Visual-Inertial Simultaneous Localization, Mapping and Sensor-to-Sensor Self-Calibration

Jonathan Kelly and Gaurav S. Sukhatme

***Abstract*— Visual and inertial sensors, in combination, are well-suited for many robot navigation and mapping tasks. However, correct data fusion, and hence overall system performance, depends on accurate calibration of the 6-DOF transform between the sensors (one or more camera(s) and an inertial measurement unit). Obtaining this calibration information is typically difficult and time-consuming. In this paper, we describe an algorithm, based on the unscented Kalman filter (UKF), for camera-IMU simultaneous localization, mapping and sensor relative pose self-calibration. We show that the sensor-to-sensor transform, the IMU gyroscope and accelerometer biases, the local gravity vector, and the metric scene structure can all be recovered from camera and IMU measurements alone. This is possible without any prior knowledge about the environment in which the robot is operating. We present results from experiments with a monocular camera and a low-cost solid-state IMU, which demonstrate accurate estimation of the calibration parameters and the local scene structure.**

I. Introduction

The majority of future robots will be mobile, and will need to navigate reliably in dynamic and unknown environments. Recent work has shown that visual and inertial sensors, in combination, can be used to estimate egomotion with high fidelity [1]–[3]. However, the six degrees-of-freedom (6-DOF) transform between the sensors must be accurately known for measurements to be properly fused in the navigation frame. Calibration of this transform is typically a complex and time-consuming process, which must be repeated whenever the sensors are repositioned or significant mechanical stresses are applied. Ideally, we would like to build 'power-up-and-go' robotic systems that are able to operate autonomously for long periods without requiring tedious manual (re-) calibration.

In this paper, we describe our work on combining visual and inertial sensing for navigation tasks, with an emphasis on the ability to self-calibrate the sensor-to-sensor transform between a camera and an inertial measurement unit (IMU) *in the field*. Self-calibration refers to the process of using imperfect (noisy) measurements from the sensors themselves to improve our estimates of related system parameters.

Camera-IMU self-calibration is challenging for several unique reasons. IMU measurements, i.e. the outputs of three orthogonal angular rate gyroscopes and three orthogonal accelerometers, can in theory be integrated to determine the change in sensor pose over time. In practice, however, all inertial sensors, and particularly low-cost MEMS[1] units, are subject to drift. The existence of time-varying drift terms (biases) implies that IMU measurements are correlated in time. Further, the IMU accelerometers sense the force of gravity in addition to forces which accelerate the platform. The magnitude of the gravity vector is typically large enough to dominate other measured accelerations. If the orientation of the IMU with respect to gravity is unknown or is misestimated, then the integrated sensor pose will diverge rapidly from the true pose.

Camera image measurements, unlike those from an IMU, reference the external environment and are therefore largely immune to drift.[2] However, cameras are bearing-only sensors, which require both parallax and a known baseline to determine the absolute depths of landmarks. This baseline distance must be provided by another sensor. Our goal is to demonstrate that it is possible to self-calibrate the camera-IMU transform, while dealing with IMU drift, the unknown IMU orientation with respect to gravity, and the lack of scale information in the camera measurements. As part of the calibration procedure, we also simultaneously localize the camera-IMU platform and (if necessary) build a map of the environment.

Following our earlier work [4], we formulate camera-IMU relative pose calibration as a filtering problem. Initially, we consider *target-based* calibration, where the camera views a known planar calibration target. We then extend our filtering algorithm to handle the situation in which the positions of the landmarks are not *a priori* known, and so must also be estimated – a problem we call *target-free* calibration. As our main contribution, we show that the 6-DOF camera-IMU transform, IMU biases, gravity vector, and the metric scene structure can *all* be estimated simultaneously, given sufficiently exciting motion of the sensor platform. To our knowledge, this is the first time that this result has been presented. Additionally, we demonstrate that accurate estimates of the calibration parameters and the metric scene structure can be obtained using an inexpensive solid-state IMU.

The remainder of the paper is organized as follows. We discuss related work in Section II, and review several important results on the observability of camera-IMU self-calibration in Section III. In Section IV, we describe our unscented Kalman filter-based estimation algorithm. We then

This work was funded in part by the National Science Foundation under grants CNS-0325875, CNS-0540420, CNS-0520305 and CCF-0120778 and by a gift from the Okawa Foundation. Jonathan Kelly was supported by an Annenberg Fellowship from the University of Southern California and by an NSERC postgraduate scholarship from the Government of Canada.

Jonathan Kelly and Gaurav S. Sukhatme are with the Robotic Embedded Systems Laboratory, University of Southern California, Los Angeles, California, USA 90089-0781 {jonathsk, gaurav}@usc.edu

[1] MEMS is an acronym for *microelectromechanical systems*.

[2] That is, camera measurements are immune to drift as long as the same static landmarks remain within the camera's field of view.

give an overview of our calibration experiments in Section V, and present results from those experiments in Section VI. Finally, we offer some conclusions and directions for future work in Section VII.

II. Related Work

Several visual-inertial calibration techniques have been proposed in the literature. For example, Lang and Pinz [5] uses a constrained nonlinear optimization algorithm to solve for the rotation angle between a camera and an IMU. The algorithm operates by comparing the change in angle measured by the camera (relative to several external markers) with the integrated IMU gyro outputs. By rotating the camera and the IMU together, it is possible to find the angular offset which best aligns the sensor frames.

Lobo and Dias [6] describes a camera-IMU calibration procedure in which the relative orientation and relative translation of the sensors are determined independently. The procedure requires a pendulum unit and a turntable, making it impractical for larger robot platforms, and does not account for time-varying IMU biases. A further drawback is that separate calibration of rotation and translation decouples their estimates, and therefore ignores any correlations that may exist between the parameters.

Closely related to our own work is the camera-IMU calibration algorithm proposed by Mirzaei and Roumeliotis [7]. They track corner points on a planar calibration target, and fuse these image measurements with IMU data in an iterated extended Kalman filter to estimate the relative pose of the sensors as well as the IMU biases. A similar approach for calibrating the relative transform between a spherical camera and an IMU is discussed in [8]. Both of these techniques require a known calibration object, however.

Jones, Vedaldi and Soatto present an observability analysis in [9] which shows that the camera-IMU relative pose, gravity vector and scene structure can be recovered from camera and IMU measurements. Their work assumes that the IMU biases are static over the calibration interval – although drift in the biases can be significant even over short durations, particularly for the low-cost inertial sensors considered in this paper. Our technique, in the spirit of [10], does not require *any* additional apparatus in the general case, and explicitly models uncertainty in the gravity vector and in the gyroscope and accelerometer biases.

III. Observability of Localization, Mapping and Relative Pose Calibration

Correct calibration of the transform between the camera and the IMU depends on the *observability* of the relevant system states. That is, we must be able recover the state values from the measured system outputs, the control inputs, and a finite number of their time derivatives [11]. Observability is a necessary condition for any filtering algorithm to converge to an unbiased state estimate [12]. Prior work on the observability of camera-IMU relative pose calibration includes [9], [13].

