Consistency of EKF-Based Visual-Inertial Odometry

Mingyang Li and Anastasios I. Mourikis
Dept. of Electrical Engineering, University of California, Riverside
E-mail: mli@ee.ucr.edu, mourikis@ee.ucr.edu

Updated: May, 2012

Abstract

In this report, we perform a rigorous analysis of EKF-based visual-inertial odometry (VIO) and present a method for improving its performance. Specifically, we examine the properties of EKF-based VIO, and show that the standard way of computing Jacobians in the filter inevitably causes inconsistency and loss of accuracy. This result is derived based on an observability analysis of the EKF's linearized system model, which proves that the yaw erroneously appears to be observable. In order to address this problem, we propose modifications to the multi-state constraint Kalman filter (MSCKF) algorithm [1], which ensure the correct observability properties without incurring additional computational cost. Extensive simulation tests and real-world experiments demonstrate that the modified MSCKF algorithm outperforms competing methods, both in terms of consistency *and* accuracy.

Contents

1 Introduction

In this work, we focus on the problem of tracking a vehicle's egomotion using a camera and an inertial measurement unit (IMU). Cameras are small and lightweight sensors, that provide very rich information about the environment. However, if *only* visual measurements are used for motion estimation, the resulting algorithms often lack robustness, due to the challenging nature of the estimation problem. Employing an IMU as an additional sensor can dramatically improve both the reliability and the accuracy of motion tracking, as demonstrated in recent work on vision-aided inertial navigation [1–4].

Our focus is on the task of estimating the pose of a vehicle moving in an unknown environment. Therefore, we do not assume that a feature map is available in advance, as in map-based localization methods (e.g., [3, 5]). Moreover, we do not aim at building such a map, as in simultaneous localization and mapping (SLAM) methods (e.g., [6, 7]). Our goal is to estimate the vehicle trajectory only, using the inertial measurements and the observations of static features that are tracked in consecutive images. This task is similar to the well-known visual odometry (VO) problem [8], with the added characteristic that an IMU is available. We thus term the approach *visual-inertial odometry* (VIO).

To date, the majority of algorithms proposed for real-time VIO are either extended Kalman filter (EKF)-based methods (e.g., [1, 2, 9]), or methods utilizing iterative minimization over a window of states (e.g., [4, 10–12]). The latter generally attain higher accuracy, as they re-linearize at each iteration to better deal with their nonlinear measurement models. However, the need for multiple iterations also incurs a higher computational cost, compared to EKF-based methods. Ideally, one would like to obtain accuracy similar to, or better than, that of an iterative-minimization algorithm, but at the computational cost of an EKF algorithm. In this paper, we show how this can be achieved.

Generally, two types of EKF algorithms can be employed for real-time VIO. On one hand, one can employ EKF-SLAM (e.g. [7, 13, 14] and references therein), in which the state vector contains the IMU state as well as feature positions. To maintain the computational cost bounded (a requirement for real-time VIO), features that leave the field of view of the camera can be removed from the state vector [14]. On the other hand, EKF algorithms exist that only maintain a sliding window of camera poses in the state vector, and use the feature observations to apply probabilistic constraints between these poses (e.g., [1, 15]). Out of this second class of methods, the multi-state constraint Kalman filter (MSCKF) [1] uses the feature measurements optimally [16], and will be our focus here.

Both EKF-based SLAM and the MSCKF use the same measurement information, and are optimal, except for the inaccuracies due to linearization. In other words, if the VIO system model was linear, then the estimation result produced by an EKF-SLAM algorithm and by the MSCKF would be identical, and equal to the optimal MAP estimate. However, in the presence of nonlinearity the MSCKF outperforms EKF-SLAM, as it does not approximate the feature's position pdf by a Gaussian. Features in the MSCKF are never included in the state vector, so this is not necessary. As a result, the MSCKF employs fewer approximations and attains higher estimation accuracy. Moveover, the MSCKF has computational complexity only *linear* in the number of features, as opposed to EKF-SLAM's cubic complexity. Thus, in this paper, we focus on improving the performance of the MSCKF, since it is a more accurate and computationally efficient approach.

By analyzing the observability properties of the linearized system model employed by the EKF, we prove that the MSCKF is *inconsistent*, i.e., that the covariance matrix of the estimation errors is larger than that computed by the filter [17, Section 5.4]. In turn, this inconsistency leads to inaccurate state updates and ultimately a loss of accuracy. We show that the root cause of this inconsistency is the way in which the Jacobians are computed in the EKF, which causes the linearized system model to have incorrect observability properties.

As a key contribution of this work, we employ these theoretical results to propose modifications to the original MSCKF algorithm that substantially improve its performance. Specifically, we here propose three key changes: First, we propose a novel *closed-form* expression for computing the elements of the IMU error-state transition matrix. This expression can be used in any case where the EKF is used for inertial navigation. Second, we adopt a different parameterization of the orientation error, and third, we propose changing the way in which the filter Jacobians are computed. Taken together, these three modifications ensure the appropriate observability properties of the linearized system model. Our simulation and experimental results in Section 6 show that the resulting algorithm is consistent, and that it attains substantially higher accuracy than the original MSCKF. More importantly, the results demonstrate that the modified MSCKF algorithm outperforms, in terms of *both* accuracy and consistency, even an iterative-minimization based fixed lag smoother, an algorithm with substantially higher computational cost.

2 Observability and EKF Consistency

Our approach is motivated by recent results in the context of 2D EKF-based SLAM [18,19]. These proved that a key factor degrading the accuracy of the EKF for 2D SLAM is a mismatch between the observability properties of the underlying nonlinear system and the linearized system-model of the EKF. To illustrate the main idea, consider a physical system described by the nonlinear model:

$$\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u}) + \mathbf{w} \tag{1}$$

$$\mathbf{z} = h(\mathbf{x}) + \mathbf{n} \tag{2}$$

where $\mathbf{x}$ is the system state, $\mathbf{u}$ is the control input, $\mathbf{z}$ is the measurement vector, and finally $\mathbf{w}$ and $\mathbf{n}$ are noise processes. To track the state vector $\mathbf{x}$ on a digital computer we must discretize the continuous-time system model shown above. Moreover, when an EKF is used for estimation, the filter equations rely on a linearized version of the discrete-time model, described by the equations:

$$\tilde{\mathbf{x}}_{k+1} = \mathbf{\Phi}_k \tilde{\mathbf{x}}_k + \mathbf{w}_{d_k} \tag{3}$$

$$\tilde{\mathbf{z}}_k = \mathbf{H}_k \tilde{\mathbf{x}}_k + \mathbf{n}_k \tag{4}$$

where $\tilde{\mathbf{x}}_k$ represents the estimation error at time step k, and $\mathbf{\Phi}_k$ and $\mathbf{H}_k$ denote the error-state transition matrix and the measurement Jacobian matrix, respectively.

Since the EKF equations (e.g., covariance propagation and update, gain computation) are derived based on the linearized system model in (3)-(4), the observability properties of this model play a crucial role in determining the performance of the estimator. Ideally, one would like these properties to match those of the actual, nonlinear system in (1)-(2): if a certain quantity is unobservable in the actual system, its error should also be unobservable in the linearized model. However, in [18] it was shown that this is *not* the case in 2D EKF SLAM: due to the way the Jacobians are computed in the EKF, the robot orientation appears to be observable in the linearized system, while it is not in the actual, nonlinear one. As a result of this mismatch, the filter produces too small values for the state covariance matrix (i.e., the filter becomes *inconsistent*), and this in turn degrades accuracy. Our analysis in Section 4 proves that the same problem affects the MSCKF for VIO.

The observability properties of the nonlinear system for visual-inertial navigation have recently been studied in [2,20]. It has been shown that when a camera/IMU system navigates in an environment with a known gravitational acceleration but *no* known features, four degrees of freedom are unobservable: three corresponding to the global position, and one corresponding to the rotation about the gravity vector (i.e., the yaw). In our work, we examine the observability properties of the MSCKF's linearized system model by analyzing the observability matrix:

$$\mathcal{O} \triangleq \begin{bmatrix} \mathbf{H}_k \\ \mathbf{H}_{k+1}\mathbf{\Phi}_k \\ \vdots \\ \mathbf{H}_{k+m}\mathbf{\Phi}_{k+m-1}\cdots\mathbf{\Phi}_k \end{bmatrix} \tag{5}$$

For the linearized system to have the correct observability properties, the nullspace of $\mathcal{O}$ should be of dimension four, in agreement with the four unobservable quantities discussed above. In Section 4 we show that this is generally *not* the case: the yaw erroneously appears to be observable in the linearized system model, with detrimental effects to the filter's consistency. Furthermore, in Section 5 we show how small modifications to the MSCKF equations can ensure appropriate properties of the matrix $\mathcal{O}$, and substantially improve the filter's performance.

3 IMU Propagation Model

As seen in (5), to analyze the observability properties of the MSCKF's linearized system model we must have an expression for the error-state transition matrix, $\mathbf{\Phi}_i$. In previous work on inertial navigation, the discrete-time error-state transition matrix for the IMU state has been computed in a number of ways. Most existing methods stem from the integration of the differential equation $\dot{\mathbf{\Phi}}(t, t_i) = \mathbf{F}(t)\mathbf{\Phi}(t, t_i)$, where $\mathbf{F}(t)$ is the Jacobian of the continuous-time system model (see (1) and (9)). For instance, [1] employs Runge-Kutta numerical integration, [21] presents a closed-form, approximate solution to the differential equation, while many algorithms employ the simple approximation $\mathbf{\Phi} \simeq \mathbf{I} + \mathbf{F}\Delta t$ (which is equivalent to using one-step Euler integration) (e.g., [22] and references therein). All these methods for computing $\mathbf{\Phi}$ have the disadvantage that, being numerical in nature, they are not amenable to theoretical analysis. More importantly, however, when $\mathbf{\Phi}$ is computed numerically and/or approximately, we have no guarantee about the properties of this matrix. Specifically, we *cannot* guarantee that the observability matrix in (5) will have the desirable nullspace, a prerequisite for consistent estimation.

To address this problem, in this section we provide a closed-form expression for the IMU error-state transition matrix, which can be used for theoretical analysis.

3.1 IMU State Modeling

We consider an IMU, to which we affix a coordinate frame $\{I\}$, moving with respect to a global frame $\{G\}$. The IMU (gyroscope and accelerometer) measurements are given by[1]

$$\boldsymbol{\omega}_m = {}^I\boldsymbol{\omega} + \mathbf{b_g} + \mathbf{n_r} \tag{6}$$

$$\mathbf{a}_m = {}^I_G\mathbf{R}\left({}^G\mathbf{a} - {}^G\mathbf{g}\right) + \mathbf{b_a} + \mathbf{n_a} \tag{7}$$

where ${}^I\boldsymbol{\omega}$ and ${}^G\mathbf{a}$ denote the IMU angular rate and linear acceleration respectively, $\mathbf{n_r}$ and $\mathbf{n_a}$ are white Gaussian noise processes, $\mathbf{b_g}$ and $\mathbf{b_a}$ are measurement biases modeled as random walk processes, and ${}^G\mathbf{g}$ is the gravity vector.

To use the IMU measurements for state propagation, we define the IMU state vector as follows [1]

$$\mathbf{x}_I = \begin{bmatrix} {}^I_G\bar{\mathbf{q}}^T & {}^G\mathbf{p}^T & {}^G\mathbf{v}^T & \mathbf{b_g}^T & \mathbf{b_a}^T \end{bmatrix}^T \tag{8}$$

where ${}^I_G\bar{\mathbf{q}}$ is the unit quaternion describing the rotation from the global frame to the IMU frame (i.e., $\mathbf{R}({}^I_G\bar{\mathbf{q}}) = {}^I_G\mathbf{R}$), and ${}^G\mathbf{p}$ and ${}^G\mathbf{v}$ denote the IMU position and velocity, respectively.

The continuous-time motion dynamics of the IMU are described by the following equations:

$$\begin{gathered} {}^I_G\dot{\bar{\mathbf{q}}}(t) = \tfrac{1}{2}\mathbf{\Omega}({}^I\boldsymbol{\omega}(t)){}^I_G\bar{\mathbf{q}}(t) \quad {}^G\dot{\mathbf{p}}(t) = {}^G\mathbf{v}(t) \\ {}^G\dot{\mathbf{v}}(t) = {}^G\mathbf{a}(t) \quad \dot{\mathbf{b}}_\mathbf{g}(t) = \mathbf{n}_{\mathbf{w}g}(t) \quad \dot{\mathbf{b}}_\mathbf{a}(t) = \mathbf{n_{wa}}(t) \end{gathered} \tag{9}$$

where $\mathbf{n}_{\mathbf{w}g}$ and $\mathbf{n_{wa}}$ are white Gaussian noise processes, and

$$\mathbf{\Omega}({}^I\boldsymbol{\omega}) = \begin{bmatrix} -\lfloor {}^I\boldsymbol{\omega}\times\rfloor & {}^I\boldsymbol{\omega} \\ -{}^I\boldsymbol{\omega}^T & 0 \end{bmatrix} \tag{10}$$

Following [1, 21], the IMU error-state is defined as:

$$\tilde{\mathbf{x}}_I = \begin{bmatrix} {}^I\tilde{\boldsymbol{\theta}}^T & {}^G\tilde{\mathbf{p}}^T & {}^G\tilde{\mathbf{v}}^T & \tilde{\mathbf{b}}_\mathbf{g}^T & \tilde{\mathbf{b}}_\mathbf{a}^T \end{bmatrix}^T \tag{11}$$

Here, for the position, velocity, and biases, the standard additive error definition is used $\left(\text{e.g., } {}^G\tilde{\mathbf{p}} = {}^G\mathbf{p} - {}^G\hat{\mathbf{p}}\right)$. On the other hand, the orientation error ${}^I\tilde{\boldsymbol{\theta}}$ satisfies the following equation [21]:

$${}^I_G\mathbf{R} \simeq \left(\mathbf{I}_3 - \lfloor {}^I\tilde{\boldsymbol{\theta}}\times\rfloor\right){}^I_G\hat{\mathbf{R}} \tag{12}$$

3.2 Error Propagation

We now derive the state transition matrix $\mathbf{\Phi}_{I_\ell}$ that describes how the errors in the IMU state estimate evolve during propagation. For simplicity, we first derive $\mathbf{\Phi}_{I_\ell}$ ignoring the IMU biases, and the result including the bias terms is shown in Section 3.3.

At time step ℓ we use the IMU state estimate $\hat{\mathbf{x}}_{I_{\ell|\ell}}$ and the IMU measurements to compute the propagated state estimate, $\hat{\mathbf{x}}_{I_{\ell+1|\ell}}$. Our goal is to derive an expression for the IMU error-state transition matrix $\mathbf{\Phi}_{I_\ell}$ such that $\tilde{\mathbf{x}}_{I_{\ell+1|\ell}} \simeq$

[1]The preceding superscript for vectors (e.g., G in ${}^G\mathbf{a}$) denotes the frame of reference with respect to which quantities are expressed. ${}^A_B\mathbf{R}$ is the rotation matrix rotating vectors from frame $\{B\}$ to $\{A\}$, $\lfloor\mathbf{c}\times\rfloor$ denotes the skew symmetric matrix corresponding to vector $\mathbf{c}$, $\mathbf{0}_3$ and $\mathbf{I}_3$ are the 3 by 3 zero and identity matrices respectively, $\hat{a}$ and $\tilde{a}$ represent the estimate and error of the estimate of a variable a respectively, and $\hat{a}_{i|j}$ is the estimate of variable a at time step i given measurements up to time step j.

