Estimator Initialization in Vision-aided Inertial Navigation with Unknown Camera-IMU Calibration

Tue-Cuong Dong-Si and Anastasios I. Mourikis
Dept. of Electrical Engineering, University of California, Riverside
E-mail: tdongsi@ee.ucr.edu, mourikis@ee.ucr.edu

Abstract— **This paper focuses on motion estimation using inertial measurements and observations of naturally occurring point features. To date, this task has primarily been addressed using filtering methods, which track the system state starting from known initial conditions. However, when no prior knowledge of the initial system state is available, (e.g., at the onset of the system's operation), the existing approaches are not applicable. To address this problem, in this work we present algorithms for computing the system's observable quantities (platform attitude and velocity, feature positions, and IMU-camera calibration) directly from the sensor measurements, without any prior knowledge. A key contribution of this work is a convex-optimization based algorithm for computing the rotation matrix between the camera and IMU. We show that once this rotation matrix has been computed, all remaining quantities can be determined by solving a quadratically constrained least-squares problem. To increase their accuracy, the initial estimates are refined by an iterative maximum-likelihood estimator.**

I. Introduction

In recent years, there has been growing interest in methods for motion estimation using visual and inertial measurements, a task often termed termed *vision-aided inertial navigation* (see, e.g. [1]–[7] and references therein). Many reasons exist for this. First, both cameras and MEMS inertial measurement units (IMUs) are compact, inexpensive, and have low power requirements. Second, these sensors can operate in virtually any environment, and allow for full-3D pose estimation, thus providing a very versatile solution for navigation. Third, in recent years we have seen a proliferation of devices (e.g., mobile phones) that contain both cameras and inertial sensors as part of their standard sensor payload. These devices are gradually becoming ubiquitous, and necessitate new techniques that will allow high-precision navigation in GPS-denied environments.

The vast majority of existing techniques for navigation using camera and IMU measurements employ either a recursive Bayesian estimation approach [1]–[4], or a smoothing formulation [5]. In both cases, an accurate initial guess (prior estimate) for the state is necessary for reliable estimation. This is due to the fact that both types of methods rely on linearization of the measurement models, and thus in the absence of an accurate initial estimate, large linearization errors can lead to divergence. In current practice, to initialize any of the state estimation methods discussed above, one typically uses domain-specific knowledge on a case-by-case basis. For instance, in certain applications, additional sensors (e.g., inclinometer and/or GPS) may be available, or it may be known that the platform is initially at rest. However, such methods are not generally applicable.

To address this limitation, in recent work [8]–[10] algorithms for initializing the state of the system based only on observations of naturally occurring features were proposed. These methods provide initial estimates for the attitude (roll and pitch) of the moving platform, its velocity, the positions of the features observed by the camera, and, in the case of [9], for the accelerometer bias. For computing these estimates, these methods assume that the camera-to-IMU transformation (rotation and translation) is known a priori. While this may be a valid requirement in cases where the platform used for navigation is known, it is not always met. Consider for example the case where the user of a mobile phone wants to use the device for indoor navigation. Ideally, the user should be able to simply download an application to his/her device, and be immediately able to use it for navigation, without any lengthy initial setup process. The easiest way to achieve this would be for the navigation algorithm to be able to estimate all necessary parameters, *including* the camera-to-IMU transformation, without the need for a prior initial guess. This is the problem addressed in this paper.

We here propose algorithms that employ the observations of naturally occurring point features, in conjunction with the inertial measurements, for estimating (i) the camera attitude, (ii) the camera velocity, (iii) the position of the features, and (iv) the camera-to-IMU transformation. A straightforward approach to this problem would be to formulate it as a nonlinear-least-squares problem, where we try to minimize the features' reprojection errors with respect to the unknown parameters. This is certainly a valid approach, but its limitation is that for it to converge to meaningful estimates, it requires a good initial guess for the unknowns.

The main contribution of this paper are methods for providing such an initial estimate. As shown in previous work [8], [9], the camera velocity and attitude as well as the feature positions can be analytically computed by solving an appropriately formulated linear system. We here show that the camera-to-IMU translation can also be computed by a similar linear system, while for the camera-to-IMU rotation we present two different methods: The first requires five or more features observed in the images, and computes the unit quaternion representing the camera-to-IMU rotation as the solution to a linear least-squares problem. The second method requires just two (or more) features, and employs a sequence of convex problems for obtaining the solution.

The initial estimates obtained as discussed above have the advantage that they are computed using the sensor data directly, but are not statistically optimal. Therefore, they are subsequently refined using a maximum-likelihood estimator (MLE), whose formulation leads to a nonlinear least squares problem, solved using the Levenberg-Marquardt algorithm. In addition to providing the statistically optimal estimates for the unknown parameters, the MLE also provides us with the covariance matrix of the estimation errors. This makes it possible to test the accuracy of estimation in different settings, and draw practically useful conclusions about the quality of estimation one can expect. These results, as well as tests demonstrating the performance of both the direct solution methods and the iterative MLE, are presented in Section VII.

II. Related work

The observability properties of the vision-aided inertial navigation system have been examined in [1], [2]. These works show that, in the absence of reference points with known global coordinates, the global position of the IMU, as well as the rotation about the axis of gravity (i.e., the yaw) are not observable. On the other hand, the following quantities are *in general* observable:

(O1) The IMU attitude with respect to the horizontal plane (i.e., the roll and pitch),

(O2) The IMU trajectory (position, velocity, and orientation) with respect to the initial IMU frame,

(O3) The feature positions with respect to the initial IMU frame.

(O4) The transformation between the IMU and camera frames (ie., the camera-to-IMU calibration), and

(O5) The IMU gyroscope and accelerometer biases.

As discussed in Section I, our focus in this paper is on estimating the quantities (O1)-(O4) above. We assume that an estimate for the IMU biases is already available (e.g., these biases can be assumed to be close to zero initially, and then high-precision estimates for them can be computed in the MLE refinement described in Section VI). The observability results of [1], [2] prove that (barring singular trajectories such as constant-velocity motion) we are able to estimate the quantities of interest, but do not show how these can be computed directly from the sensor data and without prior estimates.

If the camera-to-IMU transformation was known, one could apply the methods of [8], [9] to estimate the quantities (O1)-(O3). To compute this transformation, [11] and [12] present methods that rely on observation of a known calibration pattern (and the use of a specialized turntable in the case of [11]). While these approaches would be well-suited for a laboratory setting, the need for specialized equipment makes them less suitable for widespread use. In [1], [2], [13], [14], Kalman filter-based estimators are employed for estimating the camera-to-IMU transformation, but since these require a good initial guess, they are not applicable in the case of interest where no prior knowledge exists. In contrast to all aforementioned approaches, we here present methods for estimating the camera-to-IMU transformation directly from the sensor measurements, without the need for prior knowledge, and using only observations of naturally occurring features. In what follows, we present the details of our work.

III. Problem formulation

In this section we present the problem formulation and the measurement equations for the camera and IMU. We show that all the quantities we seek to estimate ((O1)-(O4) defined in Section II) can be linearly estimated, *except* for the rotation between the camera and IMU. In Section IV we show how this rotation can be recovered from the measurements, and finally in Section VI we present a maximum-likelihood estimator that refines these initial estimates to provide high-precision estimates of all the states in the system.

