

A Kalman Filter-based Algorithm for IMU-Camera Calibration

Faraz M. Mirzaei and Stergios I. Roumeliotis

Abstract—Vision-aided Inertial Navigation Systems (V-INS) can provide precise state estimates for the 3D motion of a vehicle when no external references (e.g., GPS) are available. This is achieved by combining inertial measurements from an IMU with visual observations from a camera under the assumption that the rigid transformation between the two sensors is known. Errors in the IMU-camera calibration process causes biases that reduce the accuracy of the estimation process and can even lead to divergence. In this paper, we present a Kalman filter-based algorithm for precisely determining the unknown transformation between a camera and an IMU. Contrary to previous approaches, we explicitly account for the time correlations of the IMU measurements and provide a figure of merit (covariance) for the estimated transformation. The proposed method does not require any special hardware (such as spin table or 3D laser scanner) except a calibration target. Simulation and experimental results are presented that validate the proposed method and quantify its accuracy.

I. INTRODUCTION

In recent years, Inertial Navigation Systems (INS) have been widely used for estimating the motion of vehicles moving in a 3-dimensional space such as airplanes, helicopters, automobiles, etc [1]. At the core of most INS lies an Inertial Measurement Unit (IMU) that measures linear accelerations and rotational velocities. By integrating these signals in real time, an INS is capable of tracking the position, velocity, and attitude of a vehicle. This deadreckoning process, however, cannot be used over extended periods of time because the errors in the computed estimates continuously increase. This is due to the noise and biases present in the inertial measurements. For this reason, current INS rely on the Global Positioning System (GPS) in order to provide periodic corrections. In most cases, a Kalman filter estimator is used for optimally combining the IMU and GPS measurements [2]. One of the main limitations of the *GPS-aided INS* configuration is that it cannot be used when the GPS signals are not available (e.g., indoors, underground, underwater, in space, etc), or their reliability is limited (e.g., in the vicinity of tall buildings and structures due to specular reflections and multi-path error). Furthermore, high precision GPS receivers are prohibitively expensive, and often the acquired level of accuracy is not sufficient for certain applications (e.g., parallel parking a car within a tight space).

An alternative approach to provide correction to an INS is via the use of visual sensors such as cameras. Cameras are small-size, light-weight, passive sensors that provide rich information for the surroundings of a vehicle at low

This work was supported by the University of Minnesota (DTC), the NASA Mars Technology Program (MTP-1263201), and the National Science Foundation (EIA-0324864, IIS-0643680). The authors would like to thank Anastasios Mourikis for his invaluable help with the experiments.

The authors are with the Dept. of Computer Science & Engineering, University of Minnesota, Minneapolis, MN 55455. Emails: {faraz|stergios}@cs.umn.edu

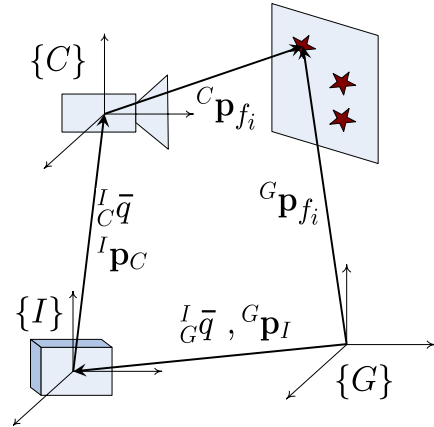


Fig. 1. The geometric relation between the known landmarks f_i and the camera, $\{C\}$, IMU, $\{I\}$, and global, $\{G\}$, frames of reference. The unknown IMU-Camera transformation is denoted by the position and quaternion pair $({}^I\mathbf{p}_C, {}^I_C\bar{q})$. This transformation is determined using estimates of the IMU motion, $({}^G\mathbf{p}_I, {}^I_G\bar{q})$, the projections of the landmarks' positions, ${}^C\mathbf{p}_{f_i}$, on the camera frame (image observations), and the known positions of the landmarks, ${}^G\mathbf{p}_{f_i}$, expressed in the global frame of reference.

cost. When observing a known scene, both the position and attitude of the camera can be computed [3]. Furthermore, by tracking visual features through sequences of images, the motion of the camera can be estimated [4]. Cameras and IMUs are complementary in terms of accuracy and frequency response. An IMU is ideal for tracking the state of a vehicle over short periods of time when it undergoes motions with high dynamic profile. On the other hand, a camera is best suited for state estimation over longer periods of time and for smoother motion profiles. Combining these two sensors to form a *Vision-aided INS* (V-INS) has recently become a popular topic of research [5].

In order to fuse measurements from an IMU and a camera in a V-INS, the 6 degrees of freedom (6-d.o.f.) *transformation* between these two devices needs to be known precisely (cf. Fig. 1). Inaccuracies in the values of the IMU-camera relative pose (position and attitude), will appear as biases that will reduce the accuracy of the estimation process or even cause the estimator to diverge. In most cases in practice, this unknown transformation is computed manually (e.g., from CAD plots) or through the use of additional sensors. For example, for the Mars Exploration Rover (MER) mission [6], a high precision 3D laser scanner was employed to measure the location of the 4 corners of the IMU housing with respect to a checker-board placed in front of the camera for calibration purposes. Although this method achieved sub-degree relative attitude accuracy and less than 1 cm relative position error [7], it is prohibitive for many applications due to the high cost of the equipment (3D laser scanner) involved. Additionally, every time one of the two sensors is removed (e.g., for service) and repositioned, the same

process needs to be repeated, which requires significant time and effort. Automating this procedure will reduce the cost of deploying a V-INS, increase the accuracy of the computed state estimates during regular operation, and minimize the probability of failure due to bias-induced divergence.

In this paper, we present an Extended Kalman Filter (EKF)-based algorithm for determining the 6 d.o.f. transformation between a single camera and an IMU using measurements only from these two sensors. Contrary to existing approaches [8], [9] that rely on modified hand-eye calibration processes (e.g., [10], [11], [12]), our method takes into account the *time correlations* of the IMU measurements by explicitly modeling them. Additionally, our algorithm computes the uncertainties in the estimated quantities, or equivalently, the *covariance*. Furthermore, we do *not* separate the task of translation estimation from rotation estimation which prevents potential error propagation. Finally, unlike existing approaches, the described method does *not* require any special test bed (except a calibration pattern which is also needed for calibrating the intrinsic parameters of the camera). Therefore it offers the inherent capability of re-calibrating the IMU-camera system as frequently as needed.

The IMU-camera calibration is achieved through a two-step process. First, camera images are processed in a *batch* algorithm to compute an *initial estimate* for the camera pose. Additionally, the approximate value of the unknown transformation (e.g., hand-measured or from CAD plots) is combined with the camera-pose estimate to compute an *initial estimate* for the IMU pose (Section III-A). In the next step, both these estimates are used to initialize the corresponding variables in the EKF estimator. By *sequentially* processing additional measurements from the camera and the IMU, the EKF is able to *refine* the initial estimate for the unknown transformation, while simultaneously tracking the position, velocity, and attitude of the two sensors (Section III-B). The developed algorithm is tested both in simulation and experimentally (Section IV) to demonstrate the precision of our calibration process.