$\mathbf{\Phi}_{I_\ell}\tilde{\mathbf{x}}_{I_{\ell|\ell}}+\mathbf{w}_\ell$. Starting with the orientation error, we note that, regardless of the method used to integrate the continuous-time motion dynamics in (9), the estimates of the rotation matrix at time-steps ℓ and $\ell+1$ satisfy:

$$\hat{\mathbf{R}}_{\ell+1|\ell} = {}^{I_{\ell+1}}_{I_\ell}\hat{\mathbf{R}} \cdot \hat{\mathbf{R}}_{\ell|\ell} \tag{13}$$

where we have used the notation $\hat{\mathbf{R}}_{\ell|\ell}=\mathbf{R}({}^I_G\hat{\bar{q}}_{\ell|\ell})$ for brevity. ${}^{I_{\ell+1}}_{I_\ell}\hat{\mathbf{R}}$ is the estimated rotation between timesteps ℓ and $\ell+1$, computed using the IMU measurements. This estimate is corrupted by an error $\tilde{\boldsymbol{\theta}}_{\Delta\ell}$, defined by:

$${}^{I_{\ell+1}}_{I_\ell}\mathbf{R} \simeq \left(\mathbf{I}-\lfloor\tilde{\boldsymbol{\theta}}_{\Delta\ell}\times\rfloor\right)\cdot {}^{I_{\ell+1}}_{I_\ell}\hat{\mathbf{R}} \tag{14}$$

On the other hand, the true rotation matrices at ℓ and $\ell+1$ satisfy ${}^{I_{\ell+1}}_{G}\mathbf{R}={}^{I_{\ell+1}}_{I_\ell}\mathbf{R}\cdot{}^{I_\ell}_{G}\mathbf{R}$. Substituting (12), (13) and (14) in this equation, we obtain the following expression for the linearized error propagation:

$${}^I\tilde{\boldsymbol{\theta}}_{\ell+1|\ell} \quad\simeq\quad \hat{\mathbf{R}}_{\ell+1|\ell}\,\hat{\mathbf{R}}^T_{\ell|\ell}\cdot{}^I\tilde{\boldsymbol{\theta}}_{\ell|\ell}+\tilde{\boldsymbol{\theta}}_{\Delta\ell} \tag{15}$$

To calculate the velocity error terms, we start with the identity:

$$\begin{aligned}{}^G\hat{\mathbf{v}}_{\ell+1|\ell} &= {}^G\hat{\mathbf{v}}_{\ell|\ell}+\int_{t_\ell}^{t_{\ell+1}}{}^G\hat{\mathbf{a}}_\tau d\tau &(16)\\ &= {}^G\hat{\mathbf{v}}_{\ell|\ell}+\int_{t_\ell}^{t_{\ell+1}}\left({}^G_{I_\tau}\hat{\mathbf{R}}^{I_\tau}\mathbf{a}_m+{}^G\mathbf{g}\right)d\tau &(17)\end{aligned}$$

where we have used (7). By defining $\hat{\mathbf{s}}_\ell=\int_{t_\ell}^{t_{\ell+1}}{}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}^{I_\tau}\mathbf{a}_m d\tau$, we can write the above equation as:

$${}^G\hat{\mathbf{v}}_{\ell+1|\ell}={}^G\hat{\mathbf{v}}_{\ell|\ell}+{}^G\mathbf{g}\Delta t+\hat{\mathbf{R}}^T_{\ell|\ell}\hat{\mathbf{s}}_\ell \tag{18}$$

A key observation here is that $\hat{\mathbf{s}}_\ell$ is a vector that depends only on the measurements, and thus by linearizing (18) we obtain:

$${}^G\tilde{\mathbf{v}}_{\ell+1|\ell}\simeq-\hat{\mathbf{R}}^T_{\ell|\ell}\lfloor\hat{\mathbf{s}}_\ell\times\rfloor{}^I\tilde{\boldsymbol{\theta}}_{\ell|\ell}+{}^G\tilde{\mathbf{v}}_{\ell|\ell}+\hat{\mathbf{R}}^T_{\ell|\ell}\tilde{\mathbf{s}}_\ell \tag{19}$$

where the error term $\tilde{\mathbf{s}}_\ell$ depends only on the IMU measurement noise. For the IMU position, we similarly write:

$$\begin{aligned}{}^G\hat{\mathbf{p}}_{\ell+1|\ell} &= {}^G\hat{\mathbf{p}}_{\ell|\ell}+\int_{t_\ell}^{t_{\ell+1}}{}^G\hat{\mathbf{v}}_\tau d\tau\\ &= {}^G\hat{\mathbf{p}}_{\ell|\ell}+{}^G\hat{\mathbf{v}}_{\ell|\ell}\Delta t+\frac{1}{2}{}^G\mathbf{g}\Delta t^2+\hat{\mathbf{R}}^T_{\ell|\ell}\hat{\mathbf{y}}_\ell &(20)\end{aligned}$$

where $\hat{\mathbf{y}}_\ell=\int_{t_\ell}^{t_{\ell+1}}\int_{t_\ell}^{s}{}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}^{I_\tau}\mathbf{a}_m d\tau ds$. Proceeding to linearize the above equation, we obtain:

$${}^G\tilde{\mathbf{p}}_{\ell+1|\ell}\simeq-\hat{\mathbf{R}}^T_{\ell|\ell}\lfloor\hat{\mathbf{y}}_\ell\times\rfloor{}^I\tilde{\boldsymbol{\theta}}_{\ell|\ell}+{}^G\tilde{\mathbf{v}}_{\ell|\ell}\Delta t+{}^G\tilde{\mathbf{p}}_{\ell|\ell}+\hat{\mathbf{R}}^T_{\ell|\ell}\tilde{\mathbf{y}}_\ell \tag{21}$$

By combining (15), (19) and (21), we can now write:

$$\underbrace{\begin{bmatrix}{}^I\tilde{\boldsymbol{\theta}}_{\ell+1|\ell}\\ {}^G\tilde{\mathbf{p}}_{\ell+1|\ell}\\ {}^G\tilde{\mathbf{v}}_{\ell+1|\ell}\end{bmatrix}}_{\tilde{\mathbf{x}}_{I_{\ell+1|\ell}}}=\underbrace{\begin{bmatrix}\hat{\mathbf{R}}_{\ell+1|\ell}\,\hat{\mathbf{R}}^T_{\ell|\ell} & \mathbf{0}_3 & \mathbf{0}_3\\ -\hat{\mathbf{R}}^T_{\ell|\ell}\lfloor\hat{\mathbf{y}}_\ell\times\rfloor & \mathbf{I}_3 & \Delta t\mathbf{I}_3\\ -\hat{\mathbf{R}}^T_{\ell|\ell}\lfloor\hat{\mathbf{s}}_\ell\times\rfloor & \mathbf{0}_3 & \mathbf{I}_3\end{bmatrix}}_{\mathbf{\Phi}_{I_\ell}}\underbrace{\begin{bmatrix}{}^I\tilde{\boldsymbol{\theta}}_{\ell|\ell}\\ {}^G\tilde{\mathbf{p}}_{\ell|\ell}\\ {}^G\tilde{\mathbf{v}}_{\ell|\ell}\end{bmatrix}}_{\tilde{\mathbf{x}}_{I_{\ell|\ell}}}+\underbrace{\begin{bmatrix}\tilde{\boldsymbol{\theta}}_{\Delta\ell}\\ \hat{\mathbf{R}}^T_{\ell|\ell}\tilde{\mathbf{y}}_\ell\\ \hat{\mathbf{R}}^T_{\ell|\ell}\tilde{\mathbf{s}}_\ell\end{bmatrix}}_{\mathbf{w}_\ell} \tag{22}$$

To write the state transition matrix as a function of the state estimates only, we solve (18) and (20) for $\hat{\mathbf{s}}_\ell$ and $\hat{\mathbf{y}}_\ell$, respectively, to obtain:

$$\begin{aligned}\hat{\mathbf{s}}_\ell &= \hat{\mathbf{R}}_{\ell|\ell}\left({}^G\hat{\mathbf{v}}_{\ell+1|\ell}-{}^G\hat{\mathbf{v}}_{\ell|\ell}-{}^G\mathbf{g}\Delta t\right) &(23)\\ \hat{\mathbf{y}}_\ell &= \hat{\mathbf{R}}_{\ell|\ell}\left({}^G\hat{\mathbf{p}}_{\ell+1|\ell}-{}^G\hat{\mathbf{p}}_{\ell|\ell}-{}^G\hat{\mathbf{v}}_{\ell|\ell}\Delta t-\frac{1}{2}{}^G\mathbf{g}\Delta t^2\right) &(24)\end{aligned}$$

Therefore the IMU error-state transition matrix can be written as:

$$
\begin{aligned}
\mathbf{\Phi}_{I_\ell}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}},\hat{\mathbf{x}}_{I_{\ell|\ell}}) &= \begin{bmatrix} \hat{\mathbf{R}}_{\ell+1|\ell}\cdot\hat{\mathbf{R}}_{\ell|\ell}^T & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{\Phi}_{\mathbf{pq}}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}},\hat{\mathbf{x}}_{I_{\ell|\ell}}) & \mathbf{I}_3 & \Delta t\mathbf{I}_3 \\ \mathbf{\Phi}_{\mathbf{vq}}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}},\hat{\mathbf{x}}_{I_{\ell|\ell}}) & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix}, \\
\mathbf{\Phi}_{\mathbf{pq}}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}},\hat{\mathbf{x}}_{I_{\ell|\ell}}) &= -\lfloor\big({}^G\hat{\mathbf{p}}_{\ell+1|\ell} - {}^G\hat{\mathbf{p}}_{\ell|\ell} - {}^G\hat{\mathbf{v}}_{\ell|\ell}\Delta t - \frac{1}{2}{}^G\mathbf{g}\Delta t^2\big)\times\rfloor\hat{\mathbf{R}}_{\ell|\ell}^T \\
\mathbf{\Phi}_{\mathbf{vq}}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}},\hat{\mathbf{x}}_{I_{\ell|\ell}}) &= -\lfloor({}^G\hat{\mathbf{v}}_{\ell+1|\ell} - {}^G\hat{\mathbf{v}}_{\ell|\ell} - {}^G\mathbf{g}\Delta t)\times\rfloor\hat{\mathbf{R}}_{\ell|\ell}^T
\end{aligned}
\tag{25}
$$

Note that this matrix is a closed-form function of the state estimates, and thus can be computed independently of the way in which the IMU state is integrated.

3.3 Full State Transition Matrix

If the biases are included in the derivations, the error-state transition matrix is computed following similar derivations as:

$$
\mathbf{\Phi}_{I_k} = \begin{bmatrix} \mathbf{\Phi}_{\mathbf{qq}} & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{\Phi}_{\mathbf{qb_g}} & \mathbf{0}_3 \\ \mathbf{\Phi}_{\mathbf{pq}} & \mathbf{I}_3 & \Delta t\mathbf{I}_3 & \mathbf{\Phi}_{\mathbf{pb_g}} & \mathbf{\Phi}_{\mathbf{pa}} \\ \mathbf{\Phi}_{\mathbf{vq}} & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{\Phi}_{\mathbf{vb_g}} & \mathbf{\Phi}_{\mathbf{va}} \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \tag{26}
$$

where

$$
\begin{aligned}
\mathbf{\Phi}_{\mathbf{qb_g}} &= -\hat{\mathbf{R}}_{\ell+1|\ell}\cdot\hat{\mathbf{R}}_{\ell|\ell}^T\int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}\,d\tau \\
\mathbf{\Phi}_{\mathbf{pb_g}} &= \int_{t_\ell}^{t_{\ell+1}}\int_{t_\ell}^{w}\lfloor({}^G\dot{\hat{\mathbf{v}}}_\tau - {}^G\mathbf{g})\times\rfloor\hat{\mathbf{R}}_{\ell|\ell}^T\int_{t_\ell}^{\tau}{}^{I_\ell}_{I_s}\hat{\mathbf{R}}\,ds\,d\tau\,dw \\
\mathbf{\Phi}_{\mathbf{pa}} &= -\hat{\mathbf{R}}_{\ell|\ell}^T\int_{t_\ell}^{t_{\ell+1}}\int_{t_\ell}^{\tau}{}^{I_\ell}_{I_s}\hat{\mathbf{R}}\,ds\,d\tau \\
\mathbf{\Phi}_{\mathbf{vb_g}} &= \int_{t_\ell}^{t_{\ell+1}}\lfloor({}^G\dot{\hat{\mathbf{v}}}_\tau - {}^G\mathbf{g})\times\rfloor\hat{\mathbf{R}}_{\ell|\ell}^T\int_{t_\ell}^{\tau}{}^{I_\ell}_{I_s}\hat{\mathbf{R}}\,ds\,d\tau \\
\mathbf{\Phi}_{\mathbf{va}} &= -\hat{\mathbf{R}}_{\ell|\ell}^T\int_{t_\ell}^{t_{\ell+1}}{}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}\,d\tau
\end{aligned}
\tag{27}
$$

The derivation of this result is shown in Appendix A.

4 Observability Analysis

In this section, we examine the observability properties of the linearized system model used in the MSCKF. For clarity, we here carry out the analysis for a state vector that does not include the IMU biases. Note however that, as shown in [20], these biases are observable for general motion. Therefore their inclusion in the state vector would not change the main result of this section, which is the artificial *increase* in the number of observable states. This result holds also when the biases are considered, as validated by the results in Section 6, were the biases are included in the estimated IMU state vector.

4.1 Camera measurement model

Assuming a calibrated perspective camera, the measurement of the i-th feature at time step ℓ is given by

$$
\begin{aligned}
\mathbf{z}_{i,\ell} &= h\big({}^{C_\ell}\mathbf{p}_{f_i}\big) + \mathbf{n}_{i,\ell}, \quad \text{with} && (28) \\
{}^{C_\ell}\mathbf{p}_{f_i} &= {}^C_I\mathbf{R}\,\mathbf{R}_\ell\big({}^G\mathbf{p}_{f_i} - {}^G\mathbf{p}_{I_\ell}\big) + {}^C\mathbf{p}_I && (29)
\end{aligned}
$$

In this expression $\{{}^C_I\mathbf{R},\ {}^C\mathbf{p}_I\}$ are the known rotation and translation between the camera and the IMU, $h(\cdot)$ is the pinhole camera model, $h(\mathbf{f}) = [f_x/f_z,\ f_y/f_z]^T$, and $\mathbf{n}_{i,\ell}$ is the measurement noise vector. In the MSCKF features are tracked for a number of frames, and then used for EKF updates. If feature i is processed for an MSCKF update at time-step $\alpha_i + 1$, the Jacobians of the measurement model with respect to the IMU state and the feature position are

$$
\begin{aligned}
\mathbf{H}_{I_{i,\ell}} &= \mathbf{J}_{i,\ell}\ {}^C_I\mathbf{R}\big[\ \lfloor \hat{\mathbf{R}}_{\ell|\alpha_i}\big({}^G\hat{\mathbf{p}}_{f_i} - {}^G\hat{\mathbf{p}}_{\ell|\alpha_i}\big)\times\rfloor\ \ -\hat{\mathbf{R}}_{\ell|\alpha_i}\ \ \mathbf{0}_3\ \big] \\
\mathbf{H}_{f_{i,\ell}} &= \mathbf{J}_{i,\ell}\ {}^C_I\mathbf{R}\ \hat{\mathbf{R}}_{\ell|\alpha_i}
\end{aligned}
\tag{30}
$$

$$
\mathbf{J}_{i,\ell} = \frac{\partial h(\mathbf{f})}{\partial \mathbf{f}}\bigg|_{\mathbf{f}={}^{C_\ell}\hat{\mathbf{p}}_{f_i}} = \frac{1}{{}^{C_\ell}\hat{z}_{f_i}}\begin{bmatrix} 1 & 0 & -\frac{{}^{C_\ell}\hat{x}_{f_i}}{{}^{C_\ell}\hat{z}_{f_i}} \\ 0 & 1 & -\frac{{}^{C_\ell}\hat{y}_{f_i}}{{}^{C_\ell}\hat{z}_{f_i}} \end{bmatrix}
\tag{31}
$$

Thus, the linearized measurement residual equation becomes;

$$
\mathbf{r}_{i,\ell} = \mathbf{H}_{I_{i,\ell}}\tilde{\mathbf{x}}_{I_{\ell|\ell-1}} + \mathbf{H}_{f_{i,\ell}}{}^G\tilde{\mathbf{p}}_{f_i} + \mathbf{n}_{i,\ell}
\tag{32}
$$