Consider the case where N images are recorded at the time instants $t_0, t_1, \ldots, t_{N-1}$. By employing a suitable image processing algorithm (e.g., KLT tracking [15] or SIFT keypoint extraction and matching [16]), we track M feature points in the images. In addition to the feature observations, the IMU (gyroscope and accelerometer) measurements for the time interval $[t_0, t_{N-1}]$ are available. The measurements of the IMU gyroscopes and accelerometers are given by the following equations [17]:

$$\boldsymbol{\omega}_m(t) = {}^B\boldsymbol{\omega}(t) + \mathbf{n}_\omega(t) \tag{1}$$

$$\mathbf{a}_m(t) = {}^B_G\mathbf{C}(t)({}^G\mathbf{a}(t) - {}^G\mathbf{g}) + \mathbf{n}_a(t) \tag{2}$$

where[1] ${}^B\boldsymbol{\omega}(t)$ denotes the 3D rotational velocity vector expressed in the IMU frame, ${}^G\mathbf{a}(t)$ is the IMU acceleration in the global frame, ${}^G\mathbf{g}$ is the gravitational acceleration vector expressed in the global frame, while $\mathbf{n}_\omega$, and $\mathbf{n}_a$ represent the noise in the gyroscope and accelerometer measurements, respectively. We here assume that the biases are known at least approximately (e.g., from prior sensor calibration) and have been removed from the IMU measurements. In the remainder of this section we will ignore the noise in the measurements, and formulate a linear system of equations that will allow us to estimate all quantities except for the rotation matrix between the camera and IMU.

To estimate the IMU orientation change in the interval $[t_0, t_i]$, we integrate the following differential equation [18]:

$$ {}^{B_0}_{B}\dot{\mathbf{C}}(t) = -{}^{B_0}_{B}\mathbf{C}(t)\lfloor\boldsymbol{\omega}_m(t)\times\rfloor \quad , \quad {}^{B_0}_{B}\mathbf{C}(t_0) = \mathbf{I}_3 \tag{3}$$

in $[t_0, t_i]$. This yields the rotation matrix ${}^{B_0}_{B_i}\mathbf{C} = {}^{B_0}_{B}\mathbf{C}(t_i)$, which describes the IMU rotation between times t_0 and t_i. The global position of the IMU at time t_i is given by:

$$ {}^G\mathbf{p}(t_i) = {}^G\mathbf{p}(t_0) + {}^G\mathbf{v}(t_0)\Delta t_i + \int_{t_0}^{t_i}\int_{t_0}^{\tau} {}^G\mathbf{a}(\varsigma)d\varsigma d\tau $$

where ${}^G\mathbf{p}(t_0)$ is the initial position, ${}^G\mathbf{v}(t_0)$ is the initial velocity, and $\Delta t_i = t_i - t_0$. Using (2) and the above equation we obtain:

$$ {}^G\mathbf{p}(t_i) = {}^G\mathbf{p}(t_0) + {}^G\mathbf{v}(t_0)\Delta t_i + {}^G\mathbf{g}\frac{\Delta t_i^2}{2} + {}^G_{B_0}\mathbf{C}\,\mathbf{s}(t_i) \tag{4}$$

[1] Throughout this paper, the IMU (body) frame is denoted by $\{B\}$, the camera frame by $\{C\}$, and the global (inertial) frame by $\{G\}$. ${}^X\mathbf{y}$ denotes the vector $\mathbf{y}$ expressed with respect to frame $\{X\}$, and ${}^X_Y\mathbf{C}$ denotes the rotation matrix transforming vectors from frame $\{Y\}$ into frame $\{X\}$. $\mathbf{I}_n$ is the $n\times n$ identity matrix, and finally, $\lfloor\mathbf{y}\times\rfloor$ is the skew-symmetric matrix associated with the 3×1 vector $\mathbf{y}$.

where $$\mathbf{s}(t_i) = \int_{t_0}^{t_i} \int_{t_0}^{\tau} {}^{B_0}_{B}\mathbf{C}(\varsigma)\mathbf{a}_m(\varsigma)\, d\varsigma d\tau \tag{5}$$

Next we note that the IMU position at time t_i with respect to $\{B_0\}$ is given by ${}^{B_0}\mathbf{p}_{B_i} = {}^{G}_{B_0}\mathbf{C}^T({}^{G}\mathbf{p}(t_i) - {}^{G}\mathbf{p}(t_0))$. Using this expression, we can re-arrange (4) to obtain:

$$ {}^{B_0}\mathbf{p}_{B_i} = {}^{B_0}\mathbf{v}_0\Delta t_i + {}^{B_0}\mathbf{g}\frac{\Delta t_i^2}{2} + \mathbf{s}(t_i) \tag{6}$$

where ${}^{B_0}\mathbf{v}_0 = {}^{G}_{B_0}\mathbf{C}^T\, {}^{G}\mathbf{v}(t_0)$ is the IMU velocity at time t_0 expressed with respect to frame $\{B_0\}$, and ${}^{B_0}\mathbf{g} = {}^{G}_{B_0}\mathbf{C}^T\, {}^{G}\mathbf{g}$ is the gravity vector expressed with respect to the same frame. Eq. (6) will be useful in what follows, as it contains only observable quantities, and none of the global (unobservable) ones. We next show how the camera measurements can be expressed as a function of the quantities appearing in (6).

Assuming an intrinsically calibrated camera, the observation of the j-th feature at time t_i is described by the perspective camera model:

$$\mathbf{z}_{ij} = \begin{bmatrix} u_{ij} \\ v_{ij} \end{bmatrix} = \begin{bmatrix} \frac{{}^{C_i}x_j}{{}^{C_i}z_j} \\ \frac{{}^{C_i}y_j}{{}^{C_i}z_j} \end{bmatrix} + \mathbf{n}_{ij}, \tag{7}$$

where ${}^{C_i}\mathbf{p}_j = \begin{bmatrix} {}^{C_i}x_j & {}^{C_i}y_j & {}^{C_i}z_j \end{bmatrix}^T$ is the position of the j-th feature with respect to the camera frame at time t_i, and $\mathbf{n}_{ij}$ is the measurement noise. We use the set $\mathcal{S}_m$ to describe all pairs of indices $\{i, j\}$ that describe the available measurements.