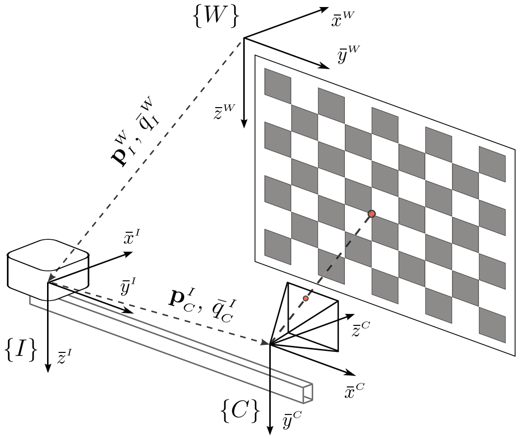


Fig. 1. Relationship between the world $\{W\}$, IMU $\{I\}$, and camera $\{C\}$ reference frames, for target-based calibration. The goal of the calibration procedure is to determine the transform $(\mathbf{p}_C^I, \bar{q}\,_C^I)$ between the camera and IMU. A previous version of this figure appeared in [4].

In [14], we show that, in the presence of a known calibration target, the 6-DOF calibration parameters, IMU biases and the local gravity vector are observable from camera and IMU measurements only. Further, they are observable independent of the linear motion of the camera-IMU platform (as shown in [13] for the biases-only case). Our analysis is based on a differential geometric characterization of observability, and relies a matrix rank test originally introduced by Hermann and Krener [15].

The task of calibrating the relative transform between a single camera and an IMU is more complicated when a known calibration object is not available. For the target-free case, we instead select a set of salient point features in one or more camera images, and use this set as a static reference for calibration. The 3-D positions of the landmarks corresponding to the image features will initially be unknown, however, and so must also be estimated.

The general problem of estimating both camera motion and scene structure has been extensively studied in computer vision and in robotics. Chiuso et al. [16] shows that monocular structure-from-motion (SFM) is observable up to an unknown similarity transform from image measurements alone. If we choose the initial camera position as the origin of our world frame, and fix the initial camera orientation (relative to three or more noncollinear points on the image plane), then following [9], [16], scene structure is observable up to an unknown scale. We prove as our main result in [14] that, if we 'lock down' the initial camera orientation, it is possible to observe the relative pose of the camera and the IMU, the gyroscope and accelerometer biases, the gravity vector *and* the local scene structure. This result holds as long as the IMU measures two nonzero angular rates and two nonzero accelerations (i.e. along at least two axes). Locking down the initial orientation can introduce a small bias in the structure measurements – by averaging several camera observations at the start of the calibration procedure, we have found that it is possible to make this bias negligible.

IV. Calibration Algorithm

We initially describe our system model below, for both target-based and self-calibration, and then review unscented filtering and our calibration algorithm. Three separate reference frames are involved:

1) the *camera frame* $\{C\}$, with its origin at the optical center of the camera and with the z-axis aligned with the optical axis of the lens,
2) the *IMU frame* $\{I\}$, with its origin at the center of the IMU body, in which linear accelerations and angular rates are measured, and
3) the *world frame* $\{W\}$, which serves as an absolute reference for both the camera and the IMU.

We treat the world frame as an inertial frame. As a first step, we must choose an origin for this frame. For the target-based case, we will select the upper leftmost corner point on the target as the origin; for self-calibration, we treat the initial camera position as the origin of the world frame (cf. Section III).

Because the world frame is defined with respect to either the calibration target or the initial camera pose, the relationship between the frame and the local gravity vector can be arbitrary, i.e. it will depend entirely on how the target or the camera is oriented. It is possible to manually align the vertical axis of the calibration target with the gravity vector, however this alignment will not, in general, be exact. This is one reason why we estimate the gravity vector during calibration.

We parameterize orientations in our state vector using unit quaternions. A unit quaternion is a four-component hypercomplex number, consisting of both a scalar part q_0 and a vector part $\mathbf{q}$:

$$\bar{q} \equiv q_0 + \mathbf{q} = q_0 + q_1\bar{\mathbf{i}} + q_2\bar{\mathbf{j}} + q_3\bar{\mathbf{k}} \tag{1}$$

$$\|\,\bar{q}\,\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = 1 \tag{2}$$

where $\bar{\mathbf{i}}$, $\bar{\mathbf{j}}$ and $\bar{\mathbf{k}}$ are the quaternion basis vectors.

Unit quaternions have several advantages over other orientation parameterizations, e.g. the map from the unit sphere S^3 in $\mathbb{R}^4$ to the rotation group $SO(3)$ is smooth and singularity-free [17]. However, unit quaternions have four components but only three degrees of freedom – this constraint requires special treatment in our estimation algorithm.

A. System Description

We use the UKF to simultaneously estimate the pose of the IMU in the world frame, the IMU biases, the gravity vector, and the position and orientation of the camera with respect to the IMU. The 26×1 *sensor* state vector is:

$$\begin{aligned}\mathbf{x}_s(t) \equiv \big[&(\mathbf{p}_I^W(t))^T \quad (\bar{q}_I^{\,W}(t))^T \quad (\mathbf{v}^W(t))^T \quad \ldots \\ &(\mathbf{b}_g(t))^T \quad (\mathbf{b}_a(t))^T \quad (\mathbf{g}^W)^T \quad (\mathbf{p}_C^I)^T \quad (\bar{q}_C^{\,I})^T\big]^T\end{aligned} \tag{3}$$

where $\mathbf{p}_I^W$ is the position of the IMU in the world frame, $\bar{q}_I^{\,W}$ is the orientation of the IMU frame relative to the world frame, $\mathbf{v}^W$ is the linear velocity of the IMU in the world frame, $\mathbf{b}_g$ and $\mathbf{b}_a$ are the gyroscope and accelerometer biases, respectively, and $\mathbf{g}^W$ is the gravity vector in the world frame. The remaining entries, $\mathbf{p}_C^I$ and $\bar{q}_C^{\,I}$, define the position and orientation of the camera frame relative to the IMU frame; these values are *parameters*, i.e. they do not change over time.

For target-free self-calibration, we also estimate the positions of a series of static point landmarks in the environment. The complete state vector for the target-free case is:

$$\mathbf{x}(t) \equiv \big[\mathbf{x}_s^T(t) \quad (\mathbf{p}_{l_1}^W)^T \quad \ldots \quad (\mathbf{p}_{l_n}^W)^T\big]^T \tag{4}$$

where $\mathbf{p}_{l_i}^W$ is the 3×1 vector that defines the position of landmark i in the world frame, $i = 1, \ldots, n$, for $n \geq 3$. The complete target-free state vector has size $(26 + 3n) \times 1$.

In our experiments, we have found it sufficient to use Cartesian coordinates to specify landmark positions. We initialize each landmark at a nominal depth and with a large variance along the camera ray axis, at the camera position where the landmark is first observed. If the landmark depths vary significantly (by several meters or more), an inverse-depth parameterization may be more appropriate [18].