4.2 Structure of the observability matrix

To derive the observability matrix for MSCKF-based VIO, we first note that the MSCKF and EKF-SLAM rely on the same underlying linearized discrete-time models. Specifically, both approaches are derived based on the IMU error-state propagation model (22) and the linearized measurement residual model (32), but use different estimates for computing the IMU error-state transition matrices and the measurement Jacobians. Therefore, if the MSCKF and EKF-SLAM use the same linearization points, their implementations are based on exactly the same underlying linearized equations, which in turn indicates that their ways of information acquisition and their observability properties are the same. Thus, to analyze the observability properties of the MSCKF, we can analyze the equivalent EKF-SLAM system model, as long as we adjust the linearization points. In this paper, we define the following state vector, which contains the IMU state as well as the positions of N features observed by the camera in the time interval $[k, k+m]$:

$$
\mathbf{x}_I = \begin{bmatrix} {}^I_G\bar{\mathbf{q}}^T & {}^G\mathbf{p}^T & {}^G\mathbf{v}^T & {}^G\mathbf{p}_1^T & \cdots & {}^G\mathbf{p}_N^T \end{bmatrix}^T
\tag{33}
$$

If at time-step ℓ the camera observes n_ℓ features, the Jacobian $\mathbf{H}_\ell$ contains n_ℓ block rows of the form

$$
\mathbf{H}_\ell^{(i)} = \begin{bmatrix} \mathbf{H}_{I_{i,\ell}} & \mathbf{0}_3 & \cdots & \mathbf{H}_{f_{i,\ell}} & \cdots & \mathbf{0}_3 \end{bmatrix},\ i = 1, .., n_\ell
$$

where $\mathbf{H}_{I_{i,\ell}}$ and $\mathbf{H}_{f_{i,\ell}}$ are shown in (30). Thus, the block row of the observability matrix corresponding to the measurement of feature i at time step ℓ has the following structure:

$$
\mathcal{O}_\ell^{(i)} = \mathbf{M}_\ell^{(i)}\Big[\mathbf{A}_\ell^{(i)}\boldsymbol{\Phi}_{I_{\ell-1}}\cdots\boldsymbol{\Phi}_{I_k}\quad \mathbf{0}_3\quad \cdots\quad \mathbf{I}_3\quad \cdots\quad \mathbf{0}_3\Big],
\tag{34}
$$

$$
\mathbf{M}_\ell^{(i)} = \mathbf{J}_{i,\ell}\ {}^C_I\mathbf{R}\ \hat{\mathbf{R}}_{\ell|\alpha_i}
\tag{35}
$$

$$
\mathbf{A}_\ell^{(i)} = \Big[\ \lfloor {}^G\hat{\mathbf{p}}_{f_i} - {}^G\hat{\mathbf{p}}_{\ell|\alpha_i}\times\rfloor\hat{\mathbf{R}}_{\ell|\alpha_i}^T\quad -\mathbf{I}_3\quad \mathbf{0}_3\ \Big]
\tag{36}
$$

4.3 Using "ideal" Jacobians

It is interesting to first examine the properties of the observability matrix in the "ideal" case when the Jacobians are evaluated using the true state values. If we compute the state transition matrix as $\boldsymbol{\Phi}_{I_\ell}(\mathbf{x}_{I_{\ell+1}}, \mathbf{x}_{I_\ell})$ (see (25)), and evaluate the Jacobian matrices in (30) using the true states, substitution in (34) yields:

$$
\check{\mathcal{O}}_\ell^{(i)} = \check{\mathbf{M}}_\ell^{(i)}\Big[\check{\boldsymbol{\Gamma}}_\ell^{(i)}\quad -\mathbf{I}_3\quad -\Delta t_\ell\mathbf{I}_3\quad \mathbf{0}_3\quad \cdots\quad \mathbf{I}_3\quad \cdots\quad \mathbf{0}_3\Big],
\tag{37}
$$

$$
\check{\boldsymbol{\Gamma}}_\ell^{(i)} = \big\lfloor\big({}^G\mathbf{p}_{f_i} - {}^G\mathbf{p}_k - {}^G\mathbf{v}_k\Delta t_\ell - \frac{1}{2}{}^G\mathbf{g}\Delta t_\ell^2\big)\times\big\rfloor\mathbf{R}_k^T
\tag{38}
$$

In the above equations, Δt_ℓ denotes the time interval between time steps k and ℓ, and we have used the symbol "ˇ" to denote a matrix computed using the true state values.

If we now define the matrix $\mathbf{N}$ as:

$$\mathbf{N} = \begin{bmatrix} \mathbf{0}_3 & \mathbf{R}_k{}^G\mathbf{g} \\ \mathbf{I}_3 & -\lfloor {}^G\mathbf{p}_k \times \rfloor {}^G\mathbf{g} \\ \mathbf{0}_3 & -\lfloor {}^G\mathbf{v}_k \times \rfloor {}^G\mathbf{g} \\ \mathbf{I}_3 & -\lfloor {}^G\mathbf{p}_{f_1} \times \rfloor {}^G\mathbf{g} \\ \mathbf{I}_3 & -\lfloor {}^G\mathbf{p}_{f_2} \times \rfloor {}^G\mathbf{g} \\ \vdots & \vdots \\ \mathbf{I}_3 & -\lfloor {}^G\mathbf{p}_{f_N} \times \rfloor {}^G\mathbf{g} \end{bmatrix} \tag{39}$$

it is easy to verify that $\breve{\mathcal{O}}_\ell^{(i)} \cdot \mathbf{N} = \mathbf{0}_{2\times 4}$. Since this holds for any i and any ℓ (i.e., for all block rows of the observability matrix), we conclude that $\breve{\mathcal{O}} \cdot \mathbf{N} = \mathbf{0}$, which in turn means that all four columns of $\mathbf{N}$ belong to the nullspace of $\breve{\mathcal{O}}$. In addition, in Appendix C, we prove that the dimension of the nullspace of $\breve{\mathcal{O}}$ is equal to four, which indicates that the columns of $\mathbf{N}$ exactly consist of the basis of the nullspace of $\breve{\mathcal{O}}$. Moreover, in Appendix D, we show that the first block column of $\mathbf{N}$ corresponds to a global translation of the state vector, while the last column corresponds to rotations about gravity. In other words, the nullspace of the matrix $\breve{\mathcal{O}}$, which is the unobservable subspace of the linearized system model, has properties that agree with those of the actual, nonlinear system. Thus, if we were able to estimate all the Jacobians using the true state estimates, the linearized system model would have the desired observability properties.

4.4 Using the actual Jacobians

We now examine the observability properties of the linearized system when the state transition matrix and all Jacobians are computed using the latest available state estimates. Using the Jacobians in (30), the block row of $\mathcal{O}$ corresponding to the observation of feature i at time-step ℓ becomes

$$\mathcal{O}_\ell^{(i)} = \mathbf{M}_\ell^{(i)} \begin{bmatrix} \mathbf{\Gamma}_\ell^{(i)} + \Delta\mathbf{\Gamma}_\ell^{(i)} & -\mathbf{I}_3 & -\Delta t_\ell \mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3 \end{bmatrix} \tag{40}$$

where

$$\mathbf{\Gamma}_\ell^{(i)} = \lfloor {}^G\hat{\mathbf{p}}_{f_i} - {}^G\hat{\mathbf{p}}_{k|k} - {}^G\hat{\mathbf{v}}_{k|k}\Delta t_\ell - \frac{1}{2}{}^G\mathbf{g}\Delta t_\ell^2 \times \rfloor \hat{\mathbf{R}}_{k|k}^T \tag{41}$$

and

$$\begin{aligned} \Delta\mathbf{\Gamma}_\ell^{(i)} =& \Bigg(\lfloor {}^G\hat{\mathbf{p}}_{f_i} - {}^G\hat{\mathbf{p}}_{\ell|\alpha_i} \times \rfloor \bar{\mathbf{E}}_{\mathbf{q}} + \bar{\mathbf{E}}_{\mathbf{p}} + \sum_{j=k+1}^{\ell-1} \Big(\sum_{s=k+1}^{j} \mathbf{E}_{\mathbf{v}}^s \Delta t + \mathbf{E}_{\mathbf{p}}^j + \\ & \sum_{s=k+1}^{j-1} \mathbf{\Phi}_{\mathbf{vq}}(\hat{\mathbf{x}}_{I_{s+1|s}}, \hat{\mathbf{x}}_{I_{s|s}}) \hat{\mathbf{R}}_{s|s} \mathbf{E}_{\mathbf{q}}^s \Delta t + \mathbf{\Phi}_{\mathbf{pq}}(\hat{\mathbf{x}}_{I_{j+1|j}}, \hat{\mathbf{x}}_{I_{j|j}}) \hat{\mathbf{R}}_{j|j} \mathbf{E}_{\mathbf{q}}^j \Big) \Bigg) \hat{\mathbf{R}}_{k|k}^T \end{aligned} \tag{42}$$

with

$$\begin{aligned} \bar{\mathbf{E}}_{\mathbf{q}} &= \mathbf{I}_3 - \left(\hat{\mathbf{R}}_{\ell|\alpha_i}^T \hat{\mathbf{R}}_{\ell|\ell-1}\right) \prod_{n=k+1}^{\ell-1} \left(\hat{\mathbf{R}}_{n|n}^T \hat{\mathbf{R}}_{n|n-1}\right) \\ \mathbf{E}_{\mathbf{q}}^j &= \mathbf{I}_3 - \prod_{n=k+1}^{j} \left(\hat{\mathbf{R}}_{n|n}^T \hat{\mathbf{R}}_{n|n-1}\right) \\ \mathbf{E}_{\mathbf{p}}^j &= \lfloor {}^G\hat{\mathbf{p}}_{j|j-1} - {}^G\hat{\mathbf{p}}_{j|j} \times \rfloor, \quad \bar{\mathbf{E}}_{\mathbf{p}} = \lfloor {}^G\hat{\mathbf{p}}_{\ell|\ell-1} - {}^G\hat{\mathbf{p}}_{\ell|\alpha_i} \times \rfloor \\ \mathbf{E}_{\mathbf{v}}^j &= \lfloor {}^G\hat{\mathbf{v}}_{j|j-1} - {}^G\hat{\mathbf{v}}_{j|j} \times \rfloor \end{aligned} \tag{43}$$

By comparing (40) and (41) to (37) and (38) we see that the structure of the observability matrix in both cases is similar. The key difference is that when the Jacobians are evaluated using the state *estimates*, the "disturbance" term $\Delta\mathbf{\Gamma}_\ell^{(i)}$ appears. While $\Delta\mathbf{\Gamma}_\ell^{(i)}$ is quite complex, we can observe that it contains terms that depend on the corrections (e.g., ${}^G\hat{\mathbf{p}}_{j|j} - {}^G\hat{\mathbf{p}}_{j|j-1}$,

${}^G\hat{\mathbf{v}}_{j|j} - {}^G\hat{\mathbf{v}}_{j|j-1}$) that the filter applies at different time steps. Since these corrections are random, the term $\Delta\mathbf{\Gamma}_\ell^{(i)}$ is a random one, and this "destroys" the special structure of the observability matrix. As a result, the property $\mathcal{O}_\ell^{(i)} \cdot \mathbf{N} = \mathbf{0}$ does not hold, and it can be shown that the nullspace of $\mathcal{O}$ is now of dimension only three (see Appendix E). This nullspace is spanned by the first three column vectors (the first block column) of $\mathbf{N}$ in (39), which means that the global yaw *erroneously appears* to be observable. As a result the MSCKF underestimates the uncertainty of the yaw estimates, which, in turn, leads to loss of accuracy.

4.5 Observability of EKF-SLAM

In this section, we show that the inconsistency problem also exists in EKF-SLAM. In EKF-SLAM, the Jacobian matrices are computed as:

$$\begin{aligned}\mathbf{H}_{I_{i,\ell}} &= \mathbf{J}_{i,\ell} \cdot {}_I^C\mathbf{R} \left[\lfloor \hat{\mathbf{R}}_{\ell|\ell-1} \big({}^G\hat{\mathbf{p}}_{f_i,\ell|\ell-1} - {}^G\hat{\mathbf{p}}_{\ell|\ell-1} \big) \times \rfloor \quad -\hat{\mathbf{R}}_{\ell|\ell-1} \quad \mathbf{0}_3 \right] \\ \mathbf{H}_{f_{i,\ell}} &= \mathbf{J}_{i,\ell} \cdot {}_I^C\mathbf{R} \cdot \hat{\mathbf{R}}_{\ell|\ell-1}\end{aligned} \tag{44}$$

Since the only difference of the observability matrices for the MSCKF and EKF-SLAM is the linearization points, the block row of the observability matrix of EKF-SLAM, which corresponds to the observation of feature i at time-step ℓ, has the same structure with (40) but with different terms $\mathbf{M}_\ell^{(i)}$, $\mathbf{\Gamma}_\ell^{(i)}$. Specifically, block matrices $\mathbf{M}_\ell^{(i)}$ and $\mathbf{\Gamma}_\ell^{(i)}$ are computed as:

$$\begin{aligned}\mathbf{\Gamma}_\ell^{(i)} &= \lfloor {}^G\hat{\mathbf{p}}_{f_{i,k|k-1}} - {}^G\hat{\mathbf{p}}_{k|k} - {}^G\hat{\mathbf{v}}_{k|k}\Delta t_\ell - \frac{1}{2}{}^G\mathbf{g}\Delta t_\ell^2 \times \rfloor \hat{\mathbf{R}}_{k|k}^T + \Delta\bar{\mathbf{\Gamma}}_\ell^{(i)} \\ \mathbf{M}_\ell^{(i)} &= \mathbf{J}_{i,k} \cdot {}_I^C\mathbf{R} \cdot \hat{\mathbf{R}}_{\ell|\ell}\end{aligned} \tag{45}$$

where

$$\Delta\bar{\mathbf{\Gamma}}_\ell^{(i)} = \Delta\mathbf{\Gamma}_\ell^{(i)} + \Delta{}^G\mathbf{p}_{f_i} \tag{46}$$

and

$$\Delta{}^G\mathbf{p}_{f_i} = \lfloor {}^G\hat{\mathbf{p}}_{f_{i,\ell|\ell-1}} - {}^G\hat{\mathbf{p}}_{f_{i,k|k-1}} \times \rfloor \hat{\mathbf{R}}_{k|k}^T \tag{47}$$

Moveover, the block matrix $\bar{\mathbf{E}}_{\mathbf{q}}$ and $\bar{\mathbf{E}}_{\mathbf{p}}$ in $\Delta\mathbf{\Gamma}_\ell^{(i)}$ become:

$$\bar{\mathbf{E}}_{\mathbf{q}} = \lfloor {}^G\hat{\mathbf{p}}_{f_{i,\ell|\ell-1}} - {}^G\hat{\mathbf{p}}_{\ell|\ell-1} \times \rfloor \Big(\prod_{n=k+1}^{\ell-1} \big(\hat{\mathbf{R}}_{n|n}^T \hat{\mathbf{R}}_{n|n-1} \big) - \mathbf{I}_3 \Big) \tag{48}$$

$$\bar{\mathbf{E}}_{\mathbf{p}} = \mathbf{0}_{3\times3} \tag{49}$$

We thus see that, the observability matrix of EKF-SLAM contains two disturbance terms, $\Delta\mathbf{\Gamma}_\ell^{(i)}$ and $\Delta{}^G\mathbf{p}_{f_i}$. The first term is generated due to the different estimates of the same IMU states used in the filter Jacobians, and the second term due to the different estimates of the same features. Similarly to the proof of the rank of the MSCKF observability matrix in Appendix E, we can easily prove that the dimension of the nullspace of the EKF-SLAM observability matrix is also three, where the yaw appears to be observable. Thus, EKF-SLAM is also inconsistent.

5 Improving the performance of the MSCKF

In this section, we propose modifications to the original MSCKF algorithm that ensure that the linearized system model has appropriate observability properties. As shown in the preceding section, the root cause of the problem is the fact that *different estimates of the same states* appear in the Jacobians. These estimates result in nonzero values for the terms $\bar{\mathbf{E}}_{\mathbf{q}}$, $\mathbf{E}_{\mathbf{q}}^j$, $\mathbf{E}_{\mathbf{p}}^j$, $\bar{\mathbf{E}}_{\mathbf{p}}$, $\mathbf{E}_{\mathbf{v}}^j$, and lead to incorrect properties for the observability matrix. The modifications proposed in this section aim at removing these terms, to restore the appropriate dimension of the unobservable subspace.