Using basic properties of frame transformations, we can express the vector ${}^{C_i}\mathbf{p}_j$ as follows:

$$ {}^{C_i}\mathbf{p}_j = {}^{C}_{B}\mathbf{C}\ {}^{B_i}_{B_0}\mathbf{C}\ ({}^{B_0}\mathbf{p}_j - {}^{B_0}\mathbf{p}_{B_i}) + {}^{C}\mathbf{p}_B \tag{8}$$

where ${}^{B_0}\mathbf{p}_j$ is the position of the feature with respect to $\{B_0\}$, while $\{{}^{C}_{B}\mathbf{C}, {}^{C}\mathbf{p}_B\}$ denotes the constant transformation (rotation and translation) between the IMU and camera. Using (6), we can rewrite (8) as:

$$ {}^{C_i}\mathbf{p}_j = {}^{C}_{B}\mathbf{C}\ {}^{B_i}_{B_0}\mathbf{C} \left({}^{B_0}\mathbf{p}_j - {}^{B_0}\mathbf{v}_0\Delta t_i - {}^{B_0}\mathbf{g}\frac{\Delta t_i^2}{2} - \mathbf{s}(t_i)\right) + {}^{C}\mathbf{p}_B \tag{9}$$

On the right-hand side of above equation the unknown quantities are the vectors ${}^{B_0}\mathbf{p}_j$, ${}^{B_0}\mathbf{v}_0$,${}^{B_0}\mathbf{g}$ and the IMU-camera extrinsic calibration $\{{}^{C}_{B}\mathbf{C}, {}^{C}\mathbf{p}_B\}$, while the rotation matrix ${}^{B_i}_{B_0}\mathbf{C}$ and the vector $\mathbf{s}(t_i)$ are computed using the IMU measurements. We now employ (7), to obtain (ignoring the measurement noise):

$$\begin{bmatrix} 1 & 0 & -u_{ij} \\ 0 & 1 & -v_{ij} \end{bmatrix} {}^{C_i}\mathbf{p}_j = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{10}$$

Using (9) and re-arranging terms, the above equation can be written as $\mathbf{A}_{ij}\mathbf{x} = \mathbf{b}_{ij}$, where $\mathbf{x}$ is the following $(3M+9)\times 1$ vector:

$$\mathbf{x} = \begin{bmatrix} {}^{B_0}\mathbf{p}_1^T & \dots & {}^{B_0}\mathbf{p}_M^T & {}^{B_0}\mathbf{v}_0^T & {}^{C}\mathbf{p}_B^T & {}^{B_0}\mathbf{g}^T \end{bmatrix}^T \tag{11}$$

and

$$\mathbf{A}_{ij} = \begin{bmatrix} 1 & 0 & -u_{ij} \\ 0 & 1 & -v_{ij} \end{bmatrix} {}^{C}_{B}\mathbf{C}\ {}^{B_i}_{B_0}\mathbf{C}\ \mathbf{T}_{ij} \tag{12}$$

$$\mathbf{b}_{ij} = \begin{bmatrix} 1 & 0 & -u_{ij} \\ 0 & 1 & -v_{ij} \end{bmatrix} {}^{C}_{B}\mathbf{C}\ {}^{B_i}_{B_0}\mathbf{C}\ \mathbf{s}(t_i) \tag{13}$$

$$\mathbf{T}_{ij} = \begin{bmatrix} \cdots & \underbrace{\mathbf{I}_3}_{j\text{−th block}} & \cdots & -\Delta t_i\mathbf{I}_3 & {}^{B_0}_{B_i}\mathbf{C}\ {}^{B}_{C}\mathbf{C} & -\frac{\Delta t_i^2}{2}\mathbf{I}_3 \end{bmatrix} \tag{14}$$

The equation $\mathbf{A}_{ij}\mathbf{x} = \mathbf{b}_{ij}$ is derived from the observation of the j-th feature in the i-th image. By collecting the equations resulting from all feature and IMU measurements, we obtain the linear system

$$\mathbf{A}\mathbf{x} = \mathbf{b} \tag{15}$$

where $\mathbf{A}$ is a matrix with block rows $\mathbf{A}_{ij}$, and $\mathbf{b}$ is a block vector with block elements $\mathbf{b}_{ij}$, for all $\{i, j\} \in \mathcal{S}_m$.

Let us now examine the properties of the linear system in (15). First, we note that the vector $\mathbf{x}$ contains *all* the unknowns that are necessary to recover the observable terms (O1)-(O4), except for the camera-to-IMU rotation. Specifically, knowing the direction of gravity in the local frame is equivalent to knowing the IMU's attitude with respect to the horizontal plane (quantity (O1)). Moreover, if ${}^{B_0}\mathbf{v}_0$ and ${}^{B_0}\mathbf{g}$ are known, then we can estimate the IMU trajectory in the local frame (quantity (O2)), using (6). The feature positions (O3) and the camera-to-IMU translation (part of (O4)) are contained explicitly in $\mathbf{x}$. Thus, solving this linear system would allow us to determine all the parameters we seek, except for ${}^{C}_{B}\mathbf{C}$. From (12)-(14) we see that the matrix $\mathbf{A}$ and the vector $\mathbf{b}$ can be computed using the feature measurements, the IMU measurements, and the rotation matrix ${}^{C}_{B}\mathbf{C}$. We therefore see that if we are able to determine the camera-to-IMU rotation, we can easily recover all remaining quantities.

A. Number of measurements required

An important consideration is to determine the number of images and features required in order to be able to estimate all the unknown parameters in the system. The first question we examine is the minimum number of images. In [19] we show that if the number of images is $N \leq 3$, then the matrix $\mathbf{A}$ in the linear system (15) is rank-deficient by at least 3. In other words, *even if* ${}^{C}_{B}\mathbf{C}$ was known, we would still not be able to uniquely determine the unknown parameters in the system. Thus, the minimum number of images needed is $N = 4$.

To determine the number of features required, we employ a counting argument: for each feature measurement in each image, we obtain 2 scalar equations from $\mathbf{A}_{ij}\mathbf{x} = \mathbf{b}_{ij}$. Thus, with N images and M features, the number of measurement constraints is $2NM$. On the other hand the number of observable unknowns is $3M+11$ ($3M$ for the feature positions, 6 for the camera-to-IMU transformation, 3 for the initial velocity, and 2 for the roll and pitch). To be able to uniquely determine all the unknowns, we must have $2MN \geq 3M+11$, from which we see that the minimum number of features needed is $M = 3$, if four images are available, $M = 2$, if five images are available, and $M = 1$, if seven or more images are available. In all these minimal cases, the number of measurements is higher than the number of unknowns, so the problem is over-constrained, and a unique solution can be computed.

IV. Determining the camera-to-IMU rotation

In this section, we present two methods for computing the camera-to-IMU rotation matrix. The first one is applicable when at least $M = 5$ features are observed. In that case, we can estimate the relative camera orientation between different images, ${}^{C_i}_{C_j}\mathbf{C}$, using image-based motion estimation algorithms such as [20] (the same can be accomplished if $M = 4$ features are observed, but the algorithms involved are significantly more complex [21]). When $M < 4$, the feature measurements alone cannot be used to estimate the camera rotation. For that case, we describe in Section IV-B a method that only requires $M \geq 2$ features to recover the camera-to-IMU rotation matrix.