II. RELATED WORK

A well-known related process is the hand-eye calibration [10], whose objective is to estimate the 6 d.o.f. transformation between a camera and a robot manipulator. Recently, there have been some attempts to modify existing hand-eye calibration algorithms to determine the IMU-camera alignment [8], [9]. Specifically, in [9] the rotation part of the hand-eye equation is solved by nonlinear optimization software under the assumption that the translation between the IMU and the camera is negligible. However, in most realistic situations this assumption is *not* valid and ignoring the translation introduces biases in estimation algorithms using these alignment parameters.

A different approach to this problem is proposed by Lobo and Dias in [8]. First, they obtain the vertical direction of the IMU and the camera frames by measuring the direction of gravity while viewing a vertically-installed calibration pattern. Then, using Horn’s method [13], they estimate the rotation between the IMU and the camera. Finally, they use

a *spin table* to rotate the system about the IMU’s center and zero-out the linear velocity of the IMU due to rotation. This simplification allows one to compute the translation between the camera and the IMU based only on the camera translation measurements. The main drawback of this approach is that it ignores the *time correlation* between the inertial measurements due to the IMU biases. Additionally, it does *not* provide any figure of merit of the achieved level of accuracy (e.g., covariance of the estimated quantities). Furthermore this two-stage approach that *decouples* the computation of rotation and translation allows error propagation from the rotation estimates to the translation estimates. Finally this method requires fine adjustment of the IMU-camera system on a *spin table* which restricts its applicability when re-calibration is frequently required.

The IMU-camera and hand-eye calibration problems require separate treatments due to the different *noise characteristics* of the IMU and shaft-encoders signals. Specifically, while the shaft-encoder measurements at different time instants are uncorrelated, consecutive IMU measurements are not. This is due to the IMU *biases*. Ignoring the time correlation of the inertial measurements limits the accuracy of the IMU-camera calibration process and can lead to inconsistent estimates.

To the best of our knowledge the presented EKF-based algorithm is the first approach to the IMU-camera calibration problem that does *not* ignore the correlations between the IMU measurements. Furthermore, the uncertainty in the estimated alignment parameters is provided at every time step by computing their covariance. Finally, it is shown that it suffices to rotate the camera in place in order for these parameters to become observable.

III. DESCRIPTION OF THE ALGORITHM

The IMU-camera calibration algorithm consists of two steps. During the first step, we process camera images to determine the IMU pose estimates required for initializing the EKF state vector. Then, we operate the Kalman filter over a sequence of poses where images to known features are recorded, in order to refine the estimate for the unknown IMU-camera transformation.

A. Filter Initialization

The purpose of this process is to determine the initial estimate for the *IMU pose* (${}^G\mathbf{p}_I, {}^I\bar{q}$) where ${}^G\mathbf{p}_I$ denotes the position of the IMU with respect to the global frame of reference, and ${}^I\bar{q}$ is the rotation quaternion between the IMU and the global frames.

We first compute an estimate for the *camera pose* (${}^G\mathbf{p}_C, {}^C\bar{q}$) using visual features (corners of the squares in the calibration pattern) whose positions ${}^G\mathbf{p}_{f_i}$ are known in global coordinates. Specifically, the initial estimates of the depth to these features are computed using Ansar’s method [3], while the initial estimates for the camera pose is determined by employing Horn’s method [13]. Finally, a Levenberg-Marquart least-squares algorithm refines the camera-pose estimate and additionally computes its covariance.

In the next step of the initialization process, we use an approximate estimate for the unknown IMU-camera transformation (${}^I\mathbf{p}_C, {}^I\bar{q}$). This was determined manually in our

case but it can also be found using the CAD plots showing the IMU-camera placement. We should note, that the requirement for an approximate estimate for the IMU-camera transformation is not limiting, since it can also be determined by employing any hand-eye calibration algorithm. An initial estimate for the IMU pose is then computed from the following relations (cf. Fig. 1):

$$\begin{aligned} {}^G\mathbf{p}_I &= {}^G\mathbf{p}_C - \mathbf{C}^T({}^C\bar{q})\mathbf{C}^T({}^I\bar{q}){}^I\mathbf{p}_C \\ {}^I\bar{q} &= {}^I\bar{c} \otimes {}^C\bar{q} \end{aligned} \quad (1)$$

where $\mathbf{C}(\bar{q})$ is the rotational matrix corresponding to quaternion \bar{q} , and \otimes denotes quaternion multiplication. Finally, after computing the corresponding Jacobians (by linearizing Eq. (1)) and considering the uncertainty (covariance) in the estimates of $({}^I\mathbf{p}_C, {}^I\bar{q})$ and $({}^G\mathbf{p}_C, {}^C\bar{q})$, the covariance of the initial IMU pose estimate is readily found.

B. Filter Propagation

The EKF estimates the IMU pose and linear velocity as well as the unknown transformation (rotation and translation) between the camera and the IMU. Additionally, the filter estimates the biases in the IMU signals.

1) *Continuous-time system modeling*: We first derive the linearized continuous-time system model that describes the time evolution of the errors in the state estimates. Discretization of this model allows us to employ the sampled measurements of the IMU for state propagation. The filter state is described by the vector:

$$\mathbf{x} = [{}^I\bar{q}^T \quad \mathbf{b}_g^T \quad {}^G\mathbf{v}_I^T \quad \mathbf{b}_a^T \quad {}^G\mathbf{p}_I^T \quad {}^I\bar{c}^T \quad {}^I\mathbf{p}_C^T]^T \quad (2)$$

where ${}^I\bar{q}(t)$ and ${}^C\bar{q}(t)$ are the quaternions which represent the orientation of the *global frame* and the *camera frame* in the IMU frame, respectively. The position and velocity of the IMU in the global frame are denoted by ${}^G\mathbf{p}_I(t)$ and ${}^G\mathbf{v}_I(t)$. ${}^I\mathbf{p}_C(t)$ is the position of the camera in the IMU frame, and $\mathbf{b}_g, \mathbf{b}_a$ are the 3×1 bias vectors affecting the gyroscope and accelerometer measurements. These biases are typically present in the signals of inertial sensors, and need to be modeled and estimated, in order to attain accurate state estimates. In our work the IMU biases are modeled as random walk processes driven by white Gaussian noise vectors \mathbf{n}_{wg} and \mathbf{n}_{wa} , respectively.

