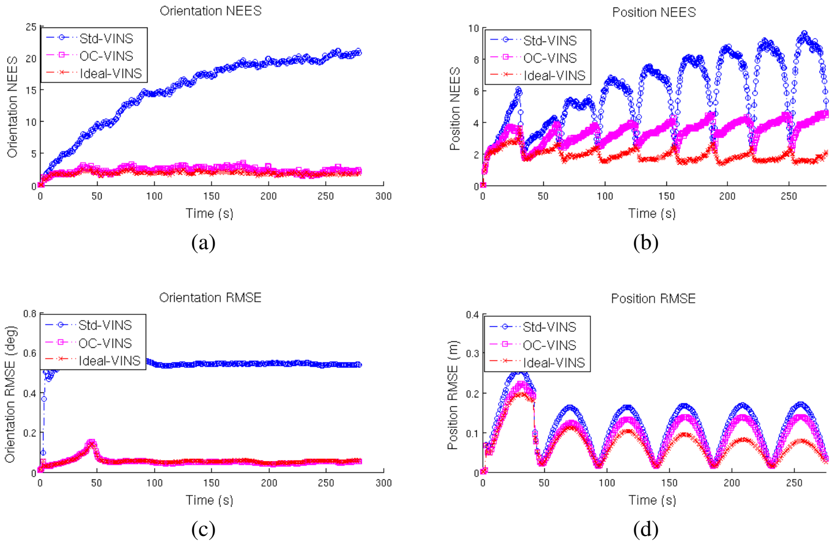
[3] The camera had 45 deg field of view, with $\sigma_{px} = 1px$, while the IMU was modeled with MEMS quality sensors.



(a) (b)

Fig. 1: (a) Camera-IMU trajectory and 3D features. (b) Error and 3σ bounds for the rotation about the gravity vector, plotted for a single run.

ideal, while significantly outperforming Std-VINS. Similarly, the OC-VINS obtains better positioning accuracy compared to Std-VINS.



(a) (b)

(c) (d)

Fig. 2: The RMSE and NEES errors for position and orientation plotted for all three filters, averaged per time step over 100 Monte Carlo trials.

6 Experimental Results

6.1 Implementation Remarks

The image processing is separated into two components: one for extracting and tracking short-term OFs, and one for extracting DFs to use in SLAM.

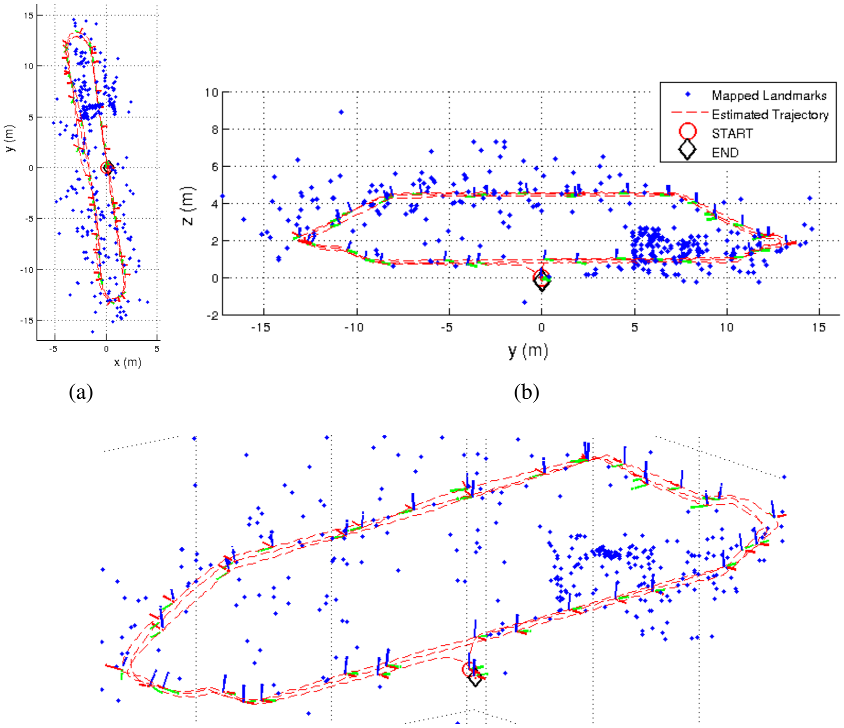
OFs are extracted from images using the Shi-Tomasi corner detector [31]. After acquiring image k, it is inserted into a sliding window buffer of m images, $\{k-m+1, k-m+2, \ldots, k\}$. We then extract features from the first image in the window and track them pairwise through the window using the KLT tracking algorithm [20]. To remove outliers from the resulting tracks, we use a two-point algorithm to find the essential matrix between successive frames. Given the filter's estimated rotation 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 traditional five-point algorithm [27] because it provides two solutions for the essential matrix rather than up to ten, and as it requires only two data points, it reaches a consensus with fewer hypotheses when used in a RANSAC framework.