A. Solution for the case $M \geq 5$

We first consider the case where enough features are observed so that we can recover the relative camera orientation between different time instants, ${}^{C_i}_{C_j}\mathbf{C}$, using only the feature observations. By using the IMU measurements, we can estimate the IMU orientation change ${}^{B_i}_{B_j}\mathbf{C}$ in the same time interval, via (3). We can then employ the following equation for the unknown ${}^{C}_{B}\mathbf{C}$:

$$ {}^{C_i}_{C_j}\mathbf{C} = {}^{C}_{B}\mathbf{C}\,{}^{B_i}_{B_j}\mathbf{C}\,{}^{B}_{C}\mathbf{C} \Rightarrow {}^{C_i}_{C_j}\mathbf{C}\,{}^{C}_{B}\mathbf{C} = {}^{C}_{B}\mathbf{C}\,{}^{B_i}_{B_j}\mathbf{C} \qquad (16) $$

To recover the matrix ${}^{C}_{B}\mathbf{C}$, we can transform the above equations into their equivalent unit-quaternion representation [22], [23]. Specifically, using this representation, we have [24]:

$$ \begin{aligned} &{}^{C_i}_{C_j}\bar{\mathbf{q}} \otimes {}^{C}_{B}\bar{\mathbf{q}} = {}^{C}_{B}\bar{\mathbf{q}} \otimes {}^{B_i}_{B_j}\bar{\mathbf{q}} \\ \Rightarrow &\mathcal{L}({}^{C_i}_{C_j}\bar{\mathbf{q}})\,{}^{C}_{B}\bar{\mathbf{q}} = \mathcal{R}({}^{B_i}_{B_j}\bar{\mathbf{q}})\,{}^{C}_{B}\bar{\mathbf{q}} \qquad (17) \\ \Rightarrow &\left(\mathcal{L}({}^{C_i}_{C_j}\bar{\mathbf{q}}) - \mathcal{R}({}^{B_i}_{B_j}\bar{\mathbf{q}})\right){}^{C}_{B}\bar{\mathbf{q}} = \mathbf{0} \qquad (18) \end{aligned} $$

where, for a 4×1 unit quaternion $\bar{\mathbf{q}}$, we denote:

$$ \bar{\mathbf{q}} = \begin{bmatrix} q_1 & q_2 & q_3 & q_4 \end{bmatrix}^T = \begin{bmatrix} \mathbf{q}^T & q_4 \end{bmatrix}^T \qquad (19) $$

and

$$ \mathcal{L}(\bar{\mathbf{q}}) = \begin{bmatrix} q_4\mathbf{I}_3 - \lfloor\mathbf{q}\times\rfloor & \mathbf{q} \\ -\mathbf{q}^T & q_4 \end{bmatrix} \qquad (20) $$

$$ \mathcal{R}(\bar{\mathbf{q}}) = \begin{bmatrix} q_4\mathbf{I}_3 + \lfloor\mathbf{q}\times\rfloor & \mathbf{q} \\ -\mathbf{q}^T & q_4 \end{bmatrix} \qquad (21) $$

Eq. (18) is a linear system of the form $\mathbf{B}_{ij}\,{}^{C}_{B}\bar{\mathbf{q}} = \mathbf{0}$, where the unknown is the quaternion ${}^{C}_{B}\bar{\mathbf{q}}$ describing the camera-to-IMU rotation. By using all the available pairs of images, we can construct an over-constrained linear system:

$$ \mathbf{B}\,{}^{C}_{B}\bar{\mathbf{q}} = \mathbf{0} \qquad (22) $$

where $\mathbf{B}$ is a matrix with block rows $\mathbf{B}_{ij}$. The least-squares solution ${}^{C}_{B}\bar{\mathbf{q}}$ is the right unit singular vector corresponding to the smallest singular value of $\mathbf{B}$, and from it we can directly recover the rotation matrix ${}^{C}_{B}\mathbf{C}$ [24]. For the solution to be unique, at least two pairs of images, where the system rotates about different axes, are required [23]. Therefore, we see that at least three images are needed, in which at least five features are tracked, for this method to be able to determine the camera-to-IMU rotation.

B. Solution for $M \geq 2$ points

We now present an alternative method, that can operate for any number of points $M \geq 2$, i.e., it can recover the matrix ${}^{C}_{B}\mathbf{C}$ even when the feature measurements alone cannot be used to determine the relative camera rotation between images.

From the properties of the cross product, we know that for any vectors $\mathbf{a}_1$ and $\mathbf{a}_2$, the following holds: $(\mathbf{a}_1 \times \mathbf{a}_2)^T(\mathbf{a}_1 - \mathbf{a}_2) = 0$. Therefore, for any two features m and n observed by the camera at time instant t_i, we can write

$$ ({}^{C_i}\mathbf{p}_m \times {}^{C_i}\mathbf{p}_n)^T({}^{C_i}\mathbf{p}_m - {}^{C_i}\mathbf{p}_n) = 0 \qquad (23) $$

Next, we use (7) to write ${}^{C_i}\mathbf{p}_j = {}^{C_i}z_j[\mathbf{z}_{ij}^T \quad 1]^T$, and thus:

$$ \begin{aligned} &\left(\begin{bmatrix} \mathbf{z}_{im} \\ 1 \end{bmatrix} \times \begin{bmatrix} \mathbf{z}_{in} \\ 1 \end{bmatrix}\right)^T ({}^{C_i}\mathbf{p}_m - {}^{C_i}\mathbf{p}_n) = 0 \\ \Rightarrow \quad &\mathbf{l}_{imn}^T({}^{C_i}\mathbf{p}_m - {}^{C_i}\mathbf{p}_n) = 0 \qquad (24) \end{aligned} $$

Applying (8) for ${}^{C_i}\mathbf{p}_m$ and ${}^{C_i}\mathbf{p}_n$ and simplifying, we obtain:

$$ \mathbf{l}_{imn}^T\,{}^{C}_{B}\mathbf{C}\,{}^{B_i}_{B_0}\mathbf{C}\left({}^{B_0}\mathbf{p}_m - {}^{B_0}\mathbf{p}_n\right) = 0 \qquad (25) $$

In this last equation, $\mathbf{l}_{imn}^T$ is a known vector computed using the feature measurements, while ${}^{B_i}_{B_0}\mathbf{C}$ is a known matrix computed using the IMU measurements. The matrix ${}^{C}_{B}\mathbf{C}$ and the vector $\Delta\mathbf{p} = {}^{B_0}\mathbf{p}_m - {}^{B_0}\mathbf{p}_n$ are unknown. We can collect all unknowns in a vector $\mathbf{y}$:

$$ \mathbf{y} = [c_1 \;\; \dots \;\; c_9 \;\; p_1 \;\; p_2 \;\; p_3]^T = [\mathbf{c}^T \;\; \Delta\mathbf{p}^T]^T \qquad (26) $$

where $c_i, i = 1\dots 9$ are the elements of the 3×3 rotation matrix ${}^{C}_{B}\mathbf{C}$, and $p_i, i = 1\dots 3$ are the elements of $\Delta\mathbf{p}$. We now see that (25) is a quadratic equation in the elements of $\mathbf{y}$. From each image we can extract one such equation, and thus from N images we obtain N quadratic equations in the elements of $\mathbf{y}$. Additionally, the rotation matrix must satisfy the orthogonality constraints, which are represented by six quadratic equations in $\mathbf{c}$. Finally, since (25) is homogeneous in $\Delta\mathbf{p}$, we must enforce a norm constraint on this vector:

$$ ||\Delta\mathbf{p}||_2 = 1 \Rightarrow \Delta\mathbf{p}^T\Delta\mathbf{p} = 1 \qquad (27) $$

which is another quadratic equation in $\Delta\mathbf{p}$. We thus see that in total, with N images we obtain $N+7$ quadratic constraints in $\mathbf{y}$. Since $\mathbf{y}$ is a 12×1 vector, when at least 5 images are available, we have a number of equations equal to the number of unknowns.