5.1 Global orientation error parametrization

We first address the orientation-dependent terms, $\bar{\mathbf{E}}_{\mathbf{q}}$ and $\mathbf{E}_{\mathbf{q}}^j$. Specifically, we propose a simple re-parameterization of the IMU orientation error: instead of using the error definition in (12), we employ the following one:

$$ {}^I_G\mathbf{R} \simeq {}^I_G\hat{\mathbf{R}} \left(\mathbf{I}_3 - \lfloor {}^G\tilde{\boldsymbol{\theta}} \times \rfloor \right) \tag{50} $$

Note that here the matrix $\mathbf{I}_3 - \lfloor {}^G\tilde{\boldsymbol{\theta}} \times \rfloor$ is a rotation matrix (to first-order approximation) that describes the rotation from the estimated global frame to the true one. Thus, the 3×1 vector ${}^G\tilde{\boldsymbol{\theta}}$ is the orientation error expressed in the *global* frame, while the original error parameterization in (12) expresses the error in the *local* frame. With this parameterization, the IMU error state at time step ℓ can be written as:

$$ \underbrace{\begin{bmatrix} {}^G\tilde{\boldsymbol{\theta}}_\ell \\ {}^G\tilde{\mathbf{p}}_\ell \\ {}^G\tilde{\mathbf{v}}_\ell \end{bmatrix}}_{\tilde{\mathbf{x}}^\star_\ell} = \begin{bmatrix} \hat{\mathbf{R}}_\ell^T \cdot {}^I\tilde{\boldsymbol{\theta}}_\ell \\ {}^G\tilde{\mathbf{p}}_\ell \\ {}^G\tilde{\mathbf{v}}_\ell \end{bmatrix} = \underbrace{\begin{bmatrix} \hat{\mathbf{R}}_\ell^T & \mathbf{0}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{I}_3 & \mathbf{0}_3 \\ \mathbf{0}_3 & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix}}_{\hat{\mathbf{C}}_\ell^T} \underbrace{\begin{bmatrix} {}^I\tilde{\boldsymbol{\theta}}_\ell \\ {}^G\tilde{\mathbf{p}}_\ell \\ {}^G\tilde{\mathbf{v}}_\ell \end{bmatrix}}_{\tilde{\mathbf{x}}_\ell} \tag{51} $$

Using (51), we can write:

$$ \tilde{\mathbf{x}}^\star_{\ell+1|\ell} = \hat{\mathbf{C}}^T_{\ell+1|\ell} \cdot \tilde{\mathbf{x}}_{\ell+1|\ell} = \hat{\mathbf{C}}^T_{\ell+1|\ell} \boldsymbol{\Phi}_{I_\ell}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}}, \hat{\mathbf{x}}_{I_{\ell|\ell}}) \cdot \tilde{\mathbf{x}}_{\ell|\ell} = \underbrace{\hat{\mathbf{C}}^T_{\ell+1|\ell} \boldsymbol{\Phi}_{I_\ell}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}}, \hat{\mathbf{x}}_{I_{\ell|\ell}}) \hat{\mathbf{C}}_{\ell|\ell}}_{\boldsymbol{\Phi}^\star_{I_\ell}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}}, \hat{\mathbf{x}}_{I_{\ell|\ell}})} \cdot \tilde{\mathbf{x}}^\star_{\ell|\ell} \tag{52} $$

Substituting (25) into the above equation, we can obtain the IMU error-state transition matrix for the global orientation parametrization:

$$ \begin{aligned} \boldsymbol{\Phi}^\star_{I_\ell}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}}, \hat{\mathbf{x}}_{I_{\ell|\ell}}) &= \begin{bmatrix} \mathbf{I}_3 & \mathbf{0}_3 & \mathbf{0}_3 \\ \boldsymbol{\Phi}^\star_{\mathbf{pq}}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}}, \hat{\mathbf{x}}_{I_{\ell|\ell}}) & \mathbf{I}_3 & \Delta t \mathbf{I}_3 \\ \boldsymbol{\Phi}^\star_{\mathbf{vq}}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}}, \hat{\mathbf{x}}_{I_{\ell|\ell}}) & \mathbf{0}_3 & \mathbf{I}_3 \end{bmatrix} \\ \boldsymbol{\Phi}^\star_{\mathbf{pq}}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}}, \hat{\mathbf{x}}_{I_{\ell|\ell}}) &= -\lfloor {}^G\hat{\mathbf{p}}_{\ell+1|\ell} - {}^G\hat{\mathbf{p}}_{\ell|\ell} - {}^G\hat{\mathbf{v}}_{\ell|\ell}\Delta t - \frac{1}{2}{}^G\mathbf{g}\Delta t^2 \times \rfloor \\ \boldsymbol{\Phi}^\star_{\mathbf{vq}}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}}, \hat{\mathbf{x}}_{I_{\ell|\ell}}) &= -\lfloor ({}^G\hat{\mathbf{v}}_{\ell+1|\ell} - {}^G\hat{\mathbf{v}}_{\ell|\ell} - {}^G\mathbf{g}\Delta t) \times \rfloor \end{aligned} $$

Moreover, the measurement Jacobian matrices become:

$$ \begin{aligned} \mathbf{H}^\star_{I_{i,\ell}} &= \mathbf{M}^{(i)}_\ell \begin{bmatrix} \lfloor \left({}^G\hat{\mathbf{p}}_{f_i} - {}^G\hat{\mathbf{p}}_{\ell|\alpha_i} \right) \times \rfloor & -\mathbf{I}_3 & \mathbf{0}_{3\times 3} \end{bmatrix} = \mathbf{M}^{(i)}_\ell \mathbf{A}^{(i)^\star}_\ell \\ \mathbf{H}^\star_{f_{i,\ell}} &= \mathbf{M}^{(i)}_\ell \end{aligned} \tag{53} $$

The key advantage of this parameterization is that both $\boldsymbol{\Phi}^\star_{I_\ell}$ and the term $\mathbf{A}^{(i)^\star}_\ell$ are independent of the orientation estimates. Substituting the above values in (34) we obtain the following for each block row of the observability matrix:

$$ \mathcal{O}^{(i)^\star}_\ell = \mathbf{M}^{(i)}_\ell \begin{bmatrix} \boldsymbol{\Gamma}^{(i)^\star}_\ell + \Delta\boldsymbol{\Gamma}^{(i)^\star}_\ell & -\mathbf{I}_3 & -\Delta t_\ell \mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3 \end{bmatrix} \tag{54} $$

where

$$ \begin{aligned} \boldsymbol{\Gamma}^{(i)^\star}_\ell &= \lfloor \left({}^G\hat{\mathbf{p}}_{f_i} - {}^G\hat{\mathbf{p}}_{k|k} - {}^G\hat{\mathbf{v}}_{k|k}\Delta t_\ell - \frac{1}{2}{}^G\mathbf{g}\Delta t_\ell^2 \right) \times \rfloor \\ \Delta\boldsymbol{\Gamma}^{(i)^\star}_\ell &= \bar{\mathbf{E}}_{\mathbf{p}} + \sum_{j=k+1}^{\ell-1} (\mathbf{E}^j_{\mathbf{p}} + \sum_{s=k+1}^{j} \mathbf{E}^s_{\mathbf{v}} \Delta t) \end{aligned} \tag{55} $$

We thus see that now the “disturbance” term $\Delta\boldsymbol{\Gamma}^{(i)^\star}_\ell$ is simplified, and does not contain any elements due to the orientation estimates. Next, we show how the remaining terms due to the position and velocity can also be removed.

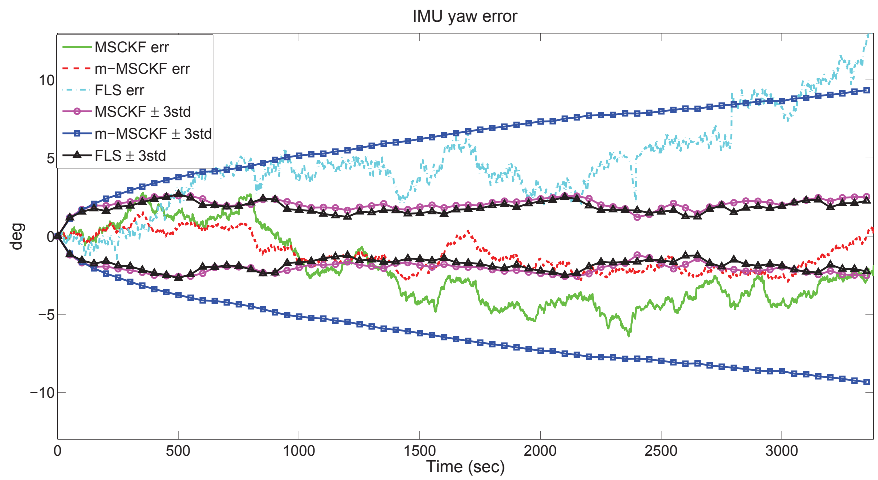

Figure 1: IMU yaw errors and $\pm 3\sigma$ bounds in one representative trial. The yaw error for the MSCKF (solid line – green), the m-MSCKF (dashed line – red), and the FLS (dashdot line – cyan). The $\pm 3\sigma$ bounds for the MSCKF (circles – magenta), the m-MSCKF (squares – blue), and the FLS (triangles – black).

5.2 Use of first-estimate Jacobians

The disturbance term $\Delta\mathbf{\Gamma}_\ell^{(i)\star}$ is a function of the differences between the estimates of the IMU position and velocity that are available at different time instances (see (43) and (55)). If we ensure that all Jacobians are computed using the *same* estimate for each of these states, the disturbance terms will vanish. Specifically, we here propose to use the *first* estimate of each IMU position and velocity when computing the filter Jacobian matrices [18]. This requires two changes. First, the state transition matrix at time-step ℓ is computed as $\mathbf{\Phi}^\star_{I_\ell}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}}, \hat{\mathbf{x}}_{I_{\ell|\ell-1}})$, instead of $\mathbf{\Phi}^\star_{I_\ell}(\hat{\mathbf{x}}_{I_{\ell+1|\ell}}, \hat{\mathbf{x}}_{I_{\ell|\ell}})$. Second, the measurement Jacobians are computed as follows:

$$\mathbf{H}^\star_{I_{i,\ell}} = \mathbf{M}_\ell^{(i)} \left[\lfloor ({}^G\hat{\mathbf{p}}_{f_i} - {}^G\hat{\mathbf{p}}_{\ell|\ell-1}) \times \rfloor \quad -\mathbf{I}_3 \quad \mathbf{0}_3 \right], \quad \mathbf{H}^\star_{f_{i,\ell}} = \mathbf{M}_\ell^{(i)}$$

As a result of these two changes, only the estimate ${}^G\hat{\mathbf{p}}_{\ell|\ell-1}$ (the first that becomes available) is used in all the Jacobians that involve ${}^G\mathbf{p}_\ell$, and the same holds for the velocity vectors ${}^G\mathbf{v}_\ell$, for all ℓ. In turn, it is easy to show that the term $\Delta\mathbf{\Gamma}_\ell^{(i)\star}$ in (55) becomes identically zero, and the observability matrix regains the correct rank. As shown in the next section, the modified MSCKF algorithm attains substantially improved performance, both in terms of consistency *and* accuracy. This occurs despite the fact that it uses older, and thus less accurate, estimates in computing Jacobians.

6 Results

6.1 Simulation tests

We first present the results of Monte-Carlo simulation tests, which allow us to examine the statistical properties of the modified MSCKF algorithm. To build a realistic simulation setting, we generate our simulation environment based on a real-world dataset, collected at the Cheddar Gorge area in the UK [23]. This dataset involves a 29.6-km long trajectory, travelled over 57 minutes. For our simulations, we generate a ground truth trajectory (position, velocity, orientation) that matches the vehicle's actual trajectory, as computed by a high-precision INS system. Using this trajectory, we subsequently generate IMU measurements corrupted with noise and bias characteristics similar to those of the Xsens MTi-G sensor used in the dataset. Moreover, we generate monocular feature tracks with statistical characteristics (feature number and distance, average track length, noise variance) similar to those of the actual dataset. Specifically, 225 features are observed in each image on average, and each feature's track length is sampled from an exponential distribution with a mean of 4.1 frames. The IMU measurements are available at 100 Hz, while the camera frame rate is 20 Hz, as in the actual dataset.

In each Monte-Carlo trial, the IMU measurements and feature tracks are randomly generated, and this data is processed by the following three algorithms: (i) The original MSCKF algorithm [1], (ii) The modified MSCKF algorithm described

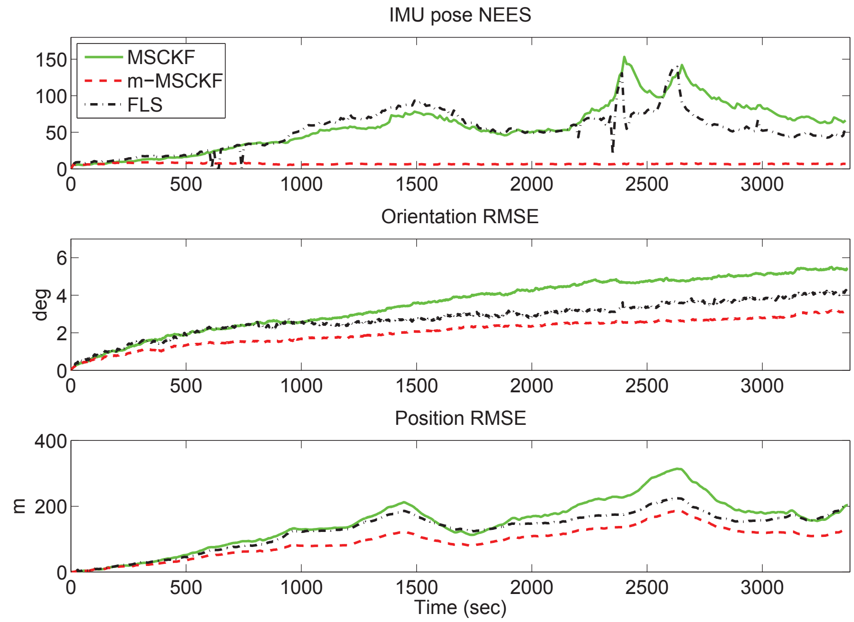

Figure 2: Average NEES and RMSE over 50 Monte Carlo trials. The solid green line corresponds to the MSCKF, the red dashed line to the m-MSCKF, and the black dashdotted line to the FLS.

in the previous section (denoted as m-MSCKF), and (iii) A fixed-lag smoother (FLS) in information form [11]. The FLS employs the same feature-marginalization approach as in the MSCKF, but uses iterative minimization, which enables it to re-linearize the measurement models at each iteration. To ensure a fair comparison all three algorithms process the same data, and use a sliding window of the same length.

Before presenting the cumulative results for all the Monte-Carlo trials, it is useful to examine the results of the three competing methods on a *single* trial. Specifically, the most interesting results are those for the estimates of the rotation about gravity (the yaw). Fig. 1 shows the yaw errors for the three algorithms, as well as the $\pm 3\sigma$ envelopes computed using the reported covariance of each method (these are the reported 99.7% confidence regions). The most important observation here is that the reported standard deviation for both the MSCKF and the FLS fluctuates about a constant value, *as if* the yaw was observable. In contrast, the reported standard deviation for the m-MSCKF continuously increases, which is what we expect given that the yaw is not actually observable. Moreover, this plot shows that the yaw errors of the MSCKF and FLS lie outside the $\pm 3\sigma$ bounds, which indicates inconsistency. Fig. 1 clearly demonstrates the effects of the incorrect observability properties of the MSCKF's linearized system model. These cause the yaw uncertainty to be underestimated, and lead to errors larger than those the filter expects. It is important to point out that the FLS also suffers from the same problem, even though it employs iterative re-linearization [10].

Fig. 2 plots the average NEES and RMS error for the IMU pose (position and orientation), averaged over 50 Monte-Carlo trials. Regarding the NEES, it becomes immediately clear that the m-MSCKF exhibits substantially higher consistency than the two competing methods. Specifically, the average NEES is 58.7 for the MSCKF, 52.7 for the FLS, and 6.8 for the m-MSCKF. We therefore see that the m-MSCKF obtains an NEES value close to the theoretically expected one for a consistent estimator, which is 6 (equal to the size of the error state). These results validate the theoretical analysis of Section 5, and demonstrate that the proposed modifications to the MSCKF significantly improve its consistency.