The system model describing the time evolution of the IMU state is given by the following equations [14], [15]:

$${}^I\dot{\bar{q}}(t) = \frac{1}{2}\Omega(\omega(t)){}^I\bar{q}(t) \quad (3)$$

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

$$\dot{\mathbf{b}}_g(t) = \mathbf{n}_{wg}(t), \quad \dot{\mathbf{b}}_a(t) = \mathbf{n}_{wa}(t) \quad (5)$$

$${}^I\dot{\bar{c}}(t) = \mathbf{0}_{3 \times 1}, \quad {}^I\dot{\mathbf{p}}_C(t) = \mathbf{0}_{3 \times 1} \quad (6)$$

In these expressions $\omega(t) = [\omega_x \quad \omega_y \quad \omega_z]^T$ is the rotational velocity of the IMU, expressed in the IMU frame, and

$$\Omega(\omega) = \begin{bmatrix} -[\omega \times] & \omega \\ \omega^T & 0 \end{bmatrix}, [\omega \times] = \begin{bmatrix} 0 & -\omega_z & \omega_y \\ \omega_z & 0 & -\omega_x \\ -\omega_y & \omega_x & 0 \end{bmatrix}$$

Finally, ${}^G\mathbf{a}$ is the acceleration of the IMU, expressed in the global frame.

The gyroscope and accelerometer measurements, ω_m , and \mathbf{a}_m respectively, which are employed for state propagation,

are modeled as

$$\omega_m = \omega + \mathbf{b}_g + \mathbf{n}_g \quad (7)$$

$$\mathbf{a}_m = \mathbf{C}({}^I\bar{q})({}^G\mathbf{a} - {}^G\mathbf{g}) + \mathbf{b}_a + \mathbf{n}_a \quad (8)$$

where \mathbf{n}_g and \mathbf{n}_a are zero-mean, white Gaussian noise processes, and ${}^G\mathbf{g}$ is the gravitational acceleration.

By applying the expectation operator on both sides of Eq. (3)-(6), we obtain the state estimates' propagation equations:

$${}^I\hat{\bar{q}}(t) = \frac{1}{2}\Omega(\hat{\omega}(t)){}^I\hat{\bar{q}}(t) \quad (9)$$

$${}^G\hat{\mathbf{p}}_I(t) = {}^G\hat{\mathbf{v}}_I(t), \quad {}^G\hat{\mathbf{v}}_I(t) = \mathbf{C}^T({}^I\hat{\bar{q}}(t))\hat{\mathbf{a}}(t) + {}^G\mathbf{g} \quad (10)$$

$$\hat{\mathbf{b}}_g(t) = \mathbf{0}_{3 \times 1}, \quad \hat{\mathbf{b}}_a(t) = \mathbf{0}_{3 \times 1} \quad (11)$$

$${}^I\hat{\bar{c}}(t) = \mathbf{0}_{3 \times 1}, \quad {}^I\hat{\mathbf{p}}_C(t) = \mathbf{0}_{3 \times 1} \quad (12)$$

with

$$\hat{\mathbf{a}}(t) = \mathbf{a}_m(t) - \hat{\mathbf{b}}_a(t), \quad \text{and} \quad \hat{\omega}(t) = \omega_m(t) - \hat{\mathbf{b}}_g(t)$$

The 21×1 filter error-state vector is defined as:

$$\tilde{\mathbf{x}}_k = \left[{}^I\delta\theta^T \quad \tilde{\mathbf{b}}_g^T \quad {}^G\tilde{\mathbf{v}}_I^T \quad \tilde{\mathbf{b}}_a^T \quad {}^G\tilde{\mathbf{p}}_I^T \quad {}^I\delta\theta^T \quad {}^I\tilde{\mathbf{p}}_C^T \right]^T \quad (13)$$

For the IMU and camera positions, and the IMU velocity and biases, the standard additive error definition is used (i.e., the error in the estimate \hat{x} of a quantity x is $\tilde{x} = x - \hat{x}$). However, for the quaternions a different error definition is employed. In particular, if \hat{q} is the estimated value of the quaternion q , then the orientation error is described by the *error quaternion*:

$$\delta\bar{q} = \bar{q} \otimes \hat{q}^{-1} \simeq \left[\frac{1}{2}\delta\theta^T \quad 1 \right]^T \quad (14)$$

Intuitively, the quaternion $\delta\bar{q}$ describes the (small) rotation that causes the true and estimated attitude to coincide. The main advantage of this error definition is that it allows us to represent the attitude uncertainty by the 3×3 covariance matrix $E\{\delta\theta\delta\theta^T\}$. Since the attitude corresponds to 3 degrees of freedom, this is a minimal representation.

The linearized continuous-time error-state equation is:

$$\dot{\tilde{\mathbf{x}}} = \mathbf{F}_c\tilde{\mathbf{x}} + \mathbf{G}_c\mathbf{n}, \quad (15)$$

where

$$\mathbf{F}_c = \begin{bmatrix} -[\hat{\omega} \times] & -\mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 9} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 9} \\ -\mathbf{C}^T({}^I\hat{\bar{q}})[\hat{\mathbf{a}} \times] & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -\mathbf{C}^T({}^I\hat{\bar{q}}) & \mathbf{0}_{3 \times 9} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 9} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 9} \\ \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 9} \end{bmatrix}$$

$$\mathbf{G}_c = \begin{bmatrix} -\mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & -\mathbf{C}^T({}^I\hat{\bar{q}}) & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} & \mathbf{I}_3 \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_3 & \mathbf{0}_{3 \times 3} & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 3} & \mathbf{0}_{6 \times 3} \end{bmatrix}, \quad \mathbf{n} = \begin{bmatrix} \mathbf{n}_g \\ \mathbf{n}_{wg} \\ \mathbf{n}_a \\ \mathbf{n}_{wa} \end{bmatrix}$$

and \mathbf{I}_3 is the 3×3 identity matrix. The covariance, \mathbf{Q}_c , of the system noise depends on the IMU noise characteristics and is computed off-line.

2) *Discrete-time implementation*: The IMU signals ω_m and \mathbf{a}_m are sampled at 100 Hz (i.e., $T=0.01$ sec.). Every time a new IMU measurement is received, the state estimate is propagated using using 4th order Runge-Kutta numerical

integration of Eqs. (9)-(12). In order to derive the covariance propagation equation, we compute the discrete-time state transition matrix:

$$\Phi_k = \Phi(t_k + T, t_k) = \exp\left(\int_{t_k}^{t_k+T} \mathbf{F}_c(\tau) d\tau\right) \quad (16)$$

and the discrete-time system noise covariance matrix:

$$\mathbf{Q}_d = \int_{t_k}^{t_k+T} \Phi(t_{k+1}, \tau) \mathbf{G}_c \mathbf{Q}_c \mathbf{G}_c^T \Phi^T(t_{k+1}, \tau) d\tau \quad (17)$$

The propagated covariance is then computed as:

$$\mathbf{P}_{k+1|k} = \Phi_k \mathbf{P}_{k|k} \Phi_k^T + \mathbf{Q}_d$$

C. Measurement Model

The IMU-camera moves continuously and records images of a calibration pattern. These are then processed to detect and identify point features whose positions ${}^G \mathbf{p}_{f_i}$ are known with respect to the global frame of reference (centered and aligned with the checker-board pattern of the calibration target). A least-squares corner extractor is used for feature detection and data association. Once this process is completed for each image, a list of point features along with their measured image coordinates (u_i, v_i) is provided to the EKF, which uses them to update the state estimates.