The DFs are extracted using SIFT descriptors [19]. To identify global features observed from several different images, we first utilize a vocabulary tree (VT) structure for image matching [28]. 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 matching, the SIFT descriptors from each are compared to those from image k to create tentative feature correspondences. The epipolar constraint is then enforced using RANSAC and Nister's five-point algorithm [27] to eliminate outliers. It is important to note that the images used to construct the VT (off-line) are not taken along our experimental trajectory, but rather are randomly selected from a set of representative images.

At every time step, the robot poses corresponding to the last m images are kept in the state vector, as described in [29]. Upon the ending of the processing of a new image, all the OFs that first appeared at the oldest augmented robot pose, are processed following the MSC-KF approach, as discussed earlier. The frequency at which new DFs are initialized into the map is a scalable option which can be tuned according to the available computational resources, while DF updates occur at any reobservation of initialized features.

6.2 Experimental Evaluation

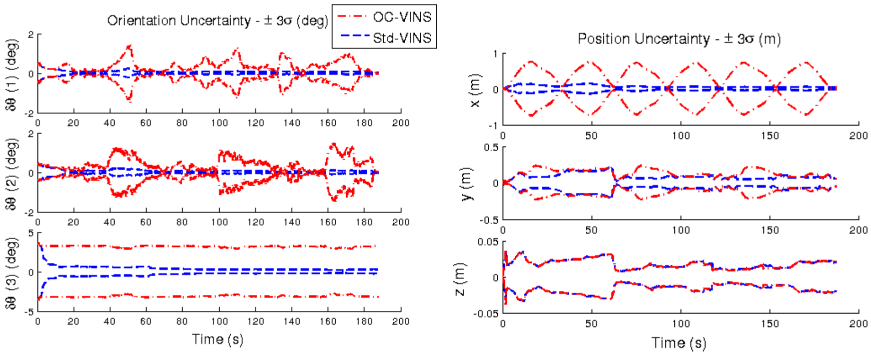
The experimental evaluation was performed with an Ascending Technologies Pelican quadrotor equipped with a PointGrey Chameleon camera, a Navchip IMU and a VersaLogic Core 2 Duo single board computer. For the purposes of this experiment, the onboard computing platform was used only for measurement logging and the quadrotor platform was simply carried along the trajectory. Note that the compu-



(a) (b)

(c)

Fig. 3: The estimated 3D trajectory over the three traversals of the two floors of the building, along with the estimated positions of the mapped landmarks. (a) projection on the x and y axis, (b) projection on the y and z axis, (c) 3D view of the overall trajectory and estimated features



(a) (b)

Fig. 4: Comparison of the estimated 3σ error bounds for attitude and position between Std-VINS and OC-VINS.

tational and sensing equipment does not exceed the weight or power limits of the quadrotor and it is still capable of flight. The platform traveled a total distance of 172.5 meters over two floors at the Walter Library at University of Minnesota. IMU signals were sampled at a frequency of 100 Hz while camera images were acquired at 7.5 Hz. The trajectory traversed in our experiment consisted of three passes over a loop that spanned two floors, so three loop-closure events occurred. 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 available DFs are initialized and the state vector is augmented with their 3D coordinates. The process of initializing DFs [9] is continued until the occurrence of the first loop closure; from that point, no new DFs are considered and the filter relies upon the re-observation of previously initialized DFs and the processing of OFs.