1) Process Model: Our filter process model uses the IMU angular velocities and linear accelerations as substitutes for control inputs in the system dynamics equations [19]. We model the IMU gyroscope and accelerometer biases as Gaussian random walk processes driven by the white noise vectors $\mathbf{n}_{gw}$ and $\mathbf{n}_{aw}$. Gyroscope and accelerometer measurements are assumed to be corrupted by the zero-mean white Gaussian noise, defined by the vectors $\mathbf{n}_g$ and $\mathbf{n}_a$, respectively. The evolution of the system state is described by:

$$\dot{\mathbf{p}}_I^W = \mathbf{v}^W \qquad \dot{\mathbf{v}}^W = \mathbf{a}^W \qquad \dot{\bar{q}}_I^{\,W} = \frac{1}{2}\boldsymbol{\Omega}(\boldsymbol{\omega}^I)\,\bar{q}_I^{\,W} \tag{5}$$

$$\dot{\mathbf{b}}_g = \mathbf{n}_{gw} \qquad \dot{\mathbf{b}}_a = \mathbf{n}_{aw} \qquad \dot{\mathbf{g}}^W = \mathbf{0}_{3\times1} \tag{6}$$

$$\dot{\mathbf{p}}_C^I = \mathbf{0}_{3\times1} \qquad \dot{\bar{q}}_C^{\,I} = \mathbf{0}_{4\times1} \tag{7}$$

where for brevity we do not indicate the dependence on time. The term $\boldsymbol{\Omega}(\boldsymbol{\omega}^I)$ above is the 4×4 quaternion kinematical matrix which relates the time rate of change of the orientation quaternion to the IMU angular velocity. The vectors $\mathbf{a}^W$ and $\boldsymbol{\omega}^I$ are the linear acceleration of the IMU in the world frame and the angular velocity of the IMU in the IMU frame, respectively. These terms are related to the *measured* IMU angular velocity, ω_m, and linear acceleration, $\mathbf{a}_m$, by:

$$\boldsymbol{\omega}_m = \boldsymbol{\omega}^I + \mathbf{b}_g + \mathbf{n}_g \tag{8}$$

$$\mathbf{a}_m = \mathbf{C}^T(\bar{q}_I^{\,W})\,(\mathbf{a}^W - \mathbf{g}^W) + \mathbf{b}_a + \mathbf{n}_a \tag{9}$$

where $\mathbf{C}(\bar{q}_I^{\,W})$ is the direction cosine matrix that describes the orientation of the IMU frame with respect to the world frame. We propagate the system state forward in time until the next camera or IMU update using fourth-order Runge-Kutta integration of (5) to (7) above.

2) Measurement Model: As the sensors move, the camera captures images of tracked landmarks in the environment. Projections of the these landmarks can be used to determine the position and the orientation of the camera in the world frame [20].[3] We use an ideal projective (pinhole) camera model, and rectify each image initially to remove lens distortions. The camera intrinsic and distortion parameters may either be calibrated separately beforehand, or calibrated using a subset of the images acquired for the target-based camera-IMU procedure. Self-calibration of the camera is also possible, although this is beyond the scope of work presented here.

Measurement $\mathbf{z}_i$ is the projection of landmark i, at position $\mathbf{p}^C_{l_i} = \begin{bmatrix} x_i & y_i & z_i \end{bmatrix}^T$ in the camera frame, onto the image plane:

$$\mathbf{p}^C_{l_i} = \begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \mathbf{C}^T(\bar{q}^{\,I}_{C})\,\mathbf{C}^T(\bar{q}^{\,W}_{I})\left(\mathbf{p}^W_{l_i} - \mathbf{p}^W_I\right) - \mathbf{C}^T(\bar{q}^{\,I}_{C})\,\mathbf{p}^I_C \tag{10}$$

$$\mathbf{z}_i = \begin{bmatrix} u_i \\ v_i \end{bmatrix} = \begin{bmatrix} x'_i \\ y'_i \end{bmatrix} + \boldsymbol{\eta}_i, \quad \begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} = \boldsymbol{\mathcal{K}} \begin{bmatrix} x_i/z_i \\ y_i/z_i \\ 1 \end{bmatrix} \tag{11}$$

where $\begin{bmatrix} u_i & v_i \end{bmatrix}^T$ is the vector of observed image coordinates, $\boldsymbol{\mathcal{K}}$ is the 3×3 camera intrinsic calibration matrix [21], and $\boldsymbol{\eta}_i$ is a Gaussian measurement noise vector with covariance matrix $\mathbf{R}_i = \sigma_i^2\,\mathbf{I}_{2\times 2}$.

When several landmarks are visible in one image, we stack the individual measurements to form a single measurement vector $\mathbf{z} = \begin{bmatrix} \mathbf{z}_1^T & \ldots & \mathbf{z}_n^T \end{bmatrix}^T$ and the associated block-diagonal covariance matrix $\mathbf{R} = \mathtt{diag}\begin{pmatrix} \mathbf{R}_1 & \ldots & \mathbf{R}_n \end{pmatrix}$. This vector can then be processed by our filtering algorithm in one step.

B. Unscented Filtering

The UKF is a Bayesian filtering algorithm which propagates and updates the system state using a set of deterministically-selected sample points called *sigma points*. These points, which lie on the covariance contours in state space, capture the mean and covariance of the state distribution. The filter applies the *unscented transform* to the sigma points, propagating each point through the (nonlinear) process and measurement models, and then computes the weighted averages of the transformed points to determine the posterior state mean and state covariance. This is a form of statistical local linearization, which produces more accurate estimates than the analytic local linearization employed by the extended Kalman filter (EKF).

We use a continuous-discrete formulation of the UKF, in which the sigma points are propagated forward by integration, while measurement updates occur at discrete time steps. Our filter implementation augments the state vector and state covariance matrix with a process noise component, as described in [22]:

$$\mathbf{x}_a(t) \equiv \begin{bmatrix} \mathbf{x}(t) \\ \mathbf{n}(t) \end{bmatrix} \tag{12}$$

where $\mathbf{x}_a(t)$ is the augmented state vector of size N at time t, and $\mathbf{n}(t) = \begin{bmatrix} \mathbf{n}_{gw} & \mathbf{n}_{aw} & \mathbf{n}_g & \mathbf{n}_a \end{bmatrix}^T$ is the 12×1 process noise vector. We employ the scaled form of the unscented transform [23], which requires a scaling term:

$$\lambda = \alpha^2(N + \beta) - N \tag{13}$$

The α parameter controls the spread of the sigma points about the state mean and is usually set to a small positive value (10^{-3} in our implementation). The β parameter is used to incorporate corrections to higher-order terms in the Taylor series expansion of the state distribution; setting $\beta = 2$ minimizes the fourth-order error for jointly Gaussian distributions.