The projective camera measurement model employed is:

$$\mathbf{z}_i = \mathbf{h}_i(\mathbf{x}, {}^G \mathbf{p}_{f_i}) + \eta_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} + \eta_i = \begin{bmatrix} x_i/z_i \\ y_i/z_i \end{bmatrix} + \eta_i \quad (18)$$

where (cf. Fig. 1),

$${}^C \mathbf{p}_{f_i} = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \mathbf{C}({}_I^C \hat{q}) \mathbf{C}({}_G^I \hat{q}) ({}^G \mathbf{p}_{f_i} - {}^G \mathbf{p}_I) - \mathbf{C}({}_I^C \hat{q}) {}^I \mathbf{p}_C \quad (19)$$

and η_i is the feature-measurement noise with covariance $\mathbf{R}_i = \sigma_i^2 \mathbf{I}_2$.

The measurement matrix \mathbf{H}_i is:

$$\mathbf{H}_i = \mathbf{J}_{cam}^i [\mathbf{J}_{\theta_G}^i \quad \mathbf{0}_{3 \times 3} \quad \mathbf{0}_{3 \times 3} \quad \mathbf{0}_{3 \times 3} \quad \mathbf{J}_{\mathbf{p}_I}^i \quad \mathbf{J}_{\theta_C}^i \quad \mathbf{J}_{\mathbf{p}_C}^i] \quad (20)$$

with

$$\begin{aligned} \mathbf{J}_{cam}^i &= \frac{1}{z_i^2} \begin{bmatrix} \hat{z}_i & 0 & -\hat{x}_i \\ 0 & \hat{z}_i & -\hat{y}_i \end{bmatrix} \\ \mathbf{J}_{\theta_G}^i &= \mathbf{C}({}_I^C \hat{q}) [\mathbf{C}({}_G^I \hat{q}) ({}^G \mathbf{p}_{f_i} - {}^G \hat{\mathbf{p}}_I) \times] \\ \mathbf{J}_{\theta_C}^i &= -\mathbf{C}({}_I^C \hat{q}) [\mathbf{C}({}_G^I \hat{q}) ({}^G \mathbf{p}_{f_i} - {}^G \hat{\mathbf{p}}_I) - {}^I \hat{\mathbf{p}}_C \times] \\ \mathbf{J}_{\mathbf{p}_I}^i &= -\mathbf{C}({}_I^C \hat{q}) \mathbf{C}({}_G^I \hat{q}), \quad \mathbf{J}_{\mathbf{p}_C}^i = -\mathbf{C}({}_I^C \hat{q}) \end{aligned} \quad (21)$$

When observations to N features are available concurrently, we stack these in one measurement vector $\mathbf{z} = [\mathbf{z}_1^T \cdots \mathbf{z}_N^T]^T$ to form a single batch-form update equation. Similarly, the batch measurement Jacobian matrix is defined as $\mathbf{H} = [\mathbf{H}_1^T \cdots \mathbf{H}_N^T]^T$. Finally, the measurement residual is computed as:

$$\mathbf{r} = \mathbf{z} - \hat{\mathbf{z}} \Rightarrow \tilde{\mathbf{z}} \simeq \mathbf{H} \tilde{\mathbf{x}} + \eta, \quad (22)$$

where $\eta = [\eta_1^T \cdots \eta_N^T]^T$ is the measurement noise with the covariance $\mathbf{R} = \text{Diag}(\mathbf{R}_i)$.

D. Iterated Extended Kalman Filter Update

In order to increase the numerical stability in the face of the highly nonlinear measurement model, we employ the *Iterated Extended Kalman Filter* [16] to update the state. The iterative scheme proceeds as follows:

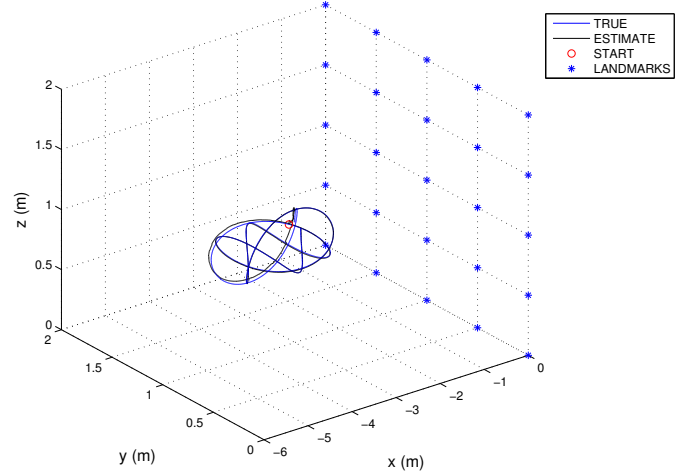


Fig. 2. The trajectory of the IMU-camera system for 15 sec.

At each iteration step j

- 1) Compute $\hat{\mathbf{z}}^j = E\{\mathbf{z}\}$ as a function of the current j -th iterate $\hat{\mathbf{x}}_{k+1|k+1}^j$ using the measurement function (18).
- 2) Linearize the measurement model around the current iterate to obtain \mathbf{H}^j by employing (20).
- 3) Form the residual $\mathbf{r}^j = \mathbf{z} - \hat{\mathbf{z}}^j$, and compute its covariance $\mathbf{S}^j = \mathbf{H}^j \mathbf{P}_{k+1|k} \mathbf{H}^{jT} + \mathbf{R}$
- 4) Using the Kalman gain $\mathbf{K}^j = \mathbf{P}_{k+1|k} \mathbf{H}^{jT} (\mathbf{S}^j)^{-1}$ compute the correction

$$\Delta \mathbf{x}^j = \mathbf{K}^j (\mathbf{r}^j + \mathbf{H}^j \Delta \mathbf{x}^{j-1}) \quad (23)$$

necessary for determining the next iterate of the updated state estimate $\hat{\mathbf{x}}_{k+1|k+1}^{j+1}$

This process is iterated to convergence. The iteration is started using the propagated state estimate $\hat{\mathbf{x}}_{k+1|k+1}^0$ as zeroth iterate, making the first iteration equivalent to a regular EKF update. Finally the covariance matrix for the current state is updated, using the values for \mathbf{K} and \mathbf{S} from the final iteration:

$$\mathbf{P}_{k+1|k+1} = \mathbf{P}_{k+1|k} - \mathbf{K} \mathbf{S} \mathbf{K}^T \quad (24)$$

E. Outlier Rejection

Before using the detected features in the measurement update, we employ a Mahalanobis-distance test to detect and reject mismatches or very noisy observations. Every time a new measurement becomes available, the following expression is employed to compute the Mahalanobis distance:

$$\chi^2 = (\mathbf{z}_{i_{k+1}} - \hat{\mathbf{z}}_{i_{k+1}})^T \mathbf{S}^{-1} (\mathbf{z}_{i_{k+1}} - \hat{\mathbf{z}}_{i_{k+1}}) \quad (25)$$

A probabilistic threshold on this distance is used to specify whether the measurement is reliable or not. Measurements which pass this test are processed by the iterative update procedure as described above.