In addition to the consistency improvement, the results in Fig. 2 show that the m-MSCKF outperforms the two other methods in terms of *accuracy*. Specifically, the RMS error for the position (averaged over all trials and through time) is 148.9 m for the MSCKF, 129.1 m for the FLS, and 94.2 m for the m-MSCKF. For the orientation errors we obtain 3.55^{o} for the MSCKF, 2.79^{o} for the FLS, and 2.06^{o} for the m-MSCKF. In both cases, the m-MSCKF attains smaller overall errors. We attribute this to the fact that, by ensuring the correct observability properties for the linearized system model, the m-MSCKF is capable of more accurately representing the uncertainty of the different states. In turn, this makes it possible to compute more suitable values for the Kalman gain and the state corrections, leading to overall better accuracy.

Figure 3: Sample images recorded during the experiment.

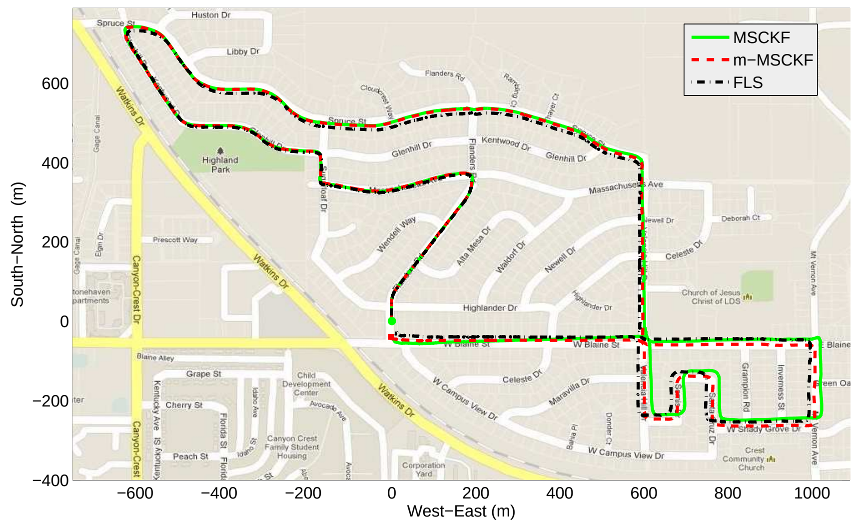

Figure 4: Trajectory estimates plotted on a map of Riverside. The initial vehicle position is shown by a green circle, and the end position by a red square. The green solid line corresponds to the MSCKF, the red dashed line to the m-MSCKF, and the black dashdotted line to the FLS.

6.2 Real-world experiment

We also present results from a real-world experiment, during which an IMU/camera platform was mounted on top of a car and driven on the streets of Riverside, CA. The sensors consisted of an Inertial Science ISIS IMU and a PointGrey Bumblebee2 stereo pair (only a single camera's images are used). The IMU provides measurements at 100 Hz, while the camera images were stored at 10 Hz. Harris feature points are extracted, and matching is carried out by normalized cross-correlation. The vehicle trajectory is approximately 5.5 km long, and a total of 7922 images are processed. Some sample images from the experiment are shown in Fig. 3.

Fig. 4 shows the trajectory estimates computed by the three algorithms (MSCKF, FLS, and m-MSCKF) on a map of the area where the vehicle drove. While a precise GPS ground truth is not available for this experiment, by closely examining the trajectory, we can observe that the m-MSCKF estimate closely follows the streets in the map. By contrast, the trajectories computed by the two other methods deviate from the street layout (this is most prominent in the south-east corner of the map). Moreover, Fig. 5 plots the reported standard deviation of the yaw for the three algorithms (since orientation ground truth is not available, the errors cannot be plotted). Similarly to what was observed in Fig. 1, we see that only the standard deviation for the m-MSCKF continuously increases, as predicted by the observability properties of the system. In contrast, the MSCKF and the FLS underestimate the yaw uncertainty, and obtain less accurate trajectory estimates. Thus, we see that the experimental results agree with the findings of the simulations, as well as the theoretical analysis.

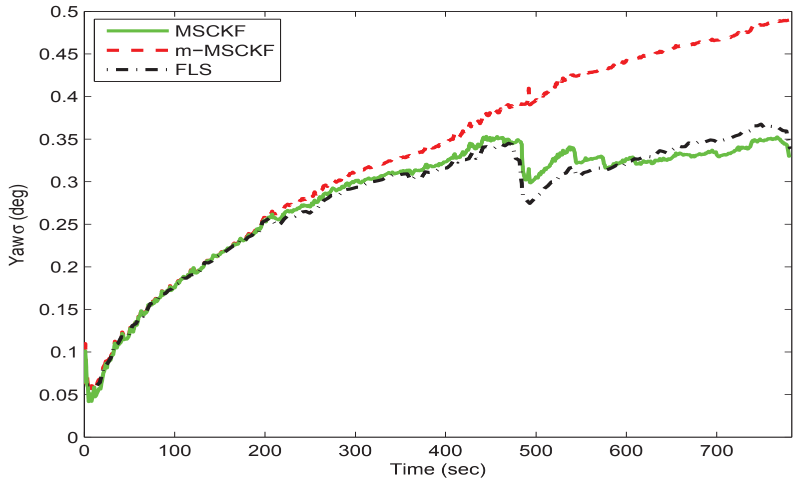

Figure 5: Comparison of the yaw standard deviation reported by the MSCKF (green solid line), the m-MSCKF (red dashed line), and the FLS (black dashdotted line).

7 Conclusion

In this paper we have presented a detailed theoretical analysis of the properties of the linearized system model used in EKF-based visual-inertial odometry. This analysis proved that this model has incorrect observability properties, which cause the global orientation to *appear to be* observable. In turn, this causes the filter to underestimate the uncertainty of the orientation estimates, i.e., to become *inconsistent*. Our results showed that this inconsistency also degrades the accuracy of the estimates. Based on the theoretical analysis, we proposed three modifications of the MSCKF algorithm for visual-inertial odometry [1]. These modifications, which incur no additional computational cost, include (i) A closed-form computation of the EKF error-state transition matrix, (ii) A new parameterization of the orientation error, and (iii) A new method of selecting the linearization points in the filter. Taken together, these modifications ensure that the resulting algorithm remains consistent. Our simulation and experimental results demonstrate that the modified MSCKF substantially outperforms the original algorithm, as well as iterative-minimization based fixed-lag smoothing. Overall, the theoretical and experimental results of the paper show that the modified MSCKF algorithm is capable of long-term, high-precision, consistent visual-inertial odometry.

Acknowledgments

This work was supported by the National Science Foundation (grant no. IIS-1117957), the UC Riverside Bourns College of Engineering, and the Hellman Family Foundation.

A IMU State transition matrix

We here show the detailed derivation of the IMU error-state transition matrix, when the biases are included. Starting with the orientation error, we note that the derivation presented in Section 3 still holds. The difference is that now the error term $\tilde{\boldsymbol{\theta}}_{\Delta\ell}$ in (15) depends on the error in the bias estimates. To derive $\tilde{\boldsymbol{\theta}}_{\Delta\ell}$, we start by introducing the following differential equation for the orientation matrix $\mathbf{R}_\ell$:

$$\dot{\mathbf{R}}_\ell = \lfloor {}^{I_\ell}\boldsymbol{\omega}\times\rfloor \mathbf{R}_\ell \tag{56}$$

Similarly, we can write:

$$\dot{\hat{\mathbf{R}}}_{\ell|\ell} = \lfloor {}^{I_\ell}\hat{\boldsymbol{\omega}}\times\rfloor \hat{\mathbf{R}}_{\ell|\ell} \tag{57}$$

Therefore, computing derivatives on both sides of (12) leads to

$$\lfloor {}^{I_\ell}\boldsymbol{\omega}\times\rfloor \mathbf{R}_\ell \simeq \lfloor {}^{I_\ell}\hat{\boldsymbol{\omega}}\times\rfloor \hat{\mathbf{R}}_{\ell|\ell} - \lfloor {}^{I}\tilde{\boldsymbol{\theta}}_{\ell|\ell}\times\rfloor\lfloor {}^{I_\ell}\hat{\boldsymbol{\omega}}\times\rfloor \hat{\mathbf{R}}_{\ell|\ell} - \lfloor {}^{I}\dot{\tilde{\boldsymbol{\theta}}}_{\ell|\ell}\times\rfloor \hat{\mathbf{R}}_{\ell|\ell} \tag{58}$$

Substituting (12) for $\mathbf{R}_\ell$ leads to

$$\lfloor {}^{I_\ell}\boldsymbol{\omega}\times\rfloor \hat{\mathbf{R}}_{\ell|\ell} - \lfloor {}^{I_\ell}\boldsymbol{\omega}\times\rfloor\lfloor {}^{I}\tilde{\boldsymbol{\theta}}_{\ell|\ell}\times\rfloor \hat{\mathbf{R}}_{\ell|\ell} \simeq \lfloor {}^{I_\ell}\hat{\boldsymbol{\omega}}\times\rfloor \hat{\mathbf{R}}_{\ell|\ell} - \lfloor {}^{I}\tilde{\boldsymbol{\theta}}_{\ell|\ell}\times\rfloor\lfloor {}^{I_\ell}\hat{\boldsymbol{\omega}}\times\rfloor \hat{\mathbf{R}}_{\ell|\ell} - \lfloor {}^{I}\dot{\tilde{\boldsymbol{\theta}}}_{\ell|\ell}\times\rfloor \hat{\mathbf{R}}_{\ell|\ell} \Rightarrow \tag{59}$$

$$\lfloor {}^{I_\ell}\tilde{\boldsymbol{\omega}}\times\rfloor - \lfloor {}^{I_\ell}\hat{\boldsymbol{\omega}}\times\rfloor\lfloor {}^{I}\tilde{\boldsymbol{\theta}}_{\ell|\ell}\times\rfloor \simeq -\lfloor {}^{I}\tilde{\boldsymbol{\theta}}_{\ell|\ell}\times\rfloor\lfloor {}^{I_\ell}\hat{\boldsymbol{\omega}}\times\rfloor - \lfloor {}^{I}\dot{\tilde{\boldsymbol{\theta}}}_{\ell|\ell}\times\rfloor \tag{60}$$

where we have used the notation ${}^{I_\ell}\tilde{\boldsymbol{\omega}} = {}^{I_\ell}\boldsymbol{\omega} - {}^{I_\ell}\hat{\boldsymbol{\omega}}$. Therefore, we can write

$$\lfloor {}^{I}\dot{\tilde{\boldsymbol{\theta}}}_{\ell|\ell}\times\rfloor \simeq -\left(\lfloor {}^{I_\ell}\hat{\boldsymbol{\omega}}\times\rfloor\lfloor {}^{I}\tilde{\boldsymbol{\theta}}_{\ell|\ell}\times\rfloor - \lfloor {}^{I}\tilde{\boldsymbol{\theta}}_{\ell|\ell}\times\rfloor\lfloor {}^{I_\ell}\hat{\boldsymbol{\omega}}\times\rfloor\right) - \lfloor {}^{I_\ell}\tilde{\boldsymbol{\omega}}\times\rfloor = -\lfloor\lfloor {}^{I_\ell}\hat{\boldsymbol{\omega}}\times\rfloor {}^{I}\tilde{\boldsymbol{\theta}}_{\ell|\ell}\times\rfloor - \lfloor {}^{I_\ell}\tilde{\boldsymbol{\omega}}\times\rfloor \tag{61}$$

Thus, the differential equation of the IMU orientation error becomes:

$${}^{I}\dot{\tilde{\boldsymbol{\theta}}}_{\ell|\ell} \simeq -\lfloor {}^{I_\ell}\hat{\boldsymbol{\omega}}\times\rfloor {}^{I}\tilde{\boldsymbol{\theta}}_{\ell|\ell} - {}^{I_\ell}\tilde{\boldsymbol{\omega}} \tag{62}$$

Solving the above equation, we obtain:

$${}^{I}\tilde{\boldsymbol{\theta}}_{\ell+1|\ell} \simeq \boldsymbol{\Phi}_{\mathbf{qq}}(\hat{\mathbf{x}}_{\ell+1|\ell}, \hat{\mathbf{x}}_{\ell|\ell}) {}^{I}\tilde{\boldsymbol{\theta}}_{\ell|\ell} + \int_{t_\ell}^{t_{\ell+1}} \boldsymbol{\Phi}_{\mathbf{qq}}(\hat{\mathbf{x}}_{\ell+1|\ell}, \hat{\mathbf{x}}_{\tau|\ell}) {}^{I_\tau}\tilde{\boldsymbol{\omega}} d\tau \tag{63}$$

where $\boldsymbol{\Phi}_{\mathbf{qq}}$ is the IMU orientation error-state transition matrix, which is shown in (15). In addition, by comparing (15) and (63) we can write:

$$\tilde{\boldsymbol{\theta}}_{\Delta\ell} \simeq \hat{\mathbf{R}}_{\ell+1|\ell}\hat{\mathbf{R}}_{\ell|\ell}^T \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}} {}^{I_\tau}\tilde{\boldsymbol{\omega}} d\tau \tag{64}$$

Using the definition of ${}^{I}\boldsymbol{\omega}_m$ in (6), we obtain:

$$\begin{aligned} {}^{I_\tau}\tilde{\boldsymbol{\omega}} &= \left({}^{I_\tau}\boldsymbol{\omega}_m - \mathbf{b}_{\mathbf{g}_\tau} - \mathbf{n}_{\mathbf{r}_\tau}\right) - \left({}^{I_\tau}\boldsymbol{\omega}_m - \hat{\mathbf{b}}_{\mathbf{g}_\tau}\right) && (65)\\ &= \tilde{\mathbf{b}}_{\mathbf{g}_\tau} - \mathbf{n}_{\mathbf{r}_\tau} && (66) \end{aligned}$$

Thus, (64) becomes:

$$\tilde{\boldsymbol{\theta}}_{\Delta\ell} \simeq \hat{\mathbf{R}}_{\ell+1|\ell}\hat{\mathbf{R}}_{\ell|\ell}^T \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}(-\tilde{\mathbf{b}}_{\mathbf{g}_\tau} - \mathbf{n}_{\mathbf{r}_\tau}) d\tau \tag{67}$$

where $\tilde{\mathbf{b}}_{\mathbf{g}_\tau}$ can be computed as:

$$\tilde{\mathbf{b}}_{\mathbf{g}_\tau} = \tilde{\mathbf{b}}_{\mathbf{g}_{\ell|\ell}} + \int_{t_\ell}^{\tau} \mathbf{n}_{\mathbf{w}g_s}\, ds \tag{68}$$

Combining (67) and (68) leads to:

$$\tilde{\boldsymbol{\theta}}_{\Delta\ell} \simeq -\hat{\mathbf{R}}_{\ell+1|\ell}\hat{\mathbf{R}}_{\ell|\ell}^T \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}} d\tau \cdot \tilde{\mathbf{b}}_{\mathbf{g}_{\ell|\ell}} + \underbrace{\hat{\mathbf{R}}_{\ell+1|\ell}\hat{\mathbf{R}}_{\ell|\ell}^T \int_{t_\ell}^{t_{\ell+1}} \left(-\mathbf{n}_{\mathbf{r}_\tau} - \int_{t_\ell}^{\tau} \mathbf{n}_{\mathbf{w}g_s}\, ds\right) d\tau}_{\mathbf{n}_{\boldsymbol{\theta}_{\ell+1}}} \tag{69}$$

Therefore, (15) becomes

$$ {}^{I}\tilde{\boldsymbol{\theta}}_{\ell+1|\ell} \simeq \hat{\mathbf{R}}_{\ell+1|\ell}\,\hat{\mathbf{R}}^{T}_{\ell|\ell} \cdot {}^{I}\tilde{\boldsymbol{\theta}}_{\ell|\ell} - \hat{\mathbf{R}}_{\ell+1|\ell}\hat{\mathbf{R}}^{T}_{\ell|\ell} \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}} d\tau \cdot \tilde{\mathbf{b}}_{\mathbf{g}_{\ell|\ell}} + \mathbf{n}_{\boldsymbol{\theta}_{\ell+1}} \tag{70} $$

and as a result we obtain:

$$ \boldsymbol{\Phi}_{\mathbf{q}\mathbf{b}_\mathbf{g}} = -\hat{\mathbf{R}}_{\ell+1|\ell} \cdot \hat{\mathbf{R}}^{T}_{\ell|\ell} \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}\, d\tau \tag{71} $$