At time $t - \tau$, immediately after the last measurement update, the augmented state mean $\hat{\mathbf{x}}^+_a(t-\tau)$ and augmented state covariance matrix $\mathbf{P}^+_a(t-\tau)$ are:

$$\hat{\mathbf{x}}^+_a(t-\tau) \equiv \begin{bmatrix} \hat{\mathbf{x}}^+(t-\tau) \\ \mathbf{0}_{12\times 1} \end{bmatrix}, \quad \mathbf{P}^+_a(t-\tau) \equiv \begin{bmatrix} \mathbf{P}^+(t-\tau) & \mathbf{0} \\ \mathbf{0} & \mathbf{Q}_c \end{bmatrix} \tag{14}$$

where τ is the update time interval and $\mathbf{Q}_c$ is the covariance matrix for the noise vector $\mathbf{n}(t)$. The augmented state vector $\hat{\mathbf{x}}^+_a(t-\tau)$ is used to generate the set of sigma points according to:

$$^0\boldsymbol{\chi}_a(t-\tau) = \hat{\mathbf{x}}^+_a(t-\tau) \tag{15}$$

$$^k\boldsymbol{\chi}_a(t-\tau) = \hat{\mathbf{x}}^+_a(t-\tau) + {}^j\mathbf{S}(t-\tau), \quad j{=}k{=}1,\ldots,N \tag{16}$$

$$^k\boldsymbol{\chi}_a(t-\tau) = \hat{\mathbf{x}}^+_a(t-\tau) - {}^j\mathbf{S}(t-\tau), \quad j{=}1,\ldots,N,\;\; k{=}N{+}1,\ldots,2N \tag{17}$$

$$\mathbf{S}(t-\tau) = \sqrt{(\lambda + N)\,\mathbf{P}^+_a(t-\tau)} \tag{18}$$

where ${}^j\mathbf{S}$ denotes the j^{th} column of the matrix $\mathbf{S}$. The matrix square root of $\mathbf{P}^+_a(t-\tau)$ can be found by Cholesky decomposition [24]. The associated sigma point weight values are:

$$^0W_m = \lambda/(\lambda + N) \tag{19}$$

$$^0W_c = \lambda/(\lambda + N) + (1 - \alpha^2 + \beta) \tag{20}$$

$$^jW_m = {}^jW_c = \frac{1}{2\,(\lambda + N)}, \quad j = 1, \ldots, 2N \tag{21}$$

Individual sigma points are propagated through the augmented nonlinear process model function $\mathbf{f}_a$, which incorporates process noise in the propagation equations, and the weights above are used to calculate the *a priori* state estimate and covariance matrix at time t:

$$^i\boldsymbol{\chi}_a(t) = \mathbf{f}_a\big({}^i\boldsymbol{\chi}_a(t-\tau)\big), \quad i = 0, \ldots, 2N \tag{22}$$

$$\hat{\mathbf{x}}^-(t) = \sum_{i=0}^{2N} {}^iW_m\,{}^i\boldsymbol{\chi}(t) \tag{23}$$

$$\mathbf{P}^-(t) = \sum_{i=0}^{2N} {}^iW_c\left({}^i\boldsymbol{\chi}(t) - \hat{\mathbf{x}}^-(t)\right)\left({}^i\boldsymbol{\chi}(t) - \hat{\mathbf{x}}^-(t)\right)^T \tag{24}$$

When a measurement arrives, we determine the predicted measurement vector by propagating each sigma point

[3] For target-free calibration, the position is determined up to scale only.

through the nonlinear measurement model function $\mathbf{h}$:

$$ {}^{i}\boldsymbol{\gamma}(t) = \mathbf{h}\left({}^{i}\boldsymbol{\chi}(t)\right), \quad i = 0, \ldots, 2N \tag{25} $$

$$ \hat{\mathbf{z}}(t) = \sum_{i=0}^{2N} {}^{i}W_m \, {}^{i}\boldsymbol{\gamma}(t) \tag{26} $$

We then perform a state update by computing the Kalman gain matrix $\mathbf{K}(t)$ and the *a posteriori* state vector and state covariance matrix:

$$ \mathbf{P}_{\hat{\mathbf{x}}\hat{\mathbf{z}}}(t) = \sum_{i=0}^{2N} {}^{i}W_c \left({}^{i}\boldsymbol{\chi}(t) - \hat{\mathbf{x}}^{-}(t)\right)\left({}^{i}\boldsymbol{\gamma}(t) - \hat{\mathbf{z}}(t)\right)^{T} \tag{27} $$

$$ \mathbf{P}_{\hat{\mathbf{z}}\hat{\mathbf{z}}}(t) = \sum_{i=0}^{2N} {}^{i}W_c \left({}^{i}\boldsymbol{\gamma}(t) - \hat{\mathbf{z}}(t)\right)\left({}^{i}\boldsymbol{\gamma}(t) - \hat{\mathbf{z}}(t)\right)^{T} \tag{28} $$

$$ \mathbf{K}(t) = \mathbf{P}_{\hat{\mathbf{x}}\hat{\mathbf{z}}}(t)\left(\mathbf{P}_{\hat{\mathbf{z}}\hat{\mathbf{z}}}(t) + \mathbf{R}(t)\right)^{-1} \tag{29} $$

$$ \hat{\mathbf{x}}^{+}(t) = \hat{\mathbf{x}}^{-}(t) + \mathbf{K}(t)\left(\mathbf{z}(t) - \hat{\mathbf{z}}(t)\right) \tag{30} $$

$$ \mathbf{P}^{+}(t) = \mathbf{P}^{-}(t) - \mathbf{K}(t)\mathbf{P}_{\hat{\mathbf{z}}\hat{\mathbf{z}}}(t)\mathbf{K}^{T}(t) \tag{31} $$

where $\mathbf{P}_{\hat{\mathbf{x}}\hat{\mathbf{z}}}(t)$ and $\mathbf{P}_{\hat{\mathbf{z}}\hat{\mathbf{z}}}(t)$ are the state-measurement cross-covariance matrix and the predicted measurement covariance matrix, respectively, while $\mathbf{R}(t)$ is the measurement covariance matrix for the current observation(s).

C. The Unscented Quaternion Estimator

The UKF computes the predicted state vector as the weighted *barycenteric mean* of the sigma points. For unit quaternions, however, the barycenter of the transformed sigma points will often *not* represent the correct mean. In particular, the weighted average of several unit quaternions may not be a unit quaternion.

There are several possible ways to enforce the quaternion unit norm constraint within the UKF, for example by incorporating pseudo-observations or by projecting the unconstrained time and measurement updates onto the quaternion constraint surface [25]. We follow the method described in [26] and reparameterize the state vector to incorporate a multiplicative, three-parameter orientation *error state* vector in place of the unit quaternions $\bar{q}_{I}^{W}$ and $\bar{q}_{C}^{I}$. This approach, called the USQUE (UnScented QUaternion Estimator) filter in [26], defines a multiplicative local error quaternion:

$$ \delta\bar{q} \equiv \begin{bmatrix} \delta q_0 & \delta\mathbf{q}^{T} \end{bmatrix}^{T} \tag{32} $$

and the following three-component vector of modified Rodrigues parameters (MRPs), derived from the error quaternion:

$$ \delta\mathbf{e} = \frac{\delta\mathbf{q}}{1 + \delta q_0} \tag{33} $$

The MRP vector is an unconstrained three-parameter rotation representation, which is singular at 2π, and can be expressed in axis-angle form as:

$$ \delta\mathbf{e} \equiv \bar{\mathbf{u}} \tan(\theta/4) \tag{34} $$

where $\bar{\mathbf{u}}$ defines the rotation axis, and θ is the rotation angle. The inverse transformation, from the MRP vector to the error quaternion $\delta\bar{q}$, is given by:

$$ \delta q_0 = \frac{1 - \left\| \delta\mathbf{e} \right\|^{2}}{1 + \left\| \delta\mathbf{e} \right\|^{2}} \tag{35} $$

$$ \delta\mathbf{q} = (1 + \delta q_0)\,\delta\mathbf{e} \tag{36} $$

From the full sensor state vector (3), we define the modified 24×1 sensor error state vector $\mathbf{x}_{se}(t)$ as:

$$ \begin{aligned} \mathbf{x}_{se}(t) = \big[&(\mathbf{p}_{I}^{W}(t))^{T} \quad (\delta\mathbf{e}_{I}^{W}(t))^{T} \quad (\mathbf{v}^{W}(t))^{T} \quad \ldots \\ &(\mathbf{b}_{g}(t))^{T} \quad (\mathbf{b}_{a}(t))^{T} \quad (\mathbf{g}^{W})^{T} \quad (\mathbf{p}_{C}^{I})^{T} \quad (\delta\mathbf{e}_{C}^{I})^{T} \big]^{T} \end{aligned} \tag{37} $$

where $\delta\mathbf{e}_{I}^{W}$ and $\delta\mathbf{e}_{C}^{I}$ are the MRP error state vectors corresponding to the quaternions $\bar{q}_{I}^{W}$ and $\bar{q}_{C}^{I}$.