IV. SIMULATION AND EXPERIMENTAL RESULTS

A. Simulation Results

In order to validate the proposed algorithm when ground truth is available, we have performed a number of simulation experiments. In our simulation setup, an IMU-camera rig moves in front of a calibration target containing 25 known features. These correspond to the vertices of a rectangular grid with 50×50 cm cell size, which is aligned with the YZ

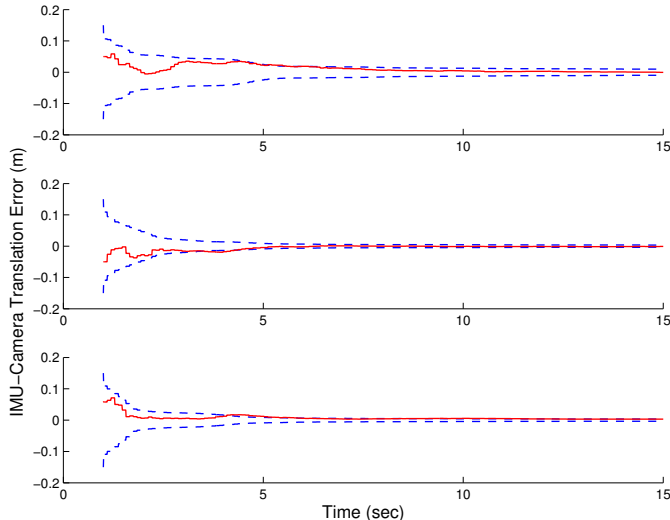


Fig. 3. The state error and 3σ bounds for the IMU-camera translation along axes X, Y and Z. The initial error is approximately [5cm, -5cm, 6cm].

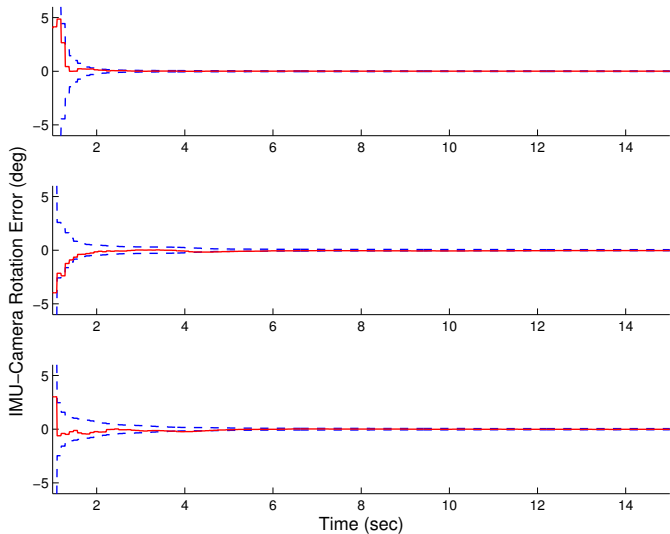


Fig. 4. The state error and 3σ bounds for the IMU-camera rotation about axes X (roll), Y (pitch) and Z (yaw). The initial errors are approximately $[4^\circ, -4^\circ, 3^\circ]$

plane (cf. Fig. 2). The camera is assumed to have 50° field of view. Additionally, the image measurements received at a rate of 10Hz, are distorted with noise of $\sigma=1$ pixel. The IMU noise characteristics are the same as those of the ISIS IMU used in the real world experiments (cf. Section IV-B). The IMU measurements are received at 100Hz.

The initial translation error is set to [5, -5, 6] cm with a standard deviation of 5 cm in each axis. The initial error in rotation is set to $[4^\circ, -4^\circ, 3^\circ]$ with 3° standard deviation of uncertainty in each axis of rotation. Consequently, the filter state vector and error-state covariance matrix are initialized according to the process described in Section III-A.

Following the initialization step, the system performs a spiral motion within 3-5 m off the calibration pattern. The actual and estimated trajectories are shown in Fig. 2. For the duration of this simulation (~ 15 sec), 150 images were processed and, on an average, 21.7 landmarks were visible in each image. The state estimate errors and their 3σ bounds for the transformation between the IMU and the camera in a

TABLE I

FINAL UNCERTAINTY (3σ) OF THE IMU-CAMERA PARAMETERS AFTER 100 SEC FOR TWO MOTION SCENARIOS. XYZ REPRESENT TRANSLATION ALONG X, Y, AND Z AXES. RPY INDICATE ROTATION ABOUT LOCAL X (ROLL), Y (PITCH), AND Z (YAW) AXES.

3σ	x(cm)	y(cm)	z(cm)	roll($^\circ$)	pitch($^\circ$)	yaw($^\circ$)
Initial	15	15	15	9	9	9
XYZ-RPY	0.18	0.14	0.13	0.013	0.013	0.013
RPY	0.25	0.22	0.22	0.0082	0.024	0.024

TABLE II

MONTE CARLO SIMULATIONS: COMPARISON OF THE STANDARD DEVIATIONS OF FINAL IMU-CAMERA TRANSFORMATION ERROR (σ_{err}), AND THE AVERAGE COMPUTED UNCERTAINTY OF THE ESTIMATES (σ_{est}).

	x(cm)	y(cm)	z(cm)	roll($^\circ$)	pitch($^\circ$)	yaw($^\circ$)
σ_{err}	0.29	0.23	0.28	0.019	0.036	0.039
σ_{est}	0.31	0.24	0.28	0.019	0.039	0.040

typical simulation are shown in Figs. 3 and 4. It can be seen that even with the relatively large initial error for the IMU-camera transformation, the algorithm is still able to attain very accurate estimates of the calibration parameters. The final standard deviations of the estimates are [0.32, 0.28, 0.30] cm for the translation and $[0.024^\circ, 0.040^\circ, 0.040^\circ]$ for rotation.

1) *General motion vs. rotation only:* In [17], we have shown that the proposed system is observable when the IMU-camera rig undergoes rotational motion even if no translation occurs. Hereafter we examine the achievable accuracy for motions with and without translation after 100 seconds when the system undergoes (i) spiral motion (i.e., exciting all 6 d.o.f.) and (ii) pure rotation (i.e., exciting only three d.o.f. corresponding to attitude). In all these simulations, the initial uncertainty of the IMU-camera translation and rotation are set to 15 cm and 9° deg (3σ) in each axis respectively. A summary of the results is shown in Table I.