So far only two features were considered. To extend our formulation to the case where $M \geq 2$ features are observed, we note that (25) can be rewritten as:

$$ \mathbf{l}_{imn}^T\,{}^{C}_{B}\mathbf{C}\,{}^{B_i}_{B_0}\mathbf{C}\left(({}^{B_0}\mathbf{p}_m - {}^{B_0}\mathbf{p}_1) - ({}^{B_0}\mathbf{p}_n - {}^{B_0}\mathbf{p}_1)\right) = 0 \qquad (28) $$

Proceeding similarly, we obtain a system of quadratic equations in terms of the following vector of unknowns:

$$ \mathbf{y} = [\mathbf{c}^T \;\; ({}^{B_0}\mathbf{p}_2 - {}^{B_0}\mathbf{p}_1)^T \;\; \dots \;\; ({}^{B_0}\mathbf{p}_M - {}^{B_0}\mathbf{p}_1)^T]^T \qquad (29) $$

where $\mathbf{c}$ is subject to the six orthogonality constraints, while the remaining part of $\mathbf{y}$ is subject to a scale constraint such that its norm equals one. The vector $\mathbf{y}$ contains $n = 3M + 6$ unknowns.

1) Solving the system of equations using convex iteration: The equations derived above form a system of multivariate quadratic equations. Solving such a system algebraically is known to be NP-complete [25], and therefore here we will employ a solution approach based on iterative convex approximations. Specifically, the system of equations in $\mathbf{y}$ can be written as $\mathbf{y}^T\mathbf{F}_i\mathbf{y} = l_i$, $i = 1 \ldots p$, where p is the number of constraints available (12 in the minimal case of $M = 2$), $\mathbf{F}_i$ are real symmetric matrices, and l_i are constants equal to zero or one. Thus, $\mathbf{y}$ can be found by solving the feasibility problem:

$$\begin{aligned} &\text{find} && \mathbf{y} \\ &\text{subject to} && \mathbf{y}^T\mathbf{F}_i\mathbf{y} = l_i \end{aligned} \tag{30}$$

Using the property $\mathbf{y}^T\mathbf{F}_i\mathbf{y} = \text{trace}(\mathbf{y}^T\mathbf{F}_i\mathbf{y}) = \text{trace}(\mathbf{F}_i\mathbf{y}\mathbf{y}^T)$, we can write the above problem equivalently as:

$$\begin{aligned} &\text{find} && \mathbf{Y} \\ &\text{subject to} && \text{trace}(\mathbf{F}_i\mathbf{Y}) = l_i, \quad i = 1, \ldots, p \\ &&& \mathbf{Y} \in \mathbb{S}^n_+, \quad \text{rank}(\mathbf{Y}) = 1 \end{aligned} \tag{31}$$

where $\mathbb{S}^n_+$ denotes the cone of $n \times n$ positive semidefinite matrices. Once the solution to this problem is found, $\mathbf{y}$ is given by $\mathbf{Y} = \mathbf{y}\mathbf{y}^T$. The above feasibility problem can be exactly reformulated by relaxing the rank constraint into an inequality:

$$\begin{aligned} &\text{find} && \mathbf{Y} \\ &\text{subject to} && \text{trace}(\mathbf{F}_i\mathbf{Y}) = l_i, \quad i = 1, \ldots, p \\ &&& \mathbf{Y} \in \mathbb{S}^n_+, \quad \text{rank}(\mathbf{Y}) \leq 1 \end{aligned} \tag{32}$$

Since the only matrix with rank equal to zero is the zero matrix (which is not a solution), the above problem will have the same solution as (31). To obtain the solution to the above problem we will use the results of [26]. Specifically, [26] shows that the solution to (32) can be found by iteratively solving the following two convex optimization problems:

$$\begin{aligned} &\text{minimize} && \text{trace}(\mathbf{Y}\mathbf{W}) \\ &\text{subject to} && \text{trace}(\mathbf{F}_i\mathbf{Y}) = l_i, \quad i = 1, \ldots, p \\ &&& \mathbf{Y} \in \mathbb{S}^n_+ \end{aligned} \tag{33}$$

and

$$\begin{aligned} &\text{minimize} && \text{trace}(\mathbf{Y}^\star\,\mathbf{W}) \\ &\text{subject to} && \mathbf{0} \preceq \mathbf{W} \preceq \mathbf{I}_n \\ &&& \mathbf{W} \in \mathbb{S}^n_+, \quad \text{trace}(\mathbf{W}) = n - 1 \end{aligned} \tag{34}$$

where $\preceq$ denotes matrix inequality in the positive-semidefinite sense. The process is as follows: we select an initial value for the so-called *direction matrix* $\mathbf{W}$ (set to zero in our implementation), and solve problem (33) to obtain $\mathbf{Y}^\star$. Then we use this value in problem (34) find a new value $\mathbf{W}$, and the process is repeated to convergence. In [26] it is shown that when this iteration converges, the solution obtained is the exact solution to the original problem (32) (and thus, to our original problem of finding $\mathbf{y}$). We note that both problems (33) and (34) are semidefinite programs (SDPs), and can be efficiently solved with off-the-shelf algorithms.

2) Addressing the presence of noise: Eq. (25) will only hold exactly if the measurements are perfect. When noise is present, we will have

$$\mathbf{l}'^T_{imn}\,{}^C_B\mathbf{C}\,{}^{B_i}_{B_0}\mathbf{C}\left({}^{B_0}\mathbf{p}_m - {}^{B_0}\mathbf{p}_n\right) = \epsilon_{imn} \tag{35}$$

where ϵ_{imn} is a (small) error. To be able to solve the problem using the convex-iteration formulation presented above, we can compute an upper bound for the error, such that:

$$-\,e^b_{mn} \leq \mathbf{l}'^T_{imn}\,{}^C_B\mathbf{C}\,{}^{B_i}_{B_0}\mathbf{C}\left({}^{B_0}\mathbf{p}_m - {}^{B_0}\mathbf{p}_n\right) \leq e^b_{mn} \tag{36}$$

Thus, we can now write the problem of estimating the unknown vector $\mathbf{y}$ as the feasibility problem:

$$\begin{aligned} &\text{find} && \mathbf{Y} \\ &\text{subject to} && -\,e^b_{ij} \leq \text{trace}(\mathbf{F}_i\,\mathbf{Y}) \leq e^b_{ij}, \quad \forall i, j \\ &&& \text{trace}(\mathbf{F}_i\mathbf{Y}) = l_i, \quad i = 1, \ldots, 7 \\ &&& \mathbf{Y} \in \mathbb{S}^n_+, \quad \text{rank}(\mathbf{Y}) = 1 \end{aligned} \tag{37}$$

where the inequality constraints result from the measurements, while the equality constraints are due to the orthogonality and unit-norm constraints on the elements of $\mathbf{y}$. It is important to note that, as in (31), these constraints define a convex set for $\mathbf{Y}$, and thus the method of [26] can still be applied. Therefore, to find the solution to the problem (37), we use a convex iteration analogous to that of (33) and (34), with the only difference that the equality constraints in (33) are now replaced by the combination of equality and inequality constraints as shown above.