Throughout the calibration procedure, the filter maintains an estimate of the full 26×1 sensor state vector and the 24×24 error state covariance matrix. For the orientation quaternions $\bar{q}_{I}^{W}$ and $\bar{q}_{C}^{I}$, we store the covariance matrices for the MRP error state representations.

At the start of each propagation step, we compute the sigma points for the error state according to (15)–(17), setting the mean error state MRP vectors to:

$$ {}^{0}\delta\,\hat{\mathbf{e}}_{I}^{W}(t-\tau) = \mathbf{0}_{3\times1}, \quad {}^{0}\delta\,\hat{\mathbf{e}}_{C}^{I}(t-\tau) = \mathbf{0}_{3\times1} \tag{38} $$

where we indicate the component of the state vector that belongs to a specific sigma point by prefixing the vector with a superscripted index (zero above). We follow this convention throughout this section.

To propagate the IMU orientation quaternion $\hat{\bar{q}}_{I}^{W}$ forward in time, we compute the local error quaternion ${}^{j}\delta\hat{\bar{q}}_{I}^{W}(t-\tau)$ from the MRP vector associated with sigma point j using (35)–(36), and then the full orientation quaternion from the error quaternion:

$$ {}^{j}\hat{\bar{q}}_{I}^{W}(t-\tau) = {}^{j}\delta\hat{\bar{q}}_{I}^{W}(t-\tau) \otimes \hat{\bar{q}}_{I}^{W+}(t-\tau), \;\; j = 1, \ldots, 2N \tag{39} $$

where the $\otimes$ operator denotes quaternion multiplication. The other components of the sigma points are determined by addition or subtraction directly. Each sigma point, including the corresponding full IMU orientation quaternion, is then propagated through the augmented process model function $\mathbf{f}_{a}$ from time $t-\tau$ to time t.

We determine the orientation error quaternions at time t by reversing the procedure above, using the propagated mean quaternion:

$$ {}^{j}\delta\hat{\bar{q}}_{I}^{W}(t) = {}^{j}\hat{\bar{q}}_{I}^{W}(t) \otimes \left({}^{0}\hat{\bar{q}}_{I}^{W}(t)\right)^{-1} \quad j = 1, \ldots, 2N \tag{40} $$

and finally compute the orientation MRP error vector using (33). Note that this is required during the propagation step for the IMU orientation quaternion only, as the camera-IMU orientation quaternion does not change over time. We can then compute the updated *a priori* error state vector and error state covariance matrix using (23) and (24).

We store the orientation quaternions from the propagation step; when a measurement arrives, we compute the predicted measurement vector for each sigma point using the nonlinear measurement function $\mathbf{h}$. The error quaternion for the camera-IMU orientation is determined according to:

$$ {}^{j}\delta\hat{\bar{q}}^{\,I}_{C}(t) = {}^{j}\hat{\bar{q}}^{\,I}_{C}(t) \otimes \left({}^{0}\hat{\bar{q}}^{\,I}_{C}(t)\right)^{-1} \quad j = 1, \ldots, 2N \tag{41} $$

and the MRP error state vectors are found using (33). We then compute (27)–(29) and the updated *a posteriori* error state vector and error state covariance matrix. As a final step, we use the updated mean MRP error state vectors to compute the mean error quaternions, and the full state vector orientation quaternions.

D. Filter Initialization

At the start of the calibration procedure, we compute an initial estimate of the sensor state vector. For target-based calibration, we first generate initial estimates of the camera position $\hat{\mathbf{p}}^{W}_{C}$ and orientation $\hat{\bar{q}}^{\,W}_{C}$ in the world frame. Given the known positions of the corner points on the calibration target and their image projections, we use Horn's method [27] to compute the camera orientation quaternion in closed form (after coarse initial triangulation). This is followed by an iterative nonlinear least squares refinement of the translation and orientation estimates; the refinement step provides the 3×3 MRP error covariance matrix for the camera orientation in the world frame.

An initial estimate of the camera pose relative to the IMU is also required. We use hand measurements of the relative pose for the experiments described in this paper – however this information may in many cases be available from CAD drawings or other sources. Using the estimate of the camera pose in the world frame and the estimate of the relative pose of the camera with respect to the IMU, we can then calculate an initial estimate of the IMU pose in the world frame, according to:

$$ \hat{\mathbf{p}}^{W}_{I} = \hat{\mathbf{p}}^{W}_{C} - \mathbf{C}(\hat{\bar{q}}^{\,W}_{C})\mathbf{C}^{T}(\hat{\bar{q}}^{\,I}_{C})\,\hat{\mathbf{p}}^{I}_{C} \tag{42} $$

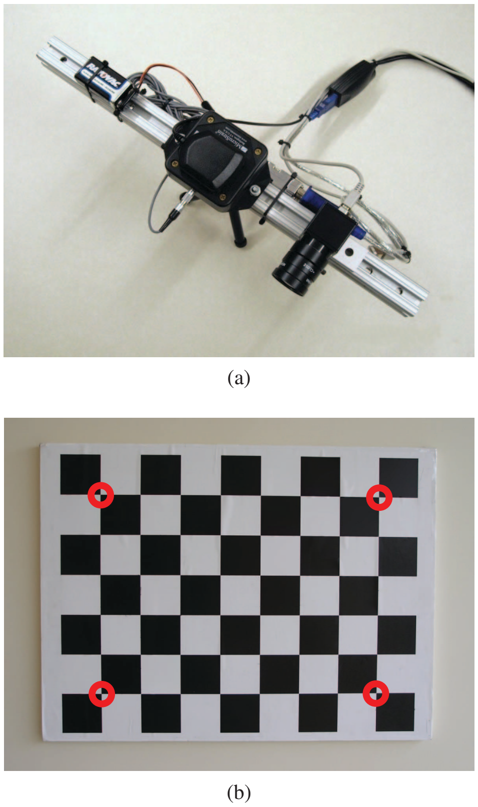
$$ \hat{\bar{q}}^{\,W}_{I} = \hat{\bar{q}}^{W}_{C} \otimes \left(\hat{\bar{q}}^{\,I}_{C}\right)^{-1} \tag{43} $$

For target-free calibration, the initialization procedure is the same as that above, except that the initial camera position is set as the origin of the world frame, and the initial camera orientation is chosen arbitrarily. The camera pose has zero initial uncertainty in this case.