Based on these results, it can be seen that by running the algorithm for longer times, we can estimate the calibration parameters with high accuracy despite the large initial uncertainty. The second row of the Table I, (XYZ-RPY), corresponds to the full motion of the system when all 6 d.o.f. are excited. In this case, after sufficient time, the translation uncertainties are reduced to less than 2 mm in each axes. Additionally, as it can be seen in this particular example, the translation uncertainty along the X axis is slightly more than the uncertainty along the two other axes. This is a typical result observed in all simulations with similar setup. The main reason for this is the limitation on the pitch and yaw rotations (i.e., about the Y and Z axes, respectively) so as to keep the landmarks in the field of view. On the other hand, the roll rotation (about X axis) is virtually unlimited and it can span a complete circle without losing visual contact with the landmarks (the optical axis of the camera is aligned with local X axes).

The third row of Table I corresponds to a scenario where the motion is constrained to pure rotation. As expected the system is still observable and both the translation and the

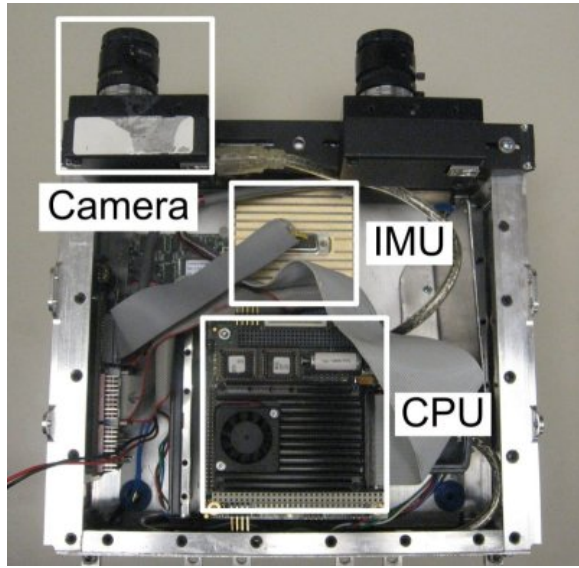


Fig. 5. The testbed used for the experiments.

rotation between the IMU and the camera are accurately estimated [17]. The accuracy of the rotation estimation between the IMU and the camera in both scenarios is shown in the last three columns of Table I. It is observed that in all cases the rotational parameters can be estimated extremely accurately, even when the system has not undergone any translation.

2) *Monte Carlo Simulations:* Finally, we have conducted Monte Carlo simulations to statistically evaluate the accuracy of the filter. We ran 100 simulations with a setup similar to the first simulation described in this section. The initial standard deviation of the IMU-camera transformation is set to 3 cm for translation and 3° for rotation. The initial values in each run are randomly generated according to a Gaussian probability distribution with these standard deviations. Each simulation is run for 15 seconds and the final calibration errors along with their estimated uncertainty are recorded. The ensemble mean of the recorded errors are $[0.058, -0.002, 0.044]$ cm for the translation and $[-0.0038^\circ, 0.0013^\circ, -0.0009^\circ]$ for the rotation. It can be seen that the mean error is at least one order of magnitude smaller than the typical error, demonstrating that the filter is indeed unbiased.

The standard deviations of the recorded errors are shown in the first row of Table II. The second row of this table shows the average of the computed standard deviations at each realization of the experiment. Comparison of these two rows indicates consistency of the filter as the standard deviation of the actual error is smaller than or equal to the computed standard deviations. Thus, the filter is not overconfident.

B. Experimental Results

In order to demonstrate the validity of our algorithm in realistic situations, we have conducted experiments using a testbed which consists of an ISIS IMU, a firewire camera, and a PC104 computer for data acquisition (cf. Fig. 5). The IMU and the camera are rigidly mounted on the chassis and their relative pose does not change during the experiment. The intrinsic parameters of the camera were calibrated prior

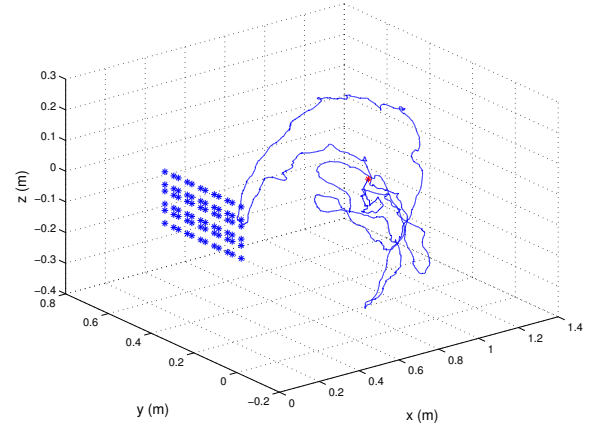


Fig. 6. The estimated trajectory of the IMU motion for 140 sec. The starting point is shown by an asterisk on the trajectory.

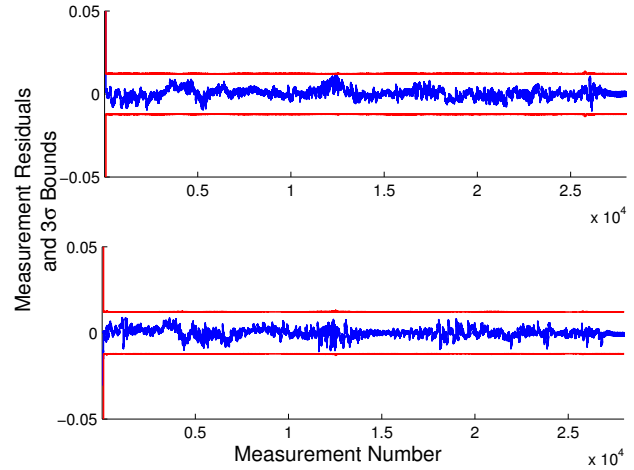


Fig. 9. [Calibrated IMU-Camera] The measurement residuals along with their 3σ bounds for the axes u and v of the images.

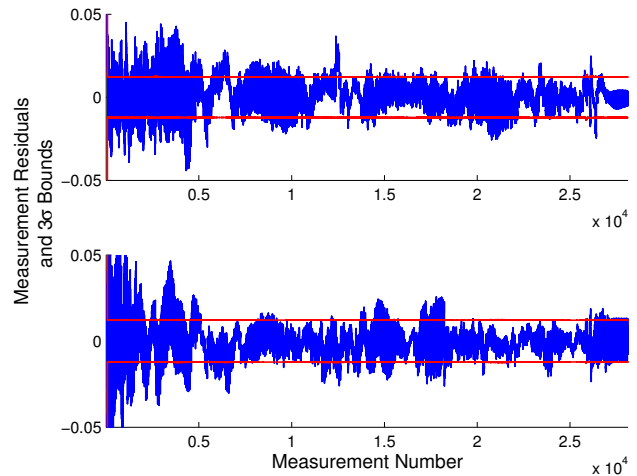


Fig. 10. [Uncalibrated IMU-Camera] The measurement residuals along with their 3σ bounds for the axes u and v of the images.