The accuracy with which the vector $\mathbf{y}$ (and therefore the rotation matrix ${}^C_B\mathbf{C}$, which we are interested in) can be estimated depends on the choice of the bounds e^b_{ij}. If too loose bounds are chosen, then the solution obtained will be inaccurate. On the other hand if too small bounds are selected the problem will become infeasible. To address this problem, in our implementation we compute an initial, low estimate for the bounds using the known statistics of the sensor noise. We begin running the convex iteration using these values, and if the problem is infeasible (which results in the iterations stalling [26]), we gradually increase the bounds until convergence succeeds.

V. Solution of the linear system $\mathbf{A}\mathbf{x} = \mathbf{b}$

Once the rotation matrix ${}^C_B\mathbf{C}$ has been determined by one of the two methods described in the previous section, we can then proceed to solve the linear system (15) to recover the remaining unknown parameters. In the presence of noise this system will have no exact solution, and therefore we can instead compute a least-squares solution, i.e., we can minimize the function $||\mathbf{A}\mathbf{x} - \mathbf{b}||_2$. It is a well-known result that the optimal value of $\mathbf{x}$ for this problem is given by $\mathbf{x}^\star = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$. Note however, that this solution does not take advantage of the fact that the norm of the gravitational acceleration vector may be known in advance. To exploit this additional information, we can formulate a *constrained* least-squares problem:

$$\begin{aligned} &\text{minimize} && ||\mathbf{A}\mathbf{x} - \mathbf{b}||_2 = \left|\left|[\mathbf{A}_1 \;\; \mathbf{A}_2]\begin{bmatrix}\mathbf{x}_1 \\ {}^{B_0}\mathbf{g}\end{bmatrix} - \mathbf{b}\right|\right|_2 \\ &\text{subject to} && ||{}^{B_0}\mathbf{g}||_2 = g \end{aligned} \tag{38}$$

where g is the known value of the norm of the gravitational acceleration, $\mathbf{x}_1$ is a vector comprising the landmark positions, IMU velocity, and IMU-camera translation (see (11)), and the partitioning of $\mathbf{A}$ is compatible with that of $\mathbf{x}$. The above problem is a quadratically-constrained least-squares problem. Its optimal solution can be derived using the method of Lagrange multipliers [19], and is given by:

$$\mathbf{x}^\star = \begin{bmatrix} -(\mathbf{A}_1^T\mathbf{A}_1)^{-1}\mathbf{A}_1^T\mathbf{A}_2{}^{B_0}\mathbf{g}^\star + (\mathbf{A}_1^T\mathbf{A}_1)^{-1}\mathbf{A}_1^T\mathbf{b} \\ {}^{B_0}\mathbf{g}^\star \end{bmatrix} \tag{39}$$

with ${}^{B_0}\mathbf{g}^\star = (\mathbf{D} - \lambda\mathbf{I}_3)^{-1}\mathbf{d}$, where

$$\mathbf{D} = \mathbf{A}_2^T\left(\mathbf{I} - \mathbf{A}_1(\mathbf{A}_1^T\mathbf{A}_1)^{-1}\mathbf{A}_1^T\right)\mathbf{A}_2 \tag{40}$$

$$\mathbf{d} = \mathbf{A}_2^T\left(\mathbf{I} - \mathbf{A}_1(\mathbf{A}_1^T\mathbf{A}_1)^{-1}\mathbf{A}_1^T\right)\mathbf{b} \tag{41}$$

and where λ is the smallest solution to the equation:

$$\det\left((\mathbf{D} - \lambda\mathbf{I}_3)^2 - \frac{1}{g^2}\mathbf{d}\mathbf{d}^T\right) = 0 \tag{42}$$

The matrix whose determinant we compute in (42) is a 3×3 matrix, whose elements are quadratic polynomials in λ. Therefore, (42) is a sixth-order polynomial equation in λ. To find the smallest root, one can simply compute all roots (which can be done numerically with very low computational cost), and choose the minimum real one. Our tests have shown that when noise is present, using the known gravity information (i.e., employing the constrained-least-squares solution (39) instead of the unconstrained one) results in substantially improved estimation accuracy.

VI. Maximum-Likelihood Estimator

The direct solutions presented to this point offer the advantage of providing the result without the need for any prior initial guess. However, they are not statistically optimal, as the presence of noise is not properly modelled. To properly account for the noise in the measurements, we formulate a MLE for estimating a parameter vector $\boldsymbol{\theta}$ comprising (i) the IMU state (position, velocity, orientation) at each time instant where an image is recorded, (ii) the positions of all features, $\mathbf{p}_j, j = 1 \ldots M$, and (iii) the camera-to-IMU transformation. Quantities (i)-(ii) are expressed with respect to a frame whose origin coincides with the origin of the initial IMU frame, while its z-axis is aligned with the direction of gravity.

Following standard practice, we model the image measurement noise vector, $\mathbf{n}_{ij}$, as a Gaussian zero-mean random variable, with covariance matrix $\mathbf{R}_{ij}$. Specifically, we use the IMU measurements in the time interval $[t_i, t_{i+1}]$ to compute the change in the IMU state:

$$\mathbf{x}_{\mathrm{IMU}_{i+1}} = \mathbf{f}(\mathbf{x}_{\mathrm{IMU}_i}, \boldsymbol{\omega}_m, \mathbf{a}_m) + \mathbf{w}_i \tag{43}$$

where $\mathbf{w}_i$ is a noise vector, modelled as zero-mean, Gaussian random variable with covariance matrix $\mathbf{Q}_i$. The function $\mathbf{f}$ and the covariance matrix $\mathbf{Q}_i$ are computed using numerical integration of the continuous-time motion equations [18].

Maximizing the likelihood of the measurements is equivalent to maximizing the log-likelihood, which, in turn, is equivalent to minimizing the cost function:

$$c(\boldsymbol{\theta}) = \sum_{i,j}\left|\left|\mathbf{z}_{ij} - \mathbf{h}(\mathbf{x}_{\mathrm{IMU}_i}, \mathbf{p}_j, {}^C_B\mathbf{C}, {}^C\mathbf{p}_B)\right|\right|^2_{\mathbf{R}_{ij}}$$

$$+ \sum_{i=0}^{N-1}\left|\left|\mathbf{x}_{\mathrm{IMU}_{i+1}} - \mathbf{f}(\mathbf{x}_{\mathrm{IMU}_i}, \boldsymbol{\omega}_m, \mathbf{a}_m)\right|\right|^2_{\mathbf{Q}_i} \tag{44}$$