As part of the target-free procedure, we estimate the map state (landmark positions in the world frame) as well as the sensor state. We typically select approximately 40 to 60 landmarks as static references for calibration. The positions of the n landmarks in the world frame are initially estimated as:

$$ \mathbf{p}^{W}_{l_i} = d\,\mathcal{K}^{-1}\begin{bmatrix} u_i & v_i & 1 \end{bmatrix}^{T}, \;\; i = 1, \ldots, n \tag{44} $$

assuming that the initial direction cosine matrix that defines the camera orientation is the identity matrix. The value d is a nominal landmark depth, while $\mathcal{K}^{-1}$ is the inverse of the camera intrinsic calibration matrix and $\begin{bmatrix} u_i & v_i \end{bmatrix}^{T}$ is the vector of observed image coordinates for landmark i. The covariance matrix for each landmark is computed by propagating the image plane uncertainty and a large initial variance along the camera ray into 3-D, yielding a covariance ellipsoid in the world frame.



(a)

(b)

Fig. 2. (a) Sensor beam, showing Flea camera (right) and 3DM-G IMU (center). The beam is 40 cm long. (b) Our planar camera calibration target. Each target square is 104 mm on a side. There are 48 interior corner points, which we use as landmarks. The small red circles in the figure identify the four points whose directions we 'lock down' for the self-calibration experiments described in Section V-B.

As a last step, we select three or four highly salient and widely dispersed landmarks as anchors to fix the orientation of the world frame. The covariance ellipsoids for these points are initialized using very small image plane uncertainties. This effectively locks down the initial camera pose and ensures that the full system state is observable.

E. Feature Detection and Matching

We use different feature detection and matching techniques for target-based and target-free calibration. For the target-based case, we first locate candidate points on the target using a fast template matching algorithm. This is followed by a homography-based check to ensure that all points lie on a planar surface. Once we have coarse estimates of the locations of the corner points, we refine those estimates to subpixel accuracy using a saddle point detector [28].

Target-free calibration is normally performed in a previously unseen and unknown environment. For this case, we select a set well-localized point features as landmarks, using a feature selection algorithm such as SIFT [29]. Feature matching between camera frames is performed by comparing the descriptors for all points that lie within a bounded image region. The size of the search region is determined based on

TABLE I

Results for target-based calibration and target-free self-calibration, using data from the same experimental trial. The initial hand-measured (HM) estimate of the camera-IMU transform (x, y, z translation and roll, pitch, yaw orientation) and the final target-based (TB) and target-free (TF) estimates are listed, along with their respective 3σ error bounds.

	$p_x \pm 3\sigma$ (cm)	$p_y \pm 3\sigma$ (cm)	$p_z \pm 3\sigma$ (cm)	Roll $\pm 3\sigma$ (°)	Pitch $\pm 3\sigma$ (°)	Yaw $\pm 3\sigma$ (°)
HM	0.00 ± 12.00	-15.00 ± 15.00	0.00 ± 6.00	90.00 ± 15.00	0.00 ± 15.00	90.00 ± 15.00
TB	6.32 ± 0.54	-14.52 ± 0.43	-1.55 ± 0.44	90.59 ± 0.08	0.80 ± 0.09	89.35 ± 0.08
TF	6.55 ± 0.54	-14.55 ± 0.43	-1.83 ± 0.44	90.56 ± 0.08	0.80 ± 0.09	89.29 ± 0.08

the integrated IMU measurements over the interval between the camera image updates.

V. Experiments

We performed a series of experiments to quantify the accuracy and performance of the target-based and the target-free calibration algorithms. Although we have tested target-free self-calibration in a variety of unstructured environments, in this paper we restrict ourselves to a comparison of the two approaches using the same dataset (images a planar calibration target). The calibration target provides known ground truth against which we can evaluate the quality of the structure recovered by the target-free algorithm.

A. Platform

We use a black and white Flea FireWire camera from Point Grey Research (640×480 pixel resolution), mated to a 4 mm Navitar lens ($58°$ horizontal FOV, $45°$ vertical FOV). Images are captured at a rate of 15 Hz. Our IMU is a MEMS-based 3DM-G unit, manufactured by Microstrain, which provides three-axis angular rate and linear acceleration measurements at approximately 60 Hz. Both sensors are securely bolted to a rigid 40 cm long 8020 aluminum beam, as shown in Figure 2 (a). Our planar calibration target, shown in Figure 2 (b), is 100 cm × 80 cm in size and has 48 interior corner points.

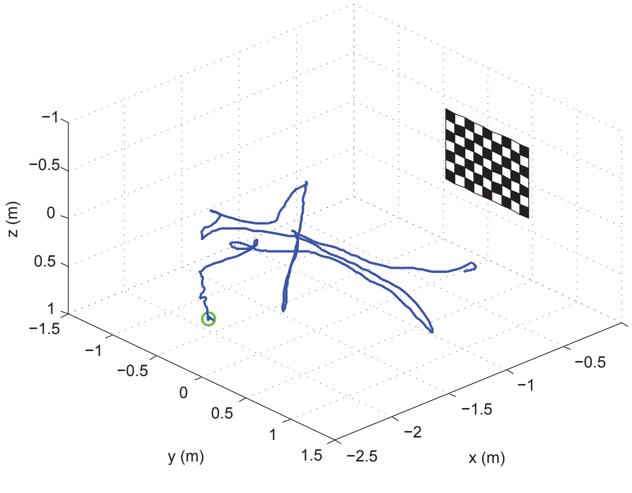


Fig. 3. Path of IMU in the world frame over the first 40 seconds of the calibration experiment. The green circle indicates the starting position.

B. Experimental Procedure

At the start of each experiment, we initialized the filter biases by holding the sensor beam still for approximately 10 seconds. After this settling time, we moved the beam manually through a series of rotation and translation maneuvers, at distances between approximately 1.0 m and 2.5 m from the calibration target. We attempted to keep the target approximately within the camera's field of view throughout the duration of the trial. Image processing and filtering were performed offline.

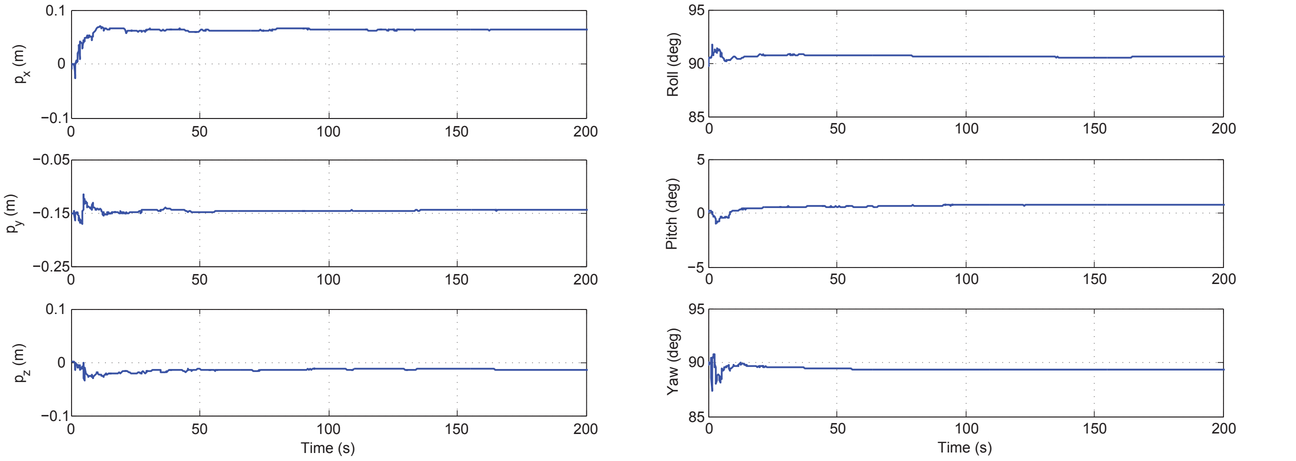
The camera-IMU transform parameters were initialized using hand measurements of the relative pose of the sensors. We used a subset of 25 images acquired during the camera-IMU procedure to calibrate the camera intrinsic and distortion parameters. Each image measurement was assumed to be corrupted by independent white Gaussian noise with a standard deviation of 2.0 pixels along the u and v image axes.