Turning our attention to the velocity-related term, we start by including the IMU biases in $\hat{\mathbf{s}}_\ell$ in (16):

$$ \hat{\mathbf{s}}_\ell = \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}({}^{I_\tau}\mathbf{a}_m - \hat{\mathbf{b}}_{\mathbf{a}_\tau})d\tau \tag{72} $$

Now the error term $\tilde{\mathbf{s}}_\ell$ can be computed as:

$$ \begin{aligned} \tilde{\mathbf{s}}_\ell &= \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\mathbf{R}({}^{I_\tau}\mathbf{a}_m - \mathbf{b}_{\mathbf{a}_\tau} - \mathbf{n}_{\mathbf{a}_\tau})d\tau - \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}({}^{I_\tau}\mathbf{a}_m - \hat{\mathbf{b}}_{\mathbf{a}_\tau})d\tau && (73)\\ &\simeq \int_{t_\ell}^{t_{\ell+1}} (\mathbf{I}_3 - \lfloor\boldsymbol{\theta}_{\Delta\tau}\times\rfloor){}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}({}^{I_\tau}\mathbf{a}_m - \mathbf{b}_{\mathbf{a}_\tau} - \mathbf{n}_{\mathbf{a}_\tau})d\tau - \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}({}^{I_\tau}\mathbf{a}_m - \hat{\mathbf{b}}_{\mathbf{a}_\tau})d\tau && (74)\\ &= \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}(-\tilde{\mathbf{b}}_{\mathbf{a}_\tau} - \mathbf{n}_{\mathbf{a}_\tau})d\tau - \int_{t_\ell}^{t_{\ell+1}} \lfloor\boldsymbol{\theta}_{\Delta\tau}\times\rfloor{}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}({}^{I_\tau}\hat{\mathbf{a}} + \hat{\mathbf{b}}_{\mathbf{a}_\tau} - \hat{\mathbf{b}}_{\mathbf{a}_\tau} - \mathbf{n}_{\mathbf{a}_\tau})d\tau && (75)\\ &\simeq \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}(-\tilde{\mathbf{b}}_{\mathbf{a}_\tau} - \mathbf{n}_{\mathbf{a}_\tau})d\tau + \int_{t_\ell}^{t_{\ell+1}} \lfloor{}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}{}^{I_\tau}\hat{\mathbf{a}}\times\rfloor\boldsymbol{\theta}_{\Delta\tau}d\tau && (76)\\ &= \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}(-\tilde{\mathbf{b}}_{\mathbf{a}_\tau} - \mathbf{n}_{\mathbf{a}_\tau})d\tau + \int_{t_\ell}^{t_{\ell+1}} \hat{\mathbf{R}}_{\ell|\ell}\lfloor\hat{\mathbf{R}}^{T}_{\ell|\ell}{}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}{}^{I_\tau}\hat{\mathbf{a}}\times\rfloor\hat{\mathbf{R}}^{T}_{\ell|\ell}\boldsymbol{\theta}_{\Delta\tau}d\tau && (77)\\ &= \int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}(-\tilde{\mathbf{b}}_{\mathbf{a}_\tau} - \mathbf{n}_{\mathbf{a}_\tau})d\tau + \int_{t_\ell}^{t_{\ell+1}} \hat{\mathbf{R}}_{\ell|\ell}\lfloor{}^{G}\dot{\hat{\mathbf{v}}}_\tau - {}^{G}\mathbf{g}\times\rfloor\hat{\mathbf{R}}^{T}_{\ell|\ell}\boldsymbol{\theta}_{\Delta\tau}d\tau && (78) \end{aligned} $$

where in line (76) we have omitted terms involving the products of errors. At this point we substitute:

$$ \boldsymbol{\theta}_{\Delta\tau} = \int_{t_\ell}^{\tau} {}^{I_\ell}_{I_s}\hat{\mathbf{R}}ds \cdot \tilde{\mathbf{b}}_{\mathbf{g}_{\ell|\ell}} + \mathbf{n}_{\boldsymbol{\theta}_\tau} \tag{79} $$

$$ \tilde{\mathbf{b}}_{\mathbf{a}_\tau} = \tilde{\mathbf{b}}_{\mathbf{a}_{\ell|\ell}} - \int_{t_\ell}^{\tau} \mathbf{n}_{\mathbf{a}_s}\, ds \tag{80} $$

Thus, (78) becomes:

$$ \tilde{\mathbf{s}}_\ell = -\int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}\, d\tau \cdot \tilde{\mathbf{b}}_{\mathbf{a}_{\ell|\ell}} + \hat{\mathbf{R}}_{\ell|\ell} \int_{t_\ell}^{t_{\ell+1}} \lfloor{}^{G}\dot{\hat{\mathbf{v}}}_\tau - {}^{G}\mathbf{g}\times\rfloor\hat{\mathbf{R}}^{T}_{\ell|\ell}\int_{t_\ell}^{\tau} {}^{I_\ell}_{I_s}\hat{\mathbf{R}}\, ds\, d\tau \cdot \tilde{\mathbf{b}}_{\mathbf{g}_{\ell|\ell}} + \mathbf{n}_{\mathbf{v}_{\ell+1}} \tag{81} $$

where $\mathbf{n}_{\mathbf{v}_{\ell+1}}$ represents all the noise terms that do not depend on the error state. By substituting the above equation into (19), we obtain:

$$ \begin{aligned} \boldsymbol{\Phi}_{\mathbf{v}\mathbf{b}_\mathbf{g}} &= \int_{t_\ell}^{t_{\ell+1}} \lfloor({}^{G}\dot{\hat{\mathbf{v}}}_\tau - {}^{G}\mathbf{g})\times\rfloor\hat{\mathbf{R}}^{T}_{\ell|\ell}\int_{t_\ell}^{\tau} {}^{I_\ell}_{I_s}\hat{\mathbf{R}}\, ds\, d\tau && (82)\\ \boldsymbol{\Phi}_{\mathbf{v}\mathbf{a}} &= -\hat{\mathbf{R}}^{T}_{\ell|\ell}\int_{t_\ell}^{t_{\ell+1}} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}\, d\tau && (83) \end{aligned} $$

For the IMU position error, we start by including the biases in the term $\hat{\mathbf{y}}_\ell$:

$$ \hat{\mathbf{y}}_\ell = \int_{t_\ell}^{t_{\ell+1}}\int_{t_\ell}^{s} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}({}^{I_\tau}\mathbf{a}_m - \hat{\mathbf{b}}_{\mathbf{a}_\tau})d\tau ds \tag{84} $$

and thus the error term $\tilde{\mathbf{y}}_\ell$ is given by:

$$\tilde{\mathbf{y}}_\ell = \int_{t_\ell}^{t_{\ell+1}} \int_{t_\ell}^{s} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}({}^{I_\tau}\mathbf{a}_m - \mathbf{b}_{\mathbf{a}_\tau} - \mathbf{n}_{\mathbf{a}_\tau}) d\tau ds - \int_{t_\ell}^{t_{\ell+1}} \int_{t_\ell}^{s} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}({}^{I_\tau}\mathbf{a}_m - \hat{\mathbf{b}}_{\mathbf{a}_\tau}) d\tau ds \tag{85}$$

$$= \int_{t_\ell}^{t_{\ell+1}} \left(\int_{t_\ell}^{s} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}({}^{I_\tau}\mathbf{a}_m - \mathbf{b}_{\mathbf{a}_\tau} - \mathbf{n}_{\mathbf{a}_\tau}) d\tau ds - \int_{t_\ell}^{s} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}}({}^{I_\tau}\mathbf{a}_m - \hat{\mathbf{b}}_{\mathbf{a}_\tau}) \right) d\tau ds \tag{86}$$

$$= \int_{t_\ell}^{t_{\ell+1}} \tilde{\mathbf{s}}_s \, ds \tag{87}$$

At this point we use the result of (87) to write:

$$\tilde{\mathbf{y}}_\ell^b = -\int_{t_\ell}^{t_{\ell+1}}\int_{t_\ell}^{\tau} {}^{I_\ell}_{I_\tau}\hat{\mathbf{R}} \, ds \, d\tau \cdot \tilde{\mathbf{b}}_{\mathbf{a}_{\ell|\ell}} + \hat{\mathbf{R}}_{\ell|\ell} \int_{t_\ell}^{t_{\ell+1}}\int_{t_\ell}^{w} \lfloor {}^{G}\dot{\hat{\mathbf{v}}}_\tau - {}^{G}\mathbf{g} \times \rfloor \hat{\mathbf{R}}_{\ell|\ell}^T \int_{t_\ell}^{\tau} {}^{I_\ell}_{I_s}\hat{\mathbf{R}} \, ds \, d\tau \, dw \cdot \tilde{\mathbf{b}}_{\mathbf{g}_{\ell|\ell}} + \mathbf{n}_{\mathbf{p}_{\ell+1}} \tag{88}$$

where $\mathbf{n}_{\mathbf{p}_{\ell+1}}$ is a noise term independent of the error state. Thus, by combining this result with (21), we obtain

$$\mathbf{\Phi}_{\mathbf{p}\mathbf{b}_\mathbf{g}} = \int_{t_\ell}^{t_{\ell+1}} \int_{t_\ell}^{w} \lfloor ({}^{G}\dot{\hat{\mathbf{v}}}_\tau - {}^{G}\mathbf{g}) \times \rfloor \hat{\mathbf{R}}_{\ell|\ell}^T \int_{t_\ell}^{\tau} {}^{I_\ell}_{I_s}\hat{\mathbf{R}} \, ds \, d\tau \, dw \tag{89}$$

$$\mathbf{\Phi}_{\mathbf{p}\mathbf{a}} = -\hat{\mathbf{R}}_{\ell|\ell}^T \int_{t_\ell}^{t_{\ell+1}} \int_{t_\ell}^{\tau} {}^{I_\ell}_{I_s}\hat{\mathbf{R}} \, ds \, d\tau \tag{90}$$

B Analysis of EKF SLAM and the MSCKF algorithm

We here prove that, in a linear-Gaussian system, the state estimate and covariance matrix computed by the MSCKF is identical to the MAP estimate for the IMU pose. Since EKF-SLAM is also a MAP estimator, this means that the MSCKF and EKF-SLAM would be identical in a linear-Gaussian system.

Let us consider the following linear system:

$$\mathbf{x}_i = \mathbf{\Phi}_i \mathbf{x}_{i-1} + \mathbf{w}_{i-1} \tag{91}$$

$$\mathbf{z}_{ij} = \mathbf{H}_{\mathbf{x}_{ij}} \mathbf{x}_i + \mathbf{H}_{\mathbf{f}_{ij}} \mathbf{p}_{f_j} + \mathbf{n}_{ij} \tag{92}$$

where $\mathbf{x}_i$, $i = 0 \dots N$ are the IMU states, $\mathbf{p}_{f_j}$, $j = 1 \dots M$ are the feature positions, $\mathbf{w}_i$ and $\mathbf{n}_{ij}$ are zero-mean white Gaussian noise processes with covariance matrices $\mathbf{Q}_i$ and $\sigma^2 \mathbf{I}_2$, respectively, and $\mathbf{\Phi}_i$, $\mathbf{H}_{\mathbf{x}_{ij}}$, and $\mathbf{H}_{\mathbf{f}_{ij}}$ are known matrices. By denoting the vector containing all the IMU states as $\mathbf{x} = \begin{bmatrix} \mathbf{x}_0^T & \mathbf{x}_1^T & \mathbf{x}_2^T & \cdots & \mathbf{x}_N^T \end{bmatrix}^T$, (91) can be written as:

$$\mathbf{x} = \begin{bmatrix} \mathbf{x}_0 \\ \mathbf{x}_1 \\ \mathbf{x}_2 \\ \vdots \\ \mathbf{x}_N \end{bmatrix} = \underbrace{\begin{bmatrix} \mathbf{I} \\ \mathbf{\Phi}_1 \\ \mathbf{\Phi}_2\mathbf{\Phi}_1 \\ \vdots \\ \mathbf{\Phi}_N \cdots \mathbf{\Phi}_1 \end{bmatrix}}_{\mathbf{B}} \mathbf{x}_0 + \begin{bmatrix} \mathbf{0} \\ \mathbf{w}_0 \\ \mathbf{\Phi}_2\mathbf{w}_0 + \mathbf{w}_1 \\ \vdots \\ \mathbf{\Phi}_N \cdots \mathbf{\Phi}_2\mathbf{w}_0 + \cdots + \mathbf{w}_{N-1} \end{bmatrix} \tag{93}$$

In addition, we denote the vector containing all the feature positions as $\mathbf{f} = \begin{bmatrix} \mathbf{f}_1^T & \mathbf{f}_2^T & \cdots & \mathbf{f}_M^T \end{bmatrix}^T$, and the vector containing all measurements as

$$\mathbf{z} = \mathbf{H}_\mathbf{x}\mathbf{x} + \mathbf{H}_\mathbf{f}\mathbf{f} + \mathbf{n} \tag{94}$$

where $\mathbf{H}_\mathbf{x}$ and $\mathbf{H}_\mathbf{f}$, are matrices with block rows $\mathbf{H}_{\mathbf{x}_{ij}}$ and $\mathbf{H}_{\mathbf{f}_{i,j}}$, respectively.

To formulate the MAP estimator, we assume an initial estimate for the first robot pose, $\mathbf{x}_0 \sim \mathcal{N}(\hat{\mathbf{x}}_0, \mathbf{P}_0)$. Thus the MAP estimate for $\mathbf{x}$ and $\mathbf{f}$ can be obtained by:

$$\mathbf{x}_{\text{MAP}}, \mathbf{f}_{\text{MAP}} = \arg\max_{\mathbf{x},\mathbf{f}} \mathbf{P}(\mathbf{x}, \mathbf{f}, \mathbf{z}) \tag{95}$$

$$= \arg\max_{\mathbf{x},\mathbf{f}} \log\Big(\mathbf{P}(\mathbf{x}, \mathbf{f}, \mathbf{z})\Big) \tag{96}$$

$$= \quad \arg\max_{\mathbf{x},\mathbf{f}} \log\Big(\mathbf{P}(\mathbf{x})\,\mathbf{P}(\mathbf{z}|\mathbf{x},\mathbf{f})\Big) \tag{97}$$

where $\mathbf{P}(\cdot)$ denotes the probability function. Using (93), the prior distribution of the state $\mathbf{x}$ could be easily obtained as $\mathbf{x} \sim \mathcal{N}(\hat{\mathbf{x}}_s, \mathbf{P}_s)$, with:

$$\hat{\mathbf{x}}_s = \mathbf{B}\hat{\mathbf{x}}_0, \;\; \mathbf{P}_s = \begin{bmatrix} \mathbf{P}_0 & \mathbf{P}_0\mathbf{\Phi}_1^T & \mathbf{P}_0\mathbf{\Phi}_1^T\mathbf{\Phi}_2^T & \cdots \\ \mathbf{\Phi}_1\mathbf{P}_0 & \mathbf{\Phi}_1\mathbf{P}_0\mathbf{\Phi}_1^T + \mathbf{Q}_0 & \mathbf{\Phi}_1\mathbf{P}_0\mathbf{\Phi}_1^T\mathbf{\Phi}_2^T + \mathbf{Q}_0\mathbf{\Phi}_2^T & \cdots \\ \mathbf{\Phi}_2\mathbf{\Phi}_1\mathbf{P}_0 & \mathbf{\Phi}_2\mathbf{\Phi}_1\mathbf{P}_0\mathbf{\Phi}_1^T + \mathbf{\Phi}_2\mathbf{Q}_0 & \mathbf{\Phi}_2\mathbf{\Phi}_1\mathbf{P}_0\mathbf{\Phi}_1^T\mathbf{\Phi}_2^T + \mathbf{\Phi}_2\mathbf{Q}_0\mathbf{\Phi}_2^T + \mathbf{Q}_1 & \cdots \\ \vdots & \vdots & \vdots & \ddots \end{bmatrix} \tag{98}$$