where $\mathbf{h}(\cdot)$ is the function describing the perspective measurement model (see (7)), and $\left|\left|\mathbf{u}\right|\right|^2_{\mathbf{M}} = \mathbf{u}^T\mathbf{M}^{-1}\mathbf{u}$.

```
begin
    if enough features available then
        Use image-based motion estimation to compute
        the camera relative rotation.
        Solve (22) to get ${}^C_B\hat{\bar{\mathbf{q}}}$.
    else
        Initialize error bounds $e^b_{ij}$, set $\mathbf{W} = \mathbf{0}$.
        Compute $\mathbf{F}_i$ in (37) from the measurements.
        while not converged do
            Solve (33) with current $\mathbf{W}$ to get $\mathbf{Y}^\star$.
            Solve (34) with $\mathbf{Y}^\star$ to update $\mathbf{W}$.
            if stall detected then
                Increment error bounds, reset $\mathbf{W}$.
            end
        end
    end
    Obtain ${}^C_B\hat{\mathbf{C}}$ from ${}^C_B\hat{\bar{\mathbf{q}}}$ or $\mathbf{Y}^\star$.
end
begin
    Compute $\mathbf{A}, \mathbf{b}$ in (15) from ${}^C_B\hat{\mathbf{C}}$ and measurements.
    Solve (39)-(42) for ${}^C\hat{\mathbf{p}}_B$, ${}^{B_0}\hat{\mathbf{v}}_{B_0}$, ${}^{B_0}\hat{\mathbf{g}}$, and ${}^{B_0}\hat{\mathbf{p}}_j$.
end
begin
    Minimize (44) using Levenberg-Marquardt starting
    from ${}^C_B\hat{\mathbf{C}}$, ${}^C\hat{\mathbf{p}}_B$, ${}^{B_0}\hat{\mathbf{v}}_{B_0}$, ${}^{B_0}\hat{\mathbf{g}}$, and ${}^{B_0}\hat{\mathbf{p}}_j$ for initial
    states.
end
```

Algorithm 1: Procedure of Estimator Initialization

The cost function $c(\boldsymbol{\theta})$ is nonlinear, and its minimization is carried out iteratively, by application of the Levenberg-Marquardt method [27]. In our testing we have observed that if the direct solutions described in Sections IV and V are used to provide an initial guess for the iterations, the convergence is rapid and requires only a few (typically less than 10) iterations. We also note that, even though the IMU biases were assumed to be known in the derivation of the direct solutions, if desired these biases can be included as unknowns in the MLE, and estimated along with all other parameters.

The full procedure to determine the observable quantities with noisy measurements is summarized in Algorithm 1. The three processing blocks correspond to the three estimation stages, namely IMU-camera rotation estimation (Section IV), constrained least-squares solution (Section V) and MLE (Section VI).

VII. Results

We now present the results of Monte-Carlo simulation trials, which illustrate the accuracy of the methods described in the preceding sections, and the dependence of this accuracy on several parameters of interest. In all the results presented here, the accelerometer and gyroscope measurements are

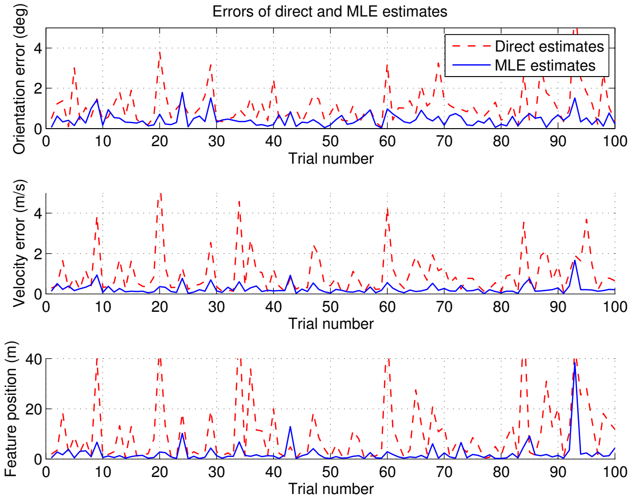


Fig. 1. Comparison of the errors of the direct least-squares solution vs. the MLE.

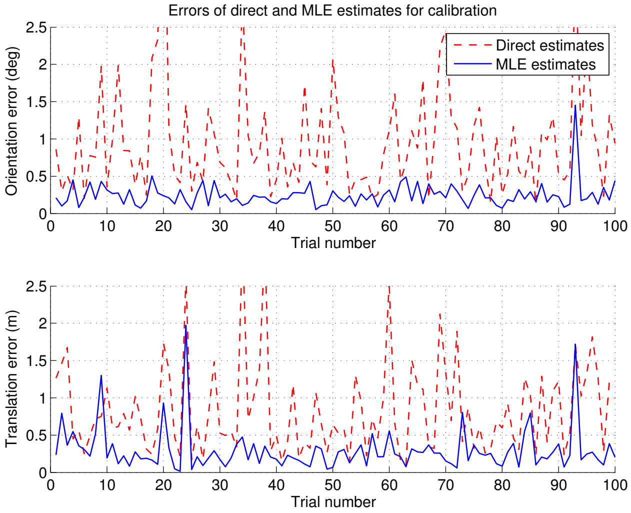


Fig. 2. Comparison of the errors of the direct solution vs. the MLE.

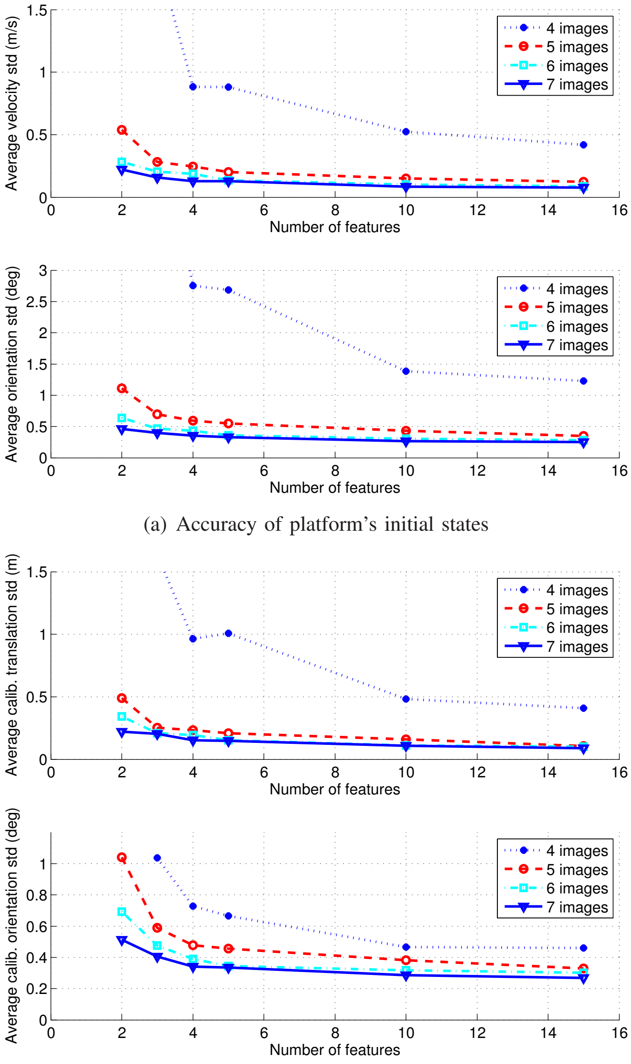
corrupted by independent zero-mean Gaussian noise processes, with standard deviation 0.005 m/s^2 and 0.005 rad/s, respectively. The image-noise standard deviation is set to 1 pixel, in a camera with a field of view of 60° and focal length equal to 500 pixels. IMU measurements are available at 100 Hz, while images are recorded at 1 Hz. The IMU-camera rotation matrix and translation vector are randomly generated and fixed during each trial. Finally, a trajectory is generated using a random initial orientation and velocity, as well as randomly generated acceleration and rotational velocities at each time-step. Similarly, the point features are randomly placed in the scene, so that they can be seen by all the camera poses.