Self-calibration requires us to first lock down the orientation of the world reference frame by fixing the directions to three or more points on the image plane. For our experiments, we chose to fix the directions to the upper and lower left and right corner points on the target (as shown in Figure 2(b)). To do so, we computed the covariance ellipsoids for the corresponding landmarks using very small image plane uncertainties (1×10^{-4} pixels), after averaging the image coordinates over 450 frames (30 seconds) to reduce noise. This averaging was performed using images acquired while the sensor beam was stationary, before the start of an experimental trial. We chose an initial depth of 3.0 m for each of the 48 points on the target, along the corresponding camera ray, with a standard deviation of 0.75 m.

It is important to emphasize that the self-calibration algorithm *does not require* a calibration target. We use the target here for both cases only to evaluate their relative accuracy, with ground truth available.

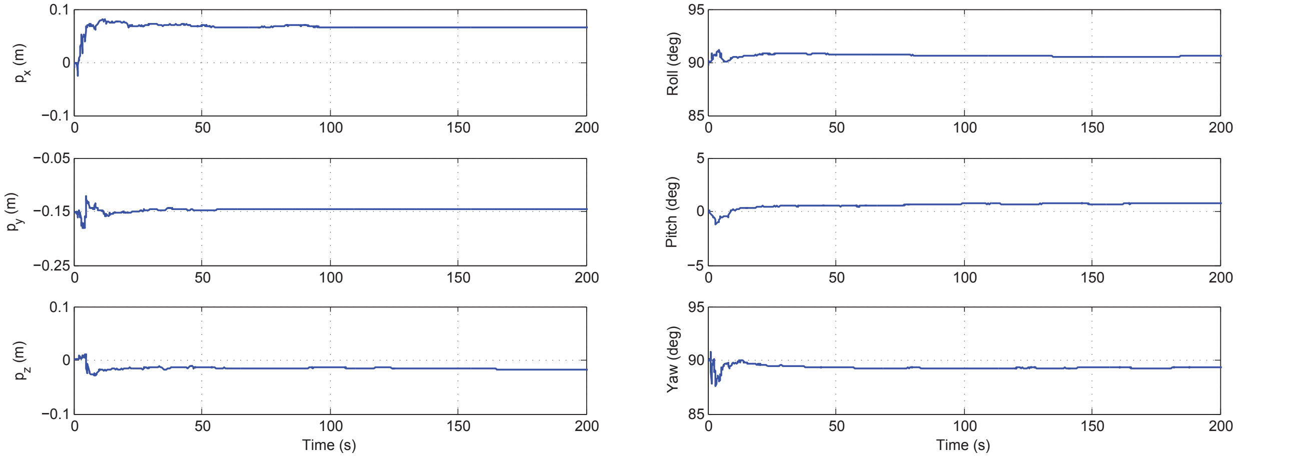
VI. Results and Discussion

We compared the target-based and the target-free self-calibration algorithms using a dataset consisting of 12,046 IMU measurements (6-DOF angular rates and linear accelerations) and 2,966 camera images, acquired over 200 seconds. The calibration target was not completely visible in 20 of the image frames – we simply discarded these frames, leaving 2,946 images for use by the estimation algorithm (in which all 48 target points were successfully identified).



(a) (b)

Fig. 4. (a) Evolution of the IMU-to-camera translation estimate over the calibration time interval (along the x, y and z axes of the IMU frame, from top to bottom) for the target-based calibration procedure. (b) Evolution of the IMU-to-camera orientation estimate (for the roll, pitch and yaw Euler angles that define the orientation of the camera frame relative to the IMU frame, from top to bottom) for the target-based procedure.



(a) (b)

Fig. 5. (a) Evolution of the IMU-to-camera translation estimate over the calibration time interval (along the x, y and z axes of the IMU frame, from top to bottom) for the *target-free self-calibration* procedure. (b) Evolution of the IMU-to-camera orientation estimate (for the roll, pitch and yaw Euler angles that define the orientation of the camera frame relative to the IMU frame, from top to bottom) for the *target-free self-calibration procedure*.

The maximum rotation rate of the IMU was 188°/s, and the maximum linear acceleration (after accounting for gravity) was 6.14 m/s^2. Figure 3 shows the estimated path of the IMU over the first 40 seconds of the experiment.

Table I lists the initial hand-measured (HM) camera-IMU relative pose estimate and the final target-based (TB) and target-free (TF) relative pose estimates. The corresponding plots for the time-evolution of the system state are shown in Figures 4 and 5, respectively. Note that the results are almost identical, and that the target-based relative pose values all lie well within the 3σ bounds of the self-calibrated relative pose values.

As ground truth measurements of the relative pose parameters were not available, we instead evaluated the residual pixel reprojection errors for the hand-measured, target-based and self-calibrated relative pose estimates. We determined these residuals by running the UKF using the respective pose parameters (listed in Table I), without estimating the parameters in the filter. For our hand-measured estimate, the RMS residual error was 4.11 pixels; for the target-based estimate, the RMS residual was 2.23 pixels; for the target-free estimate, the RMS residual was 2.26 pixels. Both the target-based and target-free RMS residuals are much lower than the hand-measured value, indicating that calibration significantly improves motion estimates. Also, the difference in the magnitude of the residuals increases as the camera frame rate is reduced (below 15 Hz).

To evaluate our ability to accurately estimate the landmark positions during self-calibration (i.e. to implement SLAM), we performed a nonlinear least-squares fit of the final landmark position estimates to the ground truth values (based on our manufacturing specifications for the planar target). The RMS error between the true and estimated positions was only 5.7 mm over all 48 target points, and the majority of this error was along the depth direction. This result clearly demonstrates that it is possible to accurately calibrate the camera-IMU transform and to *simultaneously* determine scene structure *in unknown environments* and without any additional apparatus. It is perhaps more impressive that this level of accuracy can be obtained with a sensor such as the 3DM-G, which uses automotive-grade MEMS accelerometers and can be purchased for less than $1000 US.

VII. Conclusions and Ongoing Work

In this paper, we presented an online localization, mapping and relative pose calibration algorithm for visual and inertial sensors. Our results show that it is possible to accurately calibrate the sensor-to-sensor transform *without* using a known calibration target or other calibration object. This work is a step towards building power-up-and-go robotic systems that are able to self-calibrate in the field, during normal operation.

We are currently working to deploy our visual-inertial calibration system on several platforms, including an unmanned aerial vehicle and a humanoid robot. Additionally, we are exploring the possibility of using a similar framework to calibrate the transform between different types of sensors, including laser range finders and GPS units.