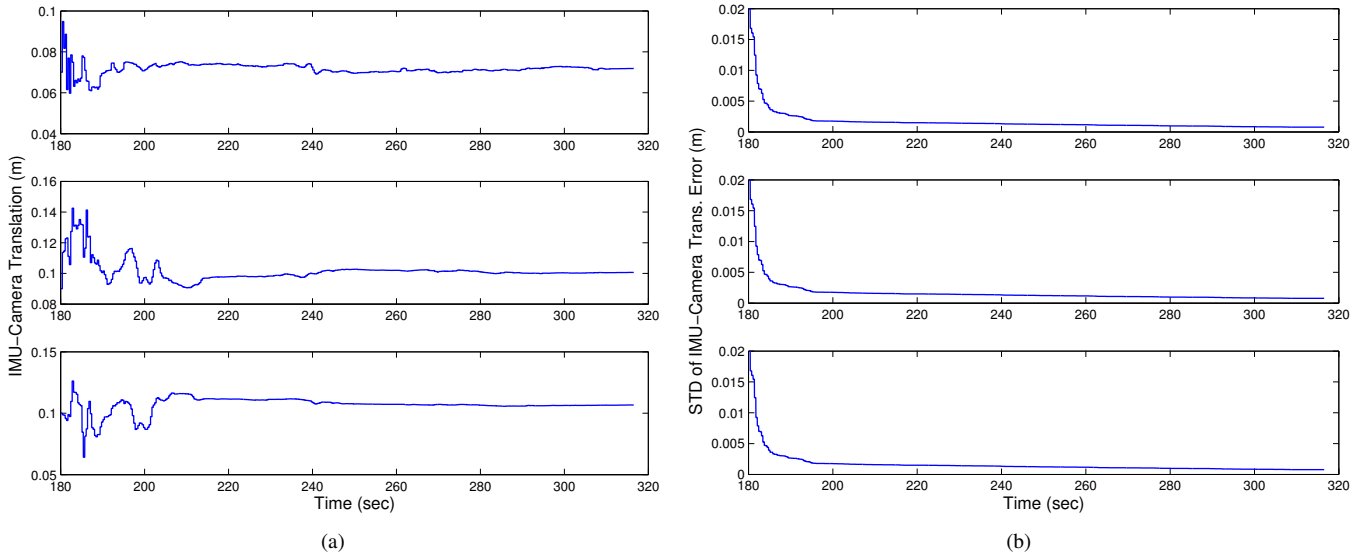


Fig. 7. The time-evolution of X, Y, and Z components of the estimated IMU-camera translation (a), and corresponding standard deviations (b).

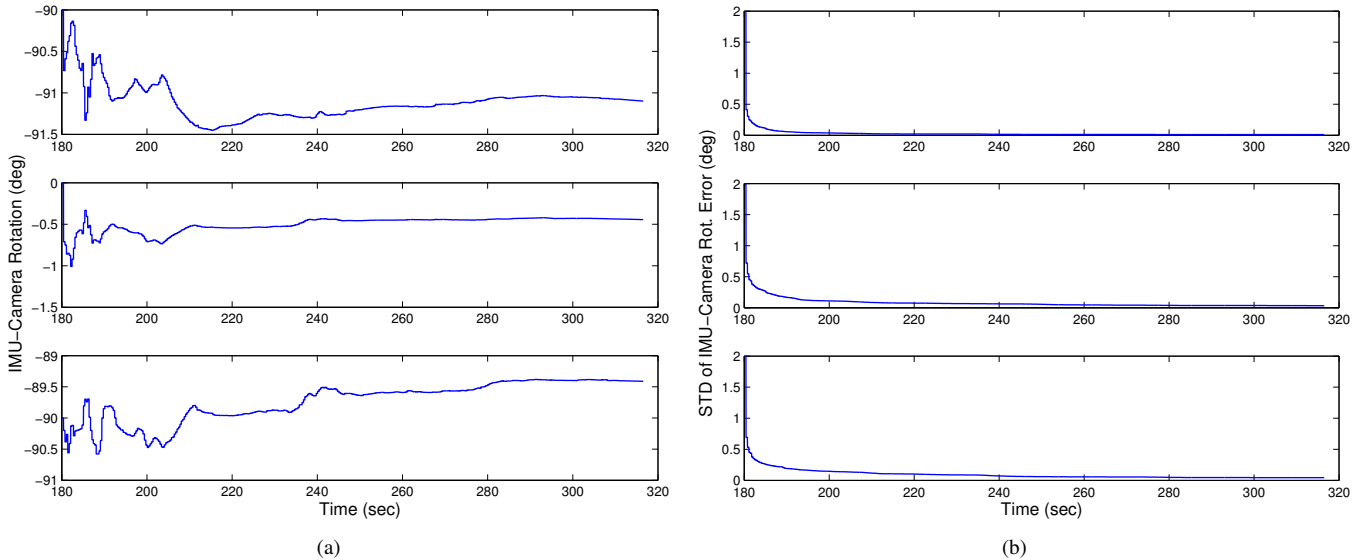


Fig. 8. The time-evolution of the estimated IMU-camera rotation about axes X, Y, and Z (a), and corresponding standard deviations (b).

to the experiment and are assumed constant. The camera's field of view is 30° with a focal length of 6 mm. The resolution of the images is 640×480 pixels. Images are received at a rate of 3Hz while the IMU provides measurements at a frequency of 100Hz. The PC104 stores the images and the IMU measurements for offline processing.

A calibration pattern (checker board) was used to provide 72 globally known landmarks which were placed 2cm-4cm apart from each other. The landmarks were the corners of the squares which were extracted using a least-squares corner detector. We have assumed that the camera measurements are corrupted by additive white Gaussian noise with standard deviation equal to 2 pixels¹.

The hand-measured translation and rotation between the

¹The actual pixel noise is less than 2 pixels. However, in order to compensate for the existence of unmodeled nonlinearities and imperfect camera calibration, we have inflated noise to 2 pixels.

IMU and the camera was used as an initial guess for the unknown transformation. Additionally, the pose of the IMU was initialized as described in Section III-A. Finally, initialization of the gyro and the accelerometer biases was performed by placing the testbed in a static position for approximately 180 sec. During this time the EKF processed IMU and camera measurements while enforcing the static constraint (zero position and attitude displacement). The resulting state vector along with the error-state covariance matrix were then directly used to run the experiment.

Once the initialization process was complete, we started moving the testbed while the camera was facing the calibration pattern. For the duration of the experiment, the distance between the camera and the calibration pattern was 0.5-2 m in order to keep the corners of the checker board visible. Additionally, the testbed was moved in such a way so as to excite all degrees of freedom while at the same time keeping

TABLE III

INITIAL AND FINAL ESTIMATES OF THE IMU-CAMERA PARAMETERS AND THEIR UNCERTAINTY FOR THE DESCRIBED EXPERIMENT.