For both the Std-VINS and the OC-VINS, the final position error was approximately 34 cm, which is less than 0.2% of the total distance traveled (see Fig. 3). However, the estimated covariances from the Std-VINS are smaller than those from the OC-VINS (see Fig. 4). Furthermore, uncertainty estimates from the Std-VINS 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.

Figure 4(a) highlights the difference in estimated yaw uncertainty between the OC-VINS and the Std-VINS. In contrast to the OC-VINS, the Std-VINS covariance rapidly decreases, violating the observability properties of the system. Similarly, large differences can be seen in the covariance estimates for the x and y position estimates [see Fig. 4(b)]. The Std-VINS estimates a much smaller uncertainty than the OC-VINS, supporting the claim that Std-VINS tends to be inconsistent.

7 Conclusion and Future Work

In this paper, we analyzed the inconsistency of VINS from the standpoint of observability. Specifically, we showed that a standard VINS filtering approach leads to spurious information gain since it does not adhere to the unobservable directions of the true system. Furthermore, we introduced an observability-constrained VINS approach to mitigate estimator inconsistency by enforcing the unobservable directions explicitly. We presented both simulation and experimental results to support our claims and validate the proposed estimator.

In our future work, we are interested in analyzing additional sources of estimator inconsistency in VINS such as the existence of multiple local minima.

Acknowledgment

This work was supported by the University of Minnesota (UMN) through the Digital Technology Center (DTC) and the Air Force Office of Scientific Research (FA9550-10-1-0567). J. A. Hesch was supported by the UMN Doctoral Dissertation Fellowship.

References

1. Y. Bar-Shalom, X. R. Li, and T. Kirubarajan. *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, New York, NY, 2001.
2. J.-Y. Bouguet. Camera calibration toolbox for matlab, 2006.
3. M. Bryson and S. Sukkarieh. Observability analysis and active control for airborne SLAM. *IEEE Trans. on Aerospace and Electronic Systems*, 44(1):261–280, Jan. 2008.
4. A. Chatfield. *Fundamentals of high accuracy inertial navigation*. AIAA (American Institute of Aeronautics & Astronautics), 1997.
5. P. Corke, J. Lobo, and J. Dias. An introduction to inertial and visual sensing. *Int. Journal of Robotics Research*, 26(6):519–535, June 2007.
6. J. Durrie, T. Gerritsen, E. W. Frew, and S. Pledgie. Vision-aided inertial navigation on an uncertain map using a particle filter. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 4189–4194, Kobe, Japan, May 12–17, 2009.
7. S. Ebcin and M. Veth. Tightly-coupled image-aided inertial navigation using the unscented Kalman filter. Technical report, Air Force Institute of Technology, Dayton, OH, 2007.
8. R. Hermann and A. Krener. Nonlinear controllability and observability. *IEEE Trans. on Automatic Control*, 22(5):728–740, Oct. 1977.
9. J. A. Hesch, D. G. Kottas, S. L. Bowman, and S. I. Roumeliotis. Observability-constrained vision-aided inertial navigation. Technical Report 2012-001, University of Minnesota, Dept. of Comp. Sci. & Eng., MARS Lab, Feb. 2012.
10. 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. of the IEEE Int. Conf. on Robotics and Automation*, pages 5376–5382, Anchorage, AK, May 3–8, 2010.
11. G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis. A first-estimates Jacobian EKF for improving SLAM consistency. In *Proc. of the Int. Symposium on Experimental Robotics*, pages 373–382, Athens, Greece, July 14–17, 2008.
12. G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis. Observability-based rules for designing consistent EKF SLAM estimators. *Int. Journal of Robotics Research*, 29(5):502–528, Apr. 2010.
13. G. P. Huang, N. Trawny, A. I. Mourikis, and S. I. Roumeliotis. Observability-based consistent EKF estimators for multi-robot cooperative localization. *Autonomous Robots*, 30(1):99–122, Jan. 2011.
14. A. Isidori. *Nonlinear Control Systems*. Springer-Verlag, 1989.
15. E. S. Jones and S. Soatto. Visual-inertial navigation, mapping and localization: A scalable real-time causal approach. *Int. Journal of Robotics Research*, 30(4):407–430, Apr. 2011.
16. J. Kelly and G. S. Sukhatme. Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration. *Int. Journal of Robotics Research*, 30(1):56–79, Jan. 2011.
17. J. Kim and S. Sukkarieh. Real-time implementation of airborne inertial-SLAM. *Robotics and Autonomous Systems*, 55(1):62–71, Jan. 2007.
18. M. Li and A. I. Mourikis. Improving the accuracy of EKF-based visual-inertial odometry. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 828–835, Minneapolis, MN, May 14–18, 2012.