References

[1] D. Strelow and S. Singh, "Online Motion Estimation from Visual and Inertial Measurements," in *Proc. 1st Workshop on Integration of Vision and Inertial Sensors (INERVIS'03)*, Coimbra, Portugal, June 2003.

[2] J. Kelly, S. Saripalli, and G. S. Sukhatme, "Combined Visual and Inertial Navigation for an Unmanned Aerial Vehicle," in *Field and Service Robotics: Results of the 6th Int'l Conf. (FSR'07)*, ser. Springer Tracts in Advanced Robotics, C. Laugier and R. Siegwart, Eds. Berlin, Germany: Springer, June 2008, vol. 42/2008, pp. 255–264.

[3] A. I. Mourikis and S. I. Roumeliotis, "A Multi-State Constraint Kalman Filter for Vision-aided Inertial Navigation," in *Proc. IEEE Int'l Conf. Robotics and Automation (ICRA'07)*, Rome, Italy, Apr. 2007, pp. 3565–3572.

[4] J. Kelly and G. S. Sukhatme, "Fast Relative Pose Calibration for Visual and Inertial Sensors," in *Proc. 11th IFRR Int'l Symp. Experimental Robotics (ISER'08)*, Athens, Greece, July 2008.

[5] P. Lang and A. Pinz, "Calibration of Hybrid Vision / Inertial Tracking Systems," in *Proc. 2nd Workshop on Integration of Vision and Inertial Sensors (INERVIS'05)*, Barcelona, Spain, Apr. 2005.

[6] J. Lobo and J. Dias, "Relative Pose Calibration Between Visual and Inertial Sensors," *Int'l J. Robotics Research*, vol. 26, no. 6, pp. 561–575, June 2007.

[7] F. M. Mirzaei and S. I. Roumeliotis, "A Kalman Filter-Based Algorithm for IMU-Camera Calibration: Observability Analysis and Performance Evaluation," *IEEE Trans. Robotics*, vol. 24, no. 5, pp. 1143–1156, Oct. 2008.

[8] J. D. Hol, T. B. Schon, and F. Gustafsson, "Relative Pose Calibration of a Spherical Camera and an IMU," in *Proc. 7th IEEE/ACM Int'l Symp. Mixed and Augmented Reality (ISMAR'08)*, Cambridge, United Kingdom, Sept. 2008, pp. 21–24.

[9] E. Jones, A. Vedaldi, and S. Soatto, "Inertial Structure From Motion with Autocalibration," in *Proc. IEEE Int'l Conf. Computer Vision Workshop on Dynamical Vision*, Rio de Janeiro, Brazil, Oct. 2007.

[10] E. M. Foxlin, "Generalized Architecture for Simultaneous Localization, Auto-Calibration, and Map-Building," in *Proc. IEEE/RSJ Int'l Conf. Intelligent Robots and Systems (IROS'02)*, vol. 1, Lausanne, Switzerland, Oct. 2002, pp. 527–533.

[11] G. Conte, C. H. Moog, and A. M. Perdon, *Algebraic Methods for Nonlinear Control Systems*, 2nd ed. Springer, Dec. 2006.

[12] T. S. Lee, K. P. Dunn, and C. B. Chang, "On the Observability and Unbiased Estimation of Nonlinear Systems," in *System Modeling and Optimization: Proc. 10th IFIP Conf.*, ser. Lecture Notes in Control and Information Sciences. Springer, 1982, vol. 38/1982, pp. 258–266.

[13] F. M. Mirzaei and S. I. Roumeliotis, "IMU-Camera Calibration: Observability Analysis," University of Minnesota, Minneapolis, USA, Tech. Rep. TR-2007-001, Aug. 2007.

[14] J. Kelly, "On the Observability and Self-Calibration of Visual-Inertial Navigation Systems," University of Southern California, Los Angeles, USA, Tech. Rep. CRES-08-005, Nov. 2008.

[15] R. Hermann and A. J. Krener, "Nonlinear Controllability and Observability," *IEEE Trans. Automatic Control*, vol. AC-22, no. 5, pp. 728–740, Oct. 1977.

[16] A. Chiuso, P. Favaro, H. Jin, and S. Soatto, "Structure from Motion Causally Integrated Over Time," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 24, no. 4, pp. 523–535, Apr. 2002.

[17] J. C. K. Chou, "Quaternion Kinematic and Dynamic Differential Equations," *IEEE Trans. Robotics and Automation*, vol. 8, no. 1, pp. 53–64, Feb. 1992.

[18] J. M. M. Montiel, J. Civera, and A. J. Davison, "Unified Inverse Depth Parametrization for Monocular SLAM," in *Proc. Robotics: Science and Systems (RSS'06)*, Philadelphia, USA, Aug. 2006.

[19] A. B. Chatfield, *Fundamentals of High Accuracy Inertial Navigation*, ser. Progress in Astronautics and Aeronautics, P. Zarchan, Ed. American Institute of Aeronautics and Astronautics, Sept. 1997, vol. 174.

[20] R. M. Haralick, H. Joo, C.-N. Lee, X. Zhuang, V. G. Vaidya, and M. B. Kim, "Pose Estimation from Corresponding Point Data," *IEEE Trans. Systems, Man and Cybernetics*, vol. 19, no. 6, pp. 1426–1446, Nov./Dec. 1989.

[21] Y. Ma, S. Soatto, J. Košecká, and S. Sastry, *An Invitation to 3-D Vision: From Images to Geometric Models*, 1st ed., ser. Interdisciplinary Applied Mathematics. Springer, Nov. 2004, vol. 26.

[22] S. J. Julier and J. K. Uhlmann, "Unscented Filtering and Nonlinear Estimation," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 401–422, Mar. 2004.

[23] S. J. Julier, "The Scaled Unscented Transform," in *Proc. IEEE American Control Conf. (ACC'02)*, vol. 6, Anchorage, USA, May 2002, pp. 4555–4559.

[24] G. H. Golub and C. F. V. Loan, *Matrix Computations*, 3rd ed., ser. Johns Hopkins Studies in Mathematical Sciences. The Johns Hopkins University Press, Oct. 1996.

[25] S. J. Julier and J. J. L. Jr., "On Kalman Filtering With Nonlinear Equality Constraints," *IEEE Trans. Signal Processing*, vol. 55, no. 6, pp. 2774–2784, June 2007.

[26] J. L. Crassidis and F. L. Markely, "Unscented Filtering for Spacecraft Attitude Estimation," in *Proc. AIAA Guidance, Navigation and Control Conf. (GN&C'03)*, no. AIAA-2003-5484, Austin, USA, Aug. 2003.

[27] B. K. P. Horn, "Closed-form solution of absolute orientation using unit quaternions," *J. the Optical Society of America A*, vol. 4, no. 4, pp. 629–642, Apr. 1987.

[28] L. Lucchese and S. K. Mitra, "Using Saddle Points for Subpixel Feature Detection in Camera Calibration Targets," in *Proc. Asia-Pacific Conf. Circuits and Systems (APCCAS'02)*, vol. 2, Singapore, Dec. 2002, pp. 191–195.

[29] D. G. Lowe, "Distinctive Image Features from Scale-Invariant Keypoints," *Int'l J. Computer Vision*, vol. 2, no. 60, pp. 91–110, Nov. 2004.