	$x \pm 3\sigma(\text{cm})$	$y \pm 3\sigma(\text{cm})$	$z \pm 3\sigma(\text{cm})$
Initial	6 ± 6	8 ± 6	10 ± 6
Final	7.19 ± 0.23	10.06 ± 0.29	10.67 ± 0.26
	$\text{roll} \pm 3\sigma(^{\circ})$	$\text{pitch} \pm 3\sigma(^{\circ})$	$\text{yaw} \pm 3\sigma(^{\circ})$
Initial	-90 ± 6	0 ± 6	-90 ± 6
Final	-91.099 ± 0.033	-0.440 ± 0.098	-89.41 ± 0.13

the landmarks within the camera's field of view.

During the motion of the testbed (~ 140 sec), 409 images were recorded. The average number of detected landmarks was 69.8/image and the average number of processed landmarks, after outlier detection, was 68.4/image (i.e., less than 2.2% of the measured landmarks were rejected as outliers). The EKF algorithm was able to estimate the IMU-camera transformation while keeping track of the IMU pose, velocity, and noise biases. The estimated trajectory of the IMU is shown in Fig. 6.

The time-evolution of the estimated calibration parameters is depicted in Figs. 7(a) and 8(a). As evident from these plots, the calibration parameters converge to steady-state values after approximately 260 seconds (including the 180 sec of the duration of the initialization process). The estimated standard deviation of the calibration errors is shown in Figs. 7(b) and 8(b). A summary of the results for this experiment is provided in Table III. It is worth mentioning that the initial standard deviation of 2 cm for the error in the translation parameters improves to less than 0.1 mm for all axes. Additionally the initial standard deviation of 2° for the error in rotation decreases to less than 0.05° for each axis of rotation.

Considering that the exact values of the IMU-camera transformation (ground truth) were not available in this experiment, the main indicator of the validity of the filter was the measurement residuals. These are shown in Fig. 9 and demonstrate the consistency of the employed EKF.

In order to stress the importance of acquiring precise IMU-camera calibration estimates, we have also tested in the same experimental setup, an EKF-based estimator that does *not* estimate the calibration parameters online. Instead this filter uses the initial guess for the unknown IMU-camera transformation to estimate the IMU pose and biases. In this case, and as evident from the resulting measurement residuals (cf. Fig. 10) the approximate values for the calibration parameters lead to large inconsistencies of the estimator.

V. CONCLUSIONS

In this paper, we have presented an EKF-based algorithm for estimating the transformation between an IMU and a camera rigidly attached on a mobile platform. To the best of our knowledge this is the first approach to the IMU-camera calibration problem that appropriately accounts for the *time correlations* between the IMU measurement. Additionally, and contrary to previous work on this subject, we do *not* separate the task of translation estimation from rotation es-

timization, and hence prevent error propagation. Moreover, by treating the problem within the Kalman filtering framework, we are also able to compute the *covariance* of the estimated quantities as an indicator of the achieved level of accuracy. Therefore by accounting for this uncertainty in the consequent navigational algorithm, we are able to explicitly model their impact. Last but not the least, an important feature of this algorithm is the ability to perform the calibration process *without* requiring any specific test bed (such as rotating table [8]) except the calibration pattern which is also needed when calibrating the intrinsic parameters of the camera. The derived estimator was tested both in simulation and experimentally and it was shown to achieve accuracy in the order of millimeters and sub-degrees, respectively, for the translational and rotational components of the IMU-camera transformation.

Currently, we are investigating the possibility to expand this work to the case of IMU-camera calibration without using the calibration pattern, i.e., by considering naturally occurring visual features whose positions are unknown.

REFERENCES

- [1] A. B. Chatfield, *Fundamentals of High Accuracy Inertial Navigation*. Reston, VA: American Institute of Aeronautics and Astronautics, 1997.
- [2] D. Titterton and J. Weston, Eds., *Strapdown Inertial Navigation Technology*. IEE, 2005.
- [3] A. Ansar and K. Daniilidis, "Linear pose estimation from points or lines," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 578–589, May 2003.
- [4] A. Azerbayejani and A. Pentland, "Recursive estimation of motion, structure, and focal length," *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 17, no. 6, pp. 562–575, 1995.
- [5] D. Strelow, "Motion estimation from image and inertial measurements," Ph.D. dissertation, Carnegie Mellon University, 2004.
- [6] A. Johnson, R. Willson, J. Goguen, J. Alexander, and D. Meller, "Field testing of the mars exploration rovers descent image motion estimation system," in *Proc. IEEE International Conference on Robotics and Automation*, Barcelona, Spain, Apr. 2005, pp. 4463–4469.
- [7] A. E. Johnson, J. F. Montgomery, and L. H. Matthies, "Vision guided landing of an autonomous helicopter in hazardous terrain," in *Proc. IEEE International Conference on Robotics and Automation*, Barcelona, Spain, Apr. 2005, pp. 3966–3971.
- [8] J. Lobo and J. Dias, "Relative pose calibration between visual and inertial sensors," in *InerVis*, Barcelona, Spain, Apr. 2005.
- [9] P. Lang and A. Pinz, "Calibration of hybrid vision/inertial tracking systems," in *InerVis*, Barcelona, Spain, Apr. 2005.
- [10] R. Tsai and R. Lenz, "A new technique for fully autonomous and efficient 3d robotics hand/eye calibration," *IEEE Transactions on Robotics and Automation*, vol. 5, no. 3, pp. 345–358, 1989.
- [11] K. Daniilidis, "Hand-eye calibration using dual quaternions," *Int. Journ. Robotics Res.*, vol. 18, no. 3, pp. 286–298, 1999.
- [12] J. Chou and M. Kamel, "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.
- [13] B. K. P. Horn, "Closed-form solution of the absolute orientation using quaternions," *J. Optical Soc. Am. A*, vol. 4, pp. 629–642, April 1987.
- [14] E. J. Lefferts, F. L. Markley, and M. D. Shuster, "Kalman filtering for spacecraft attitude estimation," *Journal of Guidance, Control, and Dynamics*, vol. 5, no. 5, pp. 417–429, 1982.
- [15] N. Trawny and S. I. Roumeliotis, "Indirect kalman filter for 3D pose estimation," Dept. of Computer Science & Engineering, University of Minnesota, Minneapolis, MN, Tech. Rep., March 2005.
- [16] P. S. Maybeck, *Stochastic Models, Estimation and Control*. New York: Academic Press, 1979, vol. 2.
- [17] F. M. Mirzaei and S. I. Roumeliotis, "IMU-camera calibration: Observability analysis," Dept. of Computer Science & Engineering, University of Minnesota, Minneapolis, MN, Tech. Rep., Jan. 2007. [Online]. Available: <http://www.cs.umn.edu/research/mars/tr/reports/Mirzaei07.pdf>