19. D. G. Lowe. Distinctive image features from scale-invariant keypoints. *Int. Journal of Computer Vision*, 60(2):91–110, Nov. 2004.
20. B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. of the Int. Joint Conf. on Artificial Intelligence*, pages 674–679, Vancouver, B.C., Canada, Aug. 24–28, 1981.
21. T. Lupton and S. Sukkarieh. Visual-inertial-aided navigation for high-dynamic motion in built environments without initial conditions. *IEEE Trans. on Robotics*, 28(1):61–76, Feb. 2012.
22. A. Martinelli. Vision and IMU data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination. *IEEE Trans. on Robotics*, 28(1):44–60, Feb. 2012.
23. P. S. Maybeck. *Stochastic models, estimation, and control*, volume I. Academic Press, New York, NY, 1979.
24. F. M. Mirzaei and S. I. Roumeliotis. A Kalman filter-based algorithm for IMU-camera calibration: Observability analysis and performance evaluation. *IEEE Trans. on Robotics*, 24(5):1143–1156, Oct. 2008.
25. A. I. Mourikis and S. I. Roumeliotis. A multi-state constraint Kalman filter for vision-aided inertial navigation. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 3565–3572, Rome, Italy, Apr. 10–14, 2007.
26. A. I. Mourikis and S. I. Roumeliotis. A dual-layer estimator architecture for long-term localization. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition Workshops*, pages 1–8, Anchorage, AK, June 2008.
27. D. Nistér. An efficient solution to the five-point relative pose problem. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 195–202, Madison, WI, June 16–22, 2003.
28. D. Nistér and H. Stewénius. Scalable recognition with a vocabulary tree. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 2161–2168, New York, NY, June 17–22, 2006.
29. S. I. Roumeliotis and J. W. Burdick. Stochastic cloning: A generalized framework for processing relative state measurements. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 1788–1795, Washington D.C., May 11-15, 2002.
30. S. Shen, N. Michael, and V. Kumar. Autonomous multi-floor indoor navigation with a computationally constrained MAV. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 20–25, Shanghai, China, May 9–13, 2011.
31. J. Shi and C. Tomasi. Good features to track. In *Proc. of the IEEE Conf. on Computer Vision and Pattern Recognition*, pages 593–600, Washington, DC, June 27–July 2, 1994.
32. D. W. Strelow. *Motion estimation from image and inertial measurements*. PhD thesis, Carnegie Mellon University, Pittsburgh, PA, Nov. 2004.
33. J. Teddy Yap, M. Li, A. I. Mourikis, and C. R. Shelton. A particle filter for monocular vision-aided odometry. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, pages 5663–5669, Shanghai, China, May 9–13, 2011.
34. N. Trawny and S. I. Roumeliotis. Indirect Kalman filter for 3D attitude estimation. Technical Report 2005-002, University of Minnesota, Dept. of Comp. Sci. & Eng., MARS Lab, Mar. 2005.
35. B. Triggs, P. McLauchlan, R. Hartley, and A. Fitzgibbon. Bundle adjustment – a modern synthesis. In B. Triggs, A. Zisserman, and R. Szeliski, editors, *Vision Algorithms: Theory and Practice*, volume 1883 of *Lecture Notes in Computer Science*, pages 298–372. Springer-Verlag, 2000.
36. S. Weiss, M. W. Achtelik, M. Chli, and R. Siegwart. Versatile distributed pose estimation and sensor self-calibration for an autonomous MAV. In *Proc. of the IEEE Int. Conf. on Robotics and Automation*, St. Paul, MN, May 14–18, 2012.
37. 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. of the IEEE Int. Conf. on Robotics and Automation*, pages 431–438, Shanghai, China, May 9–13, 2011.