Thus, the MAP estimate for $\mathbf{x}$ and $\mathbf{f}$ can be formulated as:

$$\mathbf{x}_{\mathrm{MAP}}, \mathbf{f}_{\mathrm{MAP}} = -\arg\min_{\mathbf{x},\mathbf{f}} \Big(\| \mathbf{x} - \hat{\mathbf{x}}_s \|^2_{\mathbf{P}_s} + \| \mathbf{z} - \mathbf{H}_x\mathbf{x} - \mathbf{H}_f\mathbf{f} \|^2_{(\sigma^2\mathbf{I})} \Big) \tag{99}$$

where we have used the notation $\| \mathbf{e} \|^2_{\mathbf{P}} = \mathbf{e}^T\mathbf{P}^{-1}\mathbf{e}$. By solving the above optimization problem, we obtain the optimal MAP estimate:

$$\begin{bmatrix} \hat{\mathbf{x}}_{\mathrm{MAP}} \\ \hat{\mathbf{f}}_{\mathrm{MAP}} \end{bmatrix} = \mathbf{\Lambda}^{-1} \begin{bmatrix} \mathbf{P}_s^{-1}\hat{\mathbf{x}}_s + \frac{1}{\sigma^2}\mathbf{H}_{\mathbf{x}}^T\mathbf{z} \\ \frac{1}{\sigma^2}\mathbf{H}_{\mathbf{f}}^T\mathbf{z} \end{bmatrix} \tag{100}$$

where $\mathbf{\Lambda}$ is the information matrix:

$$\mathbf{\Lambda} = \begin{bmatrix} \mathbf{P}_s^{-1} + \frac{1}{\sigma^2}\mathbf{H}_{\mathbf{x}}^T\mathbf{H}_{\mathbf{x}} & \frac{1}{\sigma^2}\mathbf{H}_{\mathbf{x}}^T\mathbf{H}_{\mathbf{f}} \\ \frac{1}{\sigma^2}\mathbf{H}_{\mathbf{f}}^T\mathbf{H}_{\mathbf{x}} & \frac{1}{\sigma^2}\mathbf{H}_{\mathbf{f}}^T\mathbf{H}_{\mathbf{f}} \end{bmatrix} \tag{101}$$

and $\mathbf{\Lambda}^{-1}$ is the covariance matrix of the MAP estimate. Using the standard properties of the inversion of a partitioned matrix, we can show from (100) that the estimate $\hat{\mathbf{x}}_{\mathrm{MAP}}$ and its covariance matrix equal:

$$\hat{\mathbf{x}}_{\mathrm{MAP}} = \mathbf{P}_{\mathrm{MAP}}\left(\mathbf{P}_s^{-1}\hat{\mathbf{x}}_s + \frac{1}{\sigma^2}\mathbf{H}_{\mathbf{x}}^T\left(\mathbf{I} - \mathbf{H}_{\mathbf{f}}\left(\mathbf{H}_{\mathbf{f}}^T\mathbf{H}_{\mathbf{f}}\right)^{-1}\mathbf{H}_{\mathbf{f}}^T\right)\mathbf{z}\right) \tag{102}$$

$$\mathbf{P}_{\mathrm{MAP}} = \left(\mathbf{P}_s^{-1} + \frac{1}{\sigma^2}\mathbf{H}_{\mathbf{x}}^T\left(\mathbf{I} - \mathbf{H}_{\mathbf{f}}\left(\mathbf{H}_{\mathbf{f}}^T\mathbf{H}_{\mathbf{f}}\right)^{-1}\mathbf{H}_{\mathbf{f}}^T\right)\mathbf{H}_{\mathbf{x}}\right)^{-1} \tag{103}$$

On the other hand, in the MSCKF algorithm, if we use the IMU measurements to propagate the state estimates, and then employ the camera measurements for an update, the update is performed based on the residual:

$$\mathbf{r}_o \doteq \mathbf{V}^T(\mathbf{z} - \mathbf{H}_{\mathbf{x}}^T\hat{\mathbf{x}}_s) = \left(\mathbf{V}^T\mathbf{H}_{\mathbf{x}}\right)\tilde{\mathbf{x}}_s + \mathbf{n}_o \tag{104}$$

where $\mathbf{V}$ is a matrix whose columns form an orthonormal basis for the left nullspace of $\mathbf{H}_{\mathbf{f}}$, and $\mathbf{n}_o$ is a noise vector with covariance matrix $\sigma^2\mathbf{I}$. Using the EKF equations, the state and covariance update can be written as:

$$\hat{\mathbf{x}}_{\mathrm{MSC}} = \hat{\mathbf{x}}_s + \mathbf{K}\mathbf{r}_o \tag{105}$$

$$\mathbf{P}_{\mathrm{MSC}} = \left(\mathbf{P}_s^{-1} + \frac{1}{\sigma^2}\left(\mathbf{V}^T\mathbf{H}_{\mathbf{x}}\right)^T\left(\mathbf{V}^T\mathbf{H}_{\mathbf{x}}\right)\right)^{-1} \tag{106}$$

where $\mathbf{K}$ is the Kalman gain, which can be written as [24]:

$$\mathbf{K} = \frac{1}{\sigma^2}\mathbf{P}_{\mathrm{MSC}}\left(\mathbf{V}^T\mathbf{H}_{\mathbf{x}}\right)^T \tag{107}$$

Our goal is to show that $\hat{\mathbf{x}}_{\mathrm{MSC}} = \hat{\mathbf{x}}_{\mathrm{MAP}}$, and $\mathbf{P}_{\mathrm{MSC}} = \mathbf{P}_{\mathrm{MAP}}$. To this end, we note that the matrix $\mathbf{I} - \mathbf{H}_{\mathbf{f}}\left(\mathbf{H}_{\mathbf{f}}^T\mathbf{H}_{\mathbf{f}}\right)^{-1}\mathbf{H}_{\mathbf{f}}^T$ is the orthogonal projector onto the left nullspace of $\mathbf{H}_{\mathbf{f}}$, and thus $\mathbf{I} - \mathbf{H}_{\mathbf{f}}\left(\mathbf{H}_{\mathbf{f}}^T\mathbf{H}_{\mathbf{f}}\right)^{-1}\mathbf{H}_{\mathbf{f}}^T = \mathbf{V}\mathbf{V}^T$ [25]. Using this result, the equality $\mathbf{P}_{\mathrm{MSC}} = \mathbf{P}_{\mathrm{MAP}}$ follows immediately, and we can also write (102) as:

$$\hat{\mathbf{x}}_{\mathrm{MAP}} = \mathbf{P}_{\mathrm{MSC}}\left(\mathbf{P}_s^{-1}\hat{\mathbf{x}}_s + \frac{1}{\sigma^2}\mathbf{H}_{\mathbf{x}}^T\mathbf{V}\mathbf{V}^T\mathbf{z}\right) \tag{108}$$

Substitution of (104) and (107) in (105) yields:

$$\begin{aligned}\hat{\mathbf{x}}_{\mathrm{MSC}} &= \hat{\mathbf{x}}_s + \frac{1}{\sigma^2}\mathbf{P}_{\mathrm{MSC}}\left(\mathbf{V}^T\mathbf{H}_{\mathbf{x}}\right)^T\mathbf{V}^T(\mathbf{z}-\mathbf{H}_{\mathbf{x}}^T\hat{\mathbf{x}}_s) \\ &= \mathbf{P}_{\mathrm{MSC}}\left(\left(\mathbf{P}_{\mathrm{MSC}}^{-1} - \frac{1}{\sigma^2}\mathbf{H}_{\mathbf{x}}^T\mathbf{V}\mathbf{V}^T\mathbf{H}_{\mathbf{x}}^T\right)\hat{\mathbf{x}}_s + \frac{1}{\sigma^2}\mathbf{H}_{\mathbf{x}}^T\mathbf{V}\mathbf{V}^T\mathbf{z}\right)\end{aligned}$$

Showing that the last equation is equal to (108) follows immediately by use of (106).

C Rank of the observability matrix in the "ideal" MSCKF

We here prove that the observability matrix shown in (37) has a nullspace of dimension 4. For this reason, we apply a sequence of elementary column operations on this matrix, to transform it to a different one with the same rank, but which facilitates analysis. In the following, we use the symbol $\sim$ to denote matrices related by elementary column operations. From (37) we obtain:

$$\breve{\mathcal{O}}_\ell^{(i)} = \breve{\mathbf{M}}_\ell^{(i)}\left[\begin{array}{ccc|ccccc} \lfloor\left({}^G\mathbf{p}_{f_i} - {}^G\mathbf{p}_k - {}^G\mathbf{v}_k\Delta t_\ell - \frac{1}{2}{}^G\mathbf{g}\Delta t_\ell^2\right)\times\rfloor\mathbf{R}_k^T & -\mathbf{I}_3 & \Delta t_\ell\mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3\end{array}\right] \tag{109}$$

where the partitioning denotes the separation between the columns corresponding to the IMU states and the those corresponding to the features. We now apply a sequence of elementary column operations, starting by multiplying the first block column by $\mathbf{R}_k$:

$$\breve{\mathcal{O}}_\ell^{(i)} \sim \breve{\mathbf{M}}_\ell^{(i)}\left[\begin{array}{ccc|ccccc} \lfloor\left({}^G\mathbf{p}_{f_i} - {}^G\mathbf{p}_k - {}^G\mathbf{v}_k\Delta t_\ell - \frac{1}{2}{}^G\mathbf{g}\Delta t_\ell^2\right)\times\rfloor & -\mathbf{I}_3 & \Delta t_\ell\mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3\end{array}\right]$$

Multiply second block column by -$\lfloor{}^G\mathbf{p}_k\times\rfloor$ and add to first block column]

$$\sim \breve{\mathbf{M}}_\ell^{(i)}\left[\begin{array}{ccc|ccccc} \lfloor\left({}^G\mathbf{p}_{f_i} - {}^G\mathbf{v}_k\Delta t_\ell - \frac{1}{2}{}^G\mathbf{g}\Delta t_\ell^2\right)\times\rfloor & -\mathbf{I}_3 & \Delta t_\ell\mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3\end{array}\right]$$

Multiply third block column by $\lfloor{}^G\mathbf{v}_k\times\rfloor$ and add to first block column

$$\sim \breve{\mathbf{M}}_\ell^{(i)}\left[\begin{array}{ccc|ccccc} \lfloor\left({}^G\mathbf{p}_{f_i} - \frac{1}{2}{}^G\mathbf{g}\Delta t_\ell^2\right)\times\rfloor & -\mathbf{I}_3 & \Delta t_\ell\mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3\end{array}\right]$$

Multiply column corresponding to i-th feature by $\lfloor -{}^G\mathbf{p}_{f_i}\times\rfloor$ and add to first block column, $\forall i$

$$\sim \breve{\mathbf{M}}_\ell^{(i)}\left[\begin{array}{ccc|ccccc} \lfloor -\frac{1}{2}{}^G\mathbf{g}\Delta t_\ell^2\times\rfloor & -\mathbf{I}_3 & \Delta t_\ell\mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3\end{array}\right]$$

Multiply first block column by -2

$$\sim \breve{\mathbf{M}}_\ell^{(i)}\left[\begin{array}{ccc|ccccc} \Delta t_\ell^2\lfloor{}^G\mathbf{g}\times\rfloor & -\mathbf{I}_3 & \Delta t_\ell\mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3\end{array}\right]$$

Add all block columns corresponding to the features to the second block column

$$\sim \breve{\mathbf{M}}_\ell^{(i)}\left[\begin{array}{ccc|ccccc} \Delta t_\ell^2\lfloor{}^G\mathbf{g}\times\rfloor & \mathbf{0}_3 & \Delta t_\ell\mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3\end{array}\right]$$

We now define the unitary matrix

$$\mathbf{F} = \begin{bmatrix}\mathbf{g}_{p_1} & \mathbf{g}_{p_2} & \frac{{}^G\mathbf{g}}{||{}^G\mathbf{g}||_2}\end{bmatrix}$$

where the two unit vectors $\mathbf{g}_{p_1}$ and $\mathbf{g}_{p_2}$ are on the plane perpendicular to ${}^G\mathbf{g}$, and are chosen to form an orthogonal coordinate system. Since $\mathbf{F}$ is non-singular, we can multiply the first block column of the above expression by $\mathbf{F}$ to obtain:

$$\breve{\mathcal{O}}_\ell^{(i)} \sim \breve{\mathbf{M}}_\ell^{(i)}\left[\begin{array}{cccc|ccccc} \Delta t_\ell^2\mathbf{G} & \mathbf{0}_{3\times 1} & \mathbf{0}_3 & \Delta t_\ell\mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3\end{array}\right] \tag{110}$$

where

$$\mathbf{G} = ||{}^G\mathbf{g}||_2\begin{bmatrix}\mathbf{g}_{p_1} & \mathbf{g}_{p_2}\end{bmatrix}$$

At this point, we note that through a sequence of elementary column operations, all block rows of the matrix $\breve{\mathcal{O}}$ have been transformed to a form where the third to sixth columns are all zero. Thus, the matrix $\breve{\mathcal{O}}$ is rank deficient by *at least four*, and the zero columns can be omitted without changing the rank:

$$\breve{\mathcal{O}}_\ell^{(i)} \sim \breve{\mathbf{M}}_\ell^{(i)}\left[\begin{array}{cc|ccccc} \Delta t_\ell^2\mathbf{G} & \Delta t_\ell\mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3\end{array}\right] = \breve{\mathbf{T}}_\ell^{(i)} \tag{111}$$

To show that the matrix $\breve{\mathcal{O}}$ is rank deficient by *exactly* four, we need to show that the matrix with block rows $\breve{\mathbf{T}}_\ell^{(i)}$ has full column rank. To this end, we define a vector

$$\mathbf{a} = \begin{bmatrix} \mathbf{a}_q \\ \mathbf{a}_v \\ \mathbf{a}_1 \\ \mathbf{a}_2 \\ \vdots \\ \mathbf{a}_N \end{bmatrix} \tag{112}$$

and it suffices to show that the condition $\breve{\mathbf{T}}_\ell^{(i)}\mathbf{a} = \mathbf{0}, \forall i, \ell$, is satisfied only if $\mathbf{a} = \mathbf{0}$. Substituting from (111) and (112) we obtain:

$$\begin{aligned} \breve{\mathbf{T}}_\ell^{(i)}\mathbf{a} = \mathbf{0},\ \forall i, \ell \ &\Rightarrow\ \breve{\mathbf{M}}_i^{(j)}\left(\Delta t_\ell^2 \mathbf{G}\mathbf{a}_q - \Delta t_\ell \mathbf{a}_v + \mathbf{a}_i\right) = \mathbf{0},\ \forall i, \ell \\ &\Rightarrow\ \begin{bmatrix} {}^{C_\ell}z_{f_i} & 0 & -{}^{C_\ell}x_{f_i} \\ 0 & {}^{C_\ell}z_{f_i} & -{}^{C_\ell}y_{f_i} \end{bmatrix} {}^C_I\mathbf{R}\mathbf{R}_i\left(\Delta t_\ell^2 \mathbf{G}\mathbf{a}_q - \Delta t_\ell \mathbf{a}_v + \mathbf{a}_i\right) = \mathbf{0},\ \forall i, \ell \end{aligned}$$

The above equation indicates that:

$$ {}^C_I\mathbf{R}\mathbf{R}_\ell\left(\Delta t_\ell^2 \mathbf{G}\mathbf{a}_q - \Delta t_\ell \mathbf{a}_v + \mathbf{a}_i\right) \in \mathcal{N}\left(\begin{bmatrix} {}^{C_\ell}z_{f_i} & 0 & -{}^{C_\ell}x_{f_i} \\ 0 & {}^{C_\ell}z_{f_i} & -{}^{C_\ell}y_{f_i} \end{bmatrix}\right),\ \forall i, \ell \tag{113}$$

Thus, we can write

$$ {}^C_I\mathbf{R}\mathbf{R}_\ell\left(\Delta t_\ell^2 \mathbf{G}\mathbf{a}_q - \Delta t_\ell \mathbf{a}_v + \mathbf{a}_i\right) = c_{i\ell}\,{}^{C_\ell}\mathbf{p}_i,\ \forall i, \ell \tag{114}$$

for some scalars $c_{i\ell}$. Using (28), we obtain

$$\Delta t_\ell^2 \mathbf{G}\mathbf{a}_q - \Delta t_\ell \mathbf{a}_v + \mathbf{a}_i = c_{ij}({}^G\mathbf{p}_i - {}^G\mathbf{p}_{C_\ell}),\ \forall i, \ell \tag{115}$$

Note that the above condition can be interpreted as a condition on the motion of the camera. For example, if the camera is moving with a constant acceleration $\mathbf{G}\mathbf{a}_q$, initial velocity $\mathbf{a}_v$ and initial position ${}^G\mathbf{p}_0$, and we choose $c_{i\ell} = -1$, and $\mathbf{a}_i = {}^G\mathbf{p}_0 - {}^G\mathbf{p}_i$, then the above conditions will be satisfied. However, for general camera motion, and when multiple features are observed, the above condition cannot be met for nonzero values of $c_{i\ell}, \mathbf{a}_q, \mathbf{a}_v, \mathbf{a}_i$ [26]. Thus, for general camera motion, $\breve{\mathbf{T}}_\ell^{(i)}\mathbf{a} = \mathbf{0}, \forall i, \ell$, requires $\mathbf{a} = \mathbf{0}$, which shows that the matrix with block rows $\breve{\mathbf{T}}_\ell^{(i)}$ has full column rank. This completes the proof.