We first compare the accuracy attained using the gravity-constrained least-squares (LS) solution described in Section V, against that of the MLE in Section VI. Specifically, we consider the case where four features are seen in eight images, and compute the errors of the two methods. Due to the number of features, we use the method described in Section IV-B to determine the IMU-camera rotation matrix with noisy measurements. Fig. 1 shows the norm of the errors in the orientation, velocity, and feature position, and Fig. 2 shows the norm of the errors in the calibration parameters in 100 Monte-Carlo trials using different random trajectories. The average errors for the LS and MLE methods are $\{1.1252, 0.4414\}$ degrees for orientation; $\{0.9772, 0.2412\}$ m/s for velocity; and $\{9.8463, 2.1522\}$ m for the feature positions, respectively. For the camera-to-IMU extrinsic calibration parameters, the average errors are $\{0.9560, 0.2430\}$ degrees for the relative orientation, and $\{0.8382, 0.2956\}$ m for the relative position. These results show that the MLE leads to significantly improved accuracy, as expected, due to the fact that it employs a probabilistic modelling of the measurement noise. However, the success of the MLE relies on having a good initial guess, which is provided from the direct methods of Sections IV-B and V. We tried initializing the MLE with randomly generated (but reasonable) starting points for the estimated parameters, and have observed divergence in the majority of trials. This demonstrates the practical utility of using the direct solutions.

We next examine the effect of varying the number of images, N, and features, M, on the estimation accuracy. For each selection of (N, M), we run 100 Monte Carlo trials with randomly generated trajectories, feature positions and extrinsic calibration, and plot the average standard deviation reported by the MLE. Fig. 3(a) shows the results for the IMU orientation and velocity, while Fig. 3(b) plots the results for the IMU-camera orientation and translation. As expected, the standard deviation of the errors monotonically decreases as more images or more features become available. The improvement follows a law of diminishing return: for instance, while using more than 10 features seems to offer little benefit, increasing the number of features from two to three results in a substantial accuracy improvement. Similarly, increasing the number of images from the minimum of four significantly improves the estimates' accuracy. In fact, these results show that with the particular noise settings of these tests, using the minimum four images with a small number of features may lead to unacceptably large estimation errors. On the other hand, with more than four images, good accuracy can be obtained.

VIII. Conclusion

In this paper, we present methods for initializing an estimator in vision-aided inertial navigation applications, without any prior knowledge about the system's initial state. We directly use the camera and inertial measurements to compute the system's observable quantities, namely the platform attitude and velocity, feature positions, and IMU-camera calibration. A key contribution of this work is a convex-optimization based algorithm for computing the rotation matrix between the camera and IMU, which is operational even with a small number of features ($M \geq 2$). After the rotation matrix has been computed, all other observable quantities can be determined by solving a quadratically-constrained least-squares problem. Finally, these estimates are used as the starting point of an iterative maximum-



(a) Accuracy of platform's initial states

(b) Accuracy of extrinsic calibration.

Fig. 3. Accuracy as a function of the number of features and images.

likelihood estimator to obtain more accurate, statistically optimal estimates. Through Monte-Carlo tests, the proposed algorithms are shown to be suitable for use in applications where state estimation has to be performed without any prior information.

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.

References

[1] 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.

[2] 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.

[3] 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.

[4] M. Bryson and S. Sukkarieh, "Observability analysis and active control for airborne SLAM," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 44, no. 1, pp. 261 – 280, 2008.

[5] T. Lupton and S. Sukkarieh, "Efficient integration of inertial observations into visual SLAM without initialization," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, St. Louis, MO, Oct. 2009, pp. 1547–1552.

[6] M. Li and A. I. Mourikis, "Improving the accuracy of EKF-based visual-inertial odometry," in *Proceedings of the IEEE International Conference on Robotics and Automation*, St. Paul, MN, May 2012, pp. 828–835.

[7] ——, "Optimization-based estimator design for vision-aided inertial navigation," in *Proceedings of Robotics: Science and Systems*, Sydney, Australia, July 2012.

[8] A. Martinelli, "Closed-form solution for attitude and speed determination by fusing monocular vision and inertial sensor measurements," in *Proceedings of the IEEE International Conference on Robotics and Automation*, Shanghai, China, May 2011, pp. 4538 – 4545.

[9] ——, "Vision and IMU data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination," *IEEE Transactions on Robotics*, vol. 28, no. 1, pp. 44 –60, Feb. 2012.

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

[11] J. Lobo and J. Dias, "Relative pose calibration between visual and inertial sensors," *International Journal of Robotics Research*, vol. 26, no. 6, pp. 561–575, June 2007.

[12] J. Hol, T. Schon, and F. Gustafsson, "Modeling and calibration of inertial and vision sensors," *The International Journal of Robotics Research*, vol. 29, no. 2-3, pp. 231–244, Feb. 2010.

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

[14] J. Kelly and G. S. Sukhatme, "Fast relative pose calibration for visual and inertial sensors," in *The Eleventh International Symposium on Experimental Robotics*, Berlin, Germany, Apr. 2009, vol. 54, pp. 515–524.

[15] J. Shi and C. Tomasi, "Good features to track," in *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Seattle, WA, June 1994, pp. 593–600.

[16] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 260, no. 2, pp. 91–110, Nov. 2004.

[17] A. B. Chatfield, *Fundamentals of High Accuracy Inertial Navigation*. American Institute of Aeronautics and Astronautics, Inc., 1997.

[18] J. A. Farrell, *Aided Navigation: GPS and High Rate Sensors*. McGraw-Hill, 2008.

[19] T.-C. Dong-Si and A. I. Mourikis, "Supplemental material to IROS 2012 submission," Dept. of Electrical Engineering, University of California, Riverside, Tech. Rep., 2012, www.ee.ucr.edu/∼mourikis/tech_reports/IROS2012.pdf.

[20] H. Stewenius, C. Engels, and D. Nister, "Recent developments on direct relative orientation," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 60, pp. 284–294, June 2006.

[21] D. Nister and F. Schaffalitzky, "What do four points in two calibrated images tell us about the epipoles?" in *Proceedings of the European Conference on Computer Vision*, Prague, Chech Republic, Jun. 2004, pp. 41–57.

[22] J. Chou and M. Kamel, "Quaternions approach to solve the kinematic equation of rotation AaAx=AxAb of a sensor mounted robotic manipulator."

[23] ——, "Finding the position and orientation of a sensor on a robot manipulator using quaternions," *International Journal of Robotics Research*, vol. 10, no. 3, pp. 240–254, 1991.

[24] 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.

[25] N. Courtois, A. Klimov, J. Patarin, and A. Shamir, "Efficient algorithms for solving overdefined systems of multivariate polynomial equations," in *Advances in Cryptology - EUROCRYPT 2000*. Springer, 2000, pp. 392–407.

[26] J. Dattorro, *Convex optimization & Euclidean distance geometry*. Meboo Publishing USA, 2005.

[27] B. Triggs, P. McLauchlan, R. Hartley, and Fitzgibbon, "Bundle adjustment – a modern synthesis," in *Vision Algorithms: Theory and Practice*. Springer Verlag, 2000, pp. 298–375.