D Nullspace physical interpretation

We have shown that the nullspace of the observability matrix in the "ideal" MSCKF is of dimension 4, and is spanned by the column vectors of the matrix in (39). If we write (39) as $\mathbf{N} = [\,\mathbf{n}_1\ \ \mathbf{n}_2\ \ \mathbf{n}_3\ \ \mathbf{n}_4\,]$, then:

$$\mathcal{N}\left(\breve{\mathcal{O}}\right) = \mathsf{span}\begin{bmatrix}\mathbf{n}_1 & \mathbf{n}_2 & \mathbf{n}_3 & \mathbf{n}_4\end{bmatrix} \tag{116}$$

To gain a better understanding of the physical interpretation of the basis of $\mathcal{N}\left(\breve{\mathcal{O}}\right)$, let us examine what changes in the state each of the four vectors $\mathbf{n}_i$ corresponds to. First, note that if, starting from an initial state $\mathbf{x}$, we modify it as $\mathbf{x}' = \mathbf{x} + c_1\mathbf{n}_1 + c_2\mathbf{n}_2 + c_3\mathbf{n}_3$, then the state $\mathbf{x}'$ will have the same values for the IMU orientation and velocity, but the position of the IMU and the positions of all features will be changed by the vector $[c_1\ \ c_2\ \ c_3]^T$. Thus, the first three columns in $\mathbf{N}$ correspond to shifts of the entire state vector. On the other hand, if we rotate the state vector $\mathbf{x}$ by a small angle, c, about gravity, we can write the resulting state as

$$\mathbf{x}' = \begin{bmatrix} {}^I_G\bar{\mathbf{q}}' \\ {}^G\mathbf{p}' \\ {}^G\mathbf{v}' \\ {}^G\mathbf{p}'_{f_1} \\ \vdots \\ {}^G\mathbf{p}'_{f_N} \end{bmatrix} = \begin{bmatrix} {}^I_G\bar{\mathbf{q}} \otimes {}^G_{G'}\bar{\mathbf{q}} \\ \mathbf{Diag}\left({}^{G'}_G\mathbf{R}\right)\begin{bmatrix} {}^G\mathbf{p} \\ {}^G\mathbf{v} \\ {}^G\mathbf{p}_{f_1} \\ \vdots \\ {}^G\mathbf{p}_{f_N} \end{bmatrix} \end{bmatrix} \tag{117}$$

where the rotation matrix ${}^{G'}_{G}\mathbf{R}$ expresses the applied rotation, and $\mathbf{Diag}(\cdot)$ denotes a block diagonal matrix. To show that $\mathbf{n}_4$ corresponds to rotations about gravity, we will show that the difference between $\mathbf{x}$ and $\mathbf{x}'$ can be written, to a first-order approximation, as a multiple of $\mathbf{n}_4$. We start by noting that, since the rotation angle is small, ${}^{G'}_{G}\mathbf{R}$ can be approximated as:

$$
{}^{G'}_{G}\mathbf{R} \simeq \mathbf{I}_3 - c\lfloor {}^{G}\bar{\mathbf{g}}\times\rfloor \tag{118}
$$

where ${}^{G}\bar{\mathbf{g}}$ is the unit vector along gravity. Using this result, we can write:

$$
\begin{bmatrix} {}^{G}\mathbf{p} \\ {}^{G}\mathbf{v} \\ {}^{G}\mathbf{p}_{f_1} \\ \vdots \\ {}^{G}\mathbf{p}_{f_N} \end{bmatrix} - \begin{bmatrix} {}^{G}\mathbf{p}' \\ {}^{G}\mathbf{v}' \\ {}^{G}\mathbf{p}'_{f_1} \\ \vdots \\ {}^{G}\mathbf{p}'_{f_N} \end{bmatrix} \simeq \begin{bmatrix} {}^{G}\mathbf{p} \\ {}^{G}\mathbf{v} \\ {}^{G}\mathbf{p}_{f_1} \\ \vdots \\ {}^{G}\mathbf{p}_{f_N} \end{bmatrix} - \left(\mathbf{I}_3 - c\lfloor {}^{G}\bar{\mathbf{g}}\times\rfloor\right) \begin{bmatrix} {}^{G}\mathbf{p} \\ {}^{G}\mathbf{v} \\ {}^{G}\mathbf{p}_{f_1} \\ \vdots \\ {}^{G}\mathbf{p}_{f_N} \end{bmatrix} = \frac{c}{||{}^{G}\mathbf{g}||_2} \begin{bmatrix} -\lfloor {}^{G}\mathbf{p}\times\rfloor {}^{G}\mathbf{g} \\ -\lfloor {}^{G}\mathbf{v}\times\rfloor {}^{G}\mathbf{g} \\ -\lfloor {}^{G}\mathbf{p}_{f_1}\times\rfloor {}^{G}\mathbf{g} \\ -\lfloor {}^{G}\mathbf{p}_{f_2}\times\rfloor {}^{G}\mathbf{g} \\ \vdots \\ -\lfloor {}^{G}\mathbf{p}_{f_N}\times\rfloor {}^{G}\mathbf{g} \end{bmatrix} \tag{119}
$$

Moreover, if we denote by $\delta\boldsymbol{\theta}$ the orientation difference between $\mathbf{x}$ and $\mathbf{x}'$ in (117), we obtain

$$
\begin{aligned}
{}^{I}_{G'}\mathbf{R} &\simeq (\mathbf{I}_3 - \lfloor\delta\boldsymbol{\theta}\times\rfloor){}^{I}_{G}\mathbf{R} && (120)\\
&= (\mathbf{I}_3 - \lfloor\delta\boldsymbol{\theta}\times\rfloor){}^{I}_{G'}\mathbf{R}\big(\mathbf{I}_3 - c\lfloor {}^{G}\bar{\mathbf{g}}\times\rfloor\big) && (121)\\
&\simeq {}^{I}_{G'}\mathbf{R} - {}^{I}_{G'}\mathbf{R}\lfloor {}^{I}_{G'}\mathbf{R}^T\delta\boldsymbol{\theta}\times\rfloor - c\,{}^{I}_{G'}\mathbf{R}\lfloor {}^{G}\bar{\mathbf{g}}\times\rfloor && (122)
\end{aligned}
$$

From the last expression we obtain $\lfloor {}^{I}_{G'}\mathbf{R}^T\delta\boldsymbol{\theta}\times\rfloor = c\lfloor {}^{G}\bar{\mathbf{g}}\times\rfloor$, and thus

$$
\begin{aligned}
\delta\boldsymbol{\theta} &= c\cdot {}^{I}_{G'}\mathbf{R}\,{}^{G}\bar{\mathbf{g}} \\
&= c\cdot {}^{I}_{G}\mathbf{R}\,{}^{G}_{G'}\mathbf{R}\,{}^{G}\bar{\mathbf{g}} \\
&= c\cdot {}^{I}_{G}\mathbf{R}\,{}^{G}\bar{\mathbf{g}} \\
&= \frac{c}{||{}^{G}\mathbf{g}||_2}{}^{I}_{G}\mathbf{R}\,{}^{G}\mathbf{g}
\end{aligned} \tag{123}
$$

where we have used the fact that ${}^{G}_{G'}\mathbf{R}\,{}^{G}\bar{\mathbf{g}} = {}^{G}\bar{\mathbf{g}}$, since the rotation ${}^{G'}_{G}\mathbf{R}$ occurs about the direction of gravity. The results of (123) and (119) show that if we apply a small rotation about gravity to obtain $\mathbf{x}'$ from $\mathbf{x}$, the difference between the two states is given by $\frac{c}{||{}^{G}\mathbf{g}||_2}\mathbf{n}_4$.

E Rank of the MSCKF observability matrix

In this section, we prove that the dimension of the nullspace of the MSCKF observability matrix is 3. Similarly to the analysis in Appendix C, we apply the same sequence of elementary column operations to transform each block row of the observability matrix in (40) into:

$$
\mathcal{O}_\ell^{(i)} \sim \mathbf{M}_\ell^{(i)} \left[\begin{array}{ccc|ccccc} \Delta t_\ell^2\lfloor {}^{G}\mathbf{g}\times\rfloor + \Delta\boldsymbol{\Gamma}_\ell^{(i)} & \mathbf{0}_3 & \Delta t_\ell \mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3 \end{array}\right] \tag{124}
$$

At this point, we see that the fourth to sixth columns of the matrix $\mathcal{O}_\ell^{(i)}$ are all zero, which indicates that the dimension of the nullspace of the MSCKF observability matrix is at least three. By omitting zero columns, (124) becomes:

$$
\mathcal{O}_\ell^{(i)} \sim \mathbf{M}_\ell^{(i)} \left[\begin{array}{cc|ccccc} \Delta t_\ell^2\lfloor {}^{G}\mathbf{g}\times\rfloor + \Delta\boldsymbol{\Gamma}_\ell^{(i)} & \Delta t_\ell \mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3 \end{array}\right] = \mathbf{T}_\ell^{(i)} \tag{125}
$$

To prove that $\mathcal{O}$ is rank deficient by three, we need to show that the matrix with block rows $\mathbf{T}_\ell^{(i)}$ has full column rank. We start by defining:

$$
\mathbf{Y}_\ell^{(i)} = \mathbf{M}_\ell^{(i)} \left[\begin{array}{c|ccccc} \Delta t_\ell \mathbf{I}_3 & \mathbf{0}_3 & \cdots & \mathbf{I}_3 & \cdots & \mathbf{0}_3 \end{array}\right] \tag{126}
$$

The matrix with block rows $\mathbf{Y}_\ell^{(i)}$ has full column rank, as shown in Appendix C. Next, we observe that the terms $\Delta\boldsymbol{\Gamma}_\ell^{(i)}$, which appear in the first block column in the matrix (125), are random terms. This implies that, the three columns of this matrix are linearly independent of the columns of $\mathbf{Y}_\ell^{(i)}$. This completes the proof.

References

[1] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Rome, Italy, Apr. 2007, pp. 3565–3572.

[2] E. Jones and S. Soatto, "Visual-inertial navigation, mapping and localization: A scalable real-time causal approach," *International Journal of Robotics Research*, vol. 30, no. 4, pp. 407–430, Apr. 2011.

[3] N. Trawny, A. I. Mourikis, S. I. Roumeliotis, A. E. Johnson, and J. Montgomery, "Vision-aided inertial navigation for pin-point landing using observations of mapped landmarks," *Journal of Field Robotics*, vol. 24, no. 5, pp. 357–378, May 2007.

[4] K. Konolige, M. Agrawal, and J. Sola, "Large-scale visual odometry for rough terrain," in *Proceedings of the International Symposium of Robotics Research*, Flagstaff, AZ, Nov. 2011, pp. 201–212.

[5] A. Wu, E. Johnson, and A. Proctor, "Vision-aided inertial navigation for flight control," *AIAA Journal of Aerospace Computing, Information, and Communication*, vol. 2, no. 9, pp. 348–360, Sep. 2005.

[6] J. Civera, A. Davison, and J. Montiel, "Inverse depth parametrization for monocular SLAM," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 932–945, Oct. 2008.

[7] P. Pinies and J. D. Tardos, "Scalable SLAM building conditionally independent local maps," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, San Diego, CA, Nov. 2007, pp. 3466–3471.

[8] D. Nister, O. Naroditsky, and J. Bergen, "Visual odometry," in *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, Washington, DC, Jun. 2004, pp. 652–659.

[9] J. Kelly, S. Saripalli, and G. Sukhatme, "Combined visual and inertial navigation for an unmanned aerial vehicle," in *Field and Service Robotics*, vol. 42, Jun. 2008, pp. 255–264.

[10] T. Dong-Si and A. I. Mourikis, "Motion tracking with fixed-lag smoothing: Algorithm and consistency analysis," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Shanghai, China, May 2011, pp. 5655 – 5662.

[11] G. Sibley, L. Matthies, and G. Sukhatme, "Sliding window filter with application to planetary landing," *Journal of Field Robotics*, vol. 27, no. 5, pp. 587–608, Sep. 2010.

[12] K. Konolige and M. Agrawal, "FrameSLAM: From bundle adjustment to real-time visual mapping," *IEEE Transactions on Robotics*, vol. 24, no. 5, pp. 1066 –1077, Oct. 2008.

[13] A. Davison, I. Reid, N. Molton, and O. Stasse, "MonoSLAM: Real-time single camera SLAM," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 6, pp. 1052–1067, Jun. 2007.

[14] R. Munguia and A. Grau, "Monocular SLAM for visual odometry," in *Proceedings of the IEEE International Symposium on Intelligent Signal Processing*, Alcala Henares, Spain, Oct. 2007, pp. 1–6.

[15] D. D. Diel, P. DeBitetto, and S. Teller, "Epipolar constraints for vision-aided inertial navigation," in *IEEE Workshop on Motion and Video Computing*, Breckenridge, CO, Jan. 2005, pp. 221–228.

[16] A. I. Mourikis, "Characterization and optimization of the accuracy of mobile robot localization," Ph.D. dissertation, University of Minnesota, June 2008.

[17] Y. Bar-Shalom, X. R. Li, and T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*. John Wiley & Sons, 2001.

[18] G. P. Huang, A. I. Mourikis, and S. I. Roumeliotis, "Analysis and improvement of the consistency of extended Kalman filter-based SLAM," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Pasadena, CA, May 2008, pp. 473–479.

[19] G. Huang, A. Mourikis, and S. Roumeliotis, "Observability-based rules for designing consistent EKF SLAM estimators," *The International Journal of Robotics Research*, vol. 29, no. 5, pp. 502–529, Apr. 2010.

[20] J. Kelly and G. Sukhatme, “Visual-inertial sensor fusion: Localization, mapping and sensor-to-sensor self-calibration,” *International Journal of Robotics Research*, vol. 30, no. 1, pp. 56–79, Jan. 2011.

[21] N. Trawny and S. Roumeliotis, “Indirect Kalman filter for 6D pose estimation,” *University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep*, vol. 2, 2005.

[22] J. Farrell, *Aided navigation: GPS with high rate sensors*. Mcgraw-Hill, 2008.

[23] R. Simpson, J. Cullip, and J. Revell, “The Cheddar Gorge data set,” Tech. Rep., 2011.

[24] P. S. Maybeck, *Stochastic Models, Estimation, and Control*, ser. Mathematics in Science and Engineering. London: Academic Press, 1979, vol. 141-1.

[25] C. D. Meyer, *Matrix Analysis and Applied Linear Algebra*. Society for Industrial and Applied Mathematics, 2000.

[26] T.-C. Dong-Si and A. I. Mourikis, “Motion tracking with fixed-lag smoothing: Algorithm and consistency analysis,” Dept. of Electrical Engineering, University of California, Riverside, Tech. Rep., 2010, http://www.ee.ucr.edu/∼mourikis/tech_reports/fixed_lag.pdf.