

# Tightly Coupled Visual-Inertial Navigation System Using Optical Flow

Simon Lynen Sammy Omari Matthias Wüest  
Markus Achtelik Roland Siegwart

*Autonomous Systems Lab, ETH Zürich, Switzerland.*  
first.last@mavt.ethz.ch

**Abstract:** This paper presents a state estimation framework that allows estimating the attitude, full metric speed and the orthogonal metric distance of an IMU-camera system with respect to a plane. The filter relies on optical flow- as well as gyroscope and accelerometer measurements. The underlying assumption is that the observed visual feature lies on a static plane. The orientation of the observed plane is not required to be known a-priori and is also estimated at run-time. The estimation framework fuses visual and inertial measurements in an Extended Kalman Filter (EKF). Experiments using a hand-held visual-inertial sensor successfully demonstrate the performance of the filter. It is shown that the state estimate is converging correctly, even in presence of substantial initial state errors. The minimal sensor suite, which is both light-weight and low-cost, renders the framework an appealing choice for the use as a navigation system on a wide range of robotic platforms, such as ground- or flying robots.

*Keywords:* Optical Flow, Control, State Estimation, Extended Kalman Filter, Tight Fusion

## 1. INTRODUCTION

In recent years, a considerable body of work was presented for position or velocity estimation of robots, in particular unmanned aerial vehicles (UAV), in GPS-denied environments using vision sensors (?),(2). Especially optical flow-based approaches have received a lot of attention since efficient real-time implementations are relatively straight forward and can therefore be run at high update rates on light-weight, low-power micro-controllers. Additionally, optical flow-based approaches are also able to handle scenes with only little contrast (? ) - especially compared to feature-based SLAM frameworks that require high-contrast, salient features that can be tracked over longer periods of time.

In (3), the authors use optical flow from a down-looking camera to control a UAV in hovering and vertical landing. They assume that the optical flow features lie on a horizontal plane whose normal can be estimated with the UAVs on-board IMU. Using this approach, they can compute the ratio  $\frac{v}{d}$  between the translational velocity  $v$  in the camera frame and orthogonal distance to the plane  $d$ . To resolve this ambiguity and relating the estimation to metric units, the authors of (? ) employ an ultrasound range sensor to measure the distance  $d$  to the plane. In (4), the authors estimate not only the ratio  $\frac{v}{d}$ , but also the normal vector of the plane using optical flow by incorporating the continuous homography constraint.

In contrast to these approaches ((4)), in (1), the implementation of a Unscented Kalman Filter (UKF)-based observer to estimate both the full metric velocity and dis-



Fig. 1. The sensor hardware used in the experiments: Synchronized data from the global shutter camera and the IMU is streamed to the onboard Intel Atom as well as to external devices via GigE/USB3. The presented experiments used only the data from one camera only.

tance to the plane as well as its normal vector is presented. The estimator makes no assumption on the orientation of the plane and does not rely on any distance or bearing sensors. Instead, only optical flow and inertial sensors are used as measurement input to the UKF. Using this setup, it was shown in simulation that the filter is able not only estimate the normal vector of the plane, but the full attitude of the IMU-camera system with respect to the plane.

In this paper, we describe the design and implementation of a real-time Extended Kalman Filter (EKF), extending the design of the UKF in (1). While the results in (1) were purely simulation based, we describe experimental results that were obtained by implementing this algorithm in a visual-inertial sensor setup using a highly efficient c++ implementation. We are able to show the performance of the filter in terms of robustness to outliers in the data-association in the optical-flow tracker as well as state initialization errors.

\* This work has been supported by the European Commission's Seventh Framework Programme (FP7) under grant agreements n.285417 (ICARUS), n.600958 (SHERPA) and n.266470 (myCopter).

The structure of the paper is as follows. In section II, the setup of the system is described. Section III discusses the required sensors and the corresponding stochastic models. In section IV, the estimation scheme, based on the Extended Kalman Filter is discussed. Evaluation of the estimation scheme and the discussion of the algorithm and its limitation are found in section V.

Concerning the mathematical notation, all vectors and matrices are written in bold text. We use a skew symmetric matrix notation for the vector cross product  $\mathbf{a} \times \mathbf{b} = [\mathbf{a}]_b \mathbf{b}$ . The canonical basis of  $\mathbb{R}^3$  is denoted  $\{\mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3\}$ . To describe rotations, we use quaternions of rotation in Hamilton notation.

## 2. SYSTEM SETUP

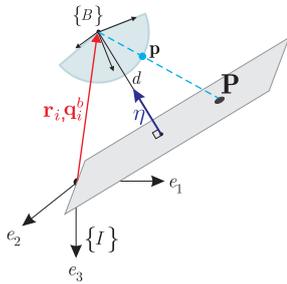


Fig. 2. Schematic representation of the system setup. The position and orientation (red) of the body frame  $\{B\}$  are described in the inertial coordinate frame  $\{I\}$ . The target plane passes through the origin of  $\{I\}$  and is described by its normal vector  $\boldsymbol{\eta}$ . The camera-IMU sensor-system in the body frame  $\{B\}$  is pointed in direction of the target plane observing at least one target point  $\mathbf{P}$ . The projection of  $\mathbf{P}$  on the spherical image plane of the camera is  $\mathbf{p}$ .

As depicted in Fig. 2 the system consists of an IMU (gyroscope and accelerometer) rigidly fixed to a spherical camera. The body frame is denoted  $\{B\}$  and is attached to the IMU-camera system. Its origin coincides with the accelerometer of the IMU. The orientation of the body frame with respect to the inertial frame  $\{I\}$  is defined by the unit quaternion  $\mathbf{q}_i^b$ . Equivalently, the same orientation mapping can also be expressed using the rotation matrix  $\mathbf{C}$  defined by the quaternion  $\mathbf{q}_i^b$ . The position and translational velocity of the body frame, expressed in the inertial frame, are denoted  $\mathbf{r}_i \in \mathbb{R}^3$  and  $\mathbf{v}_i \in \mathbb{R}^3$  respectively. The body angular rates are denoted  $\boldsymbol{\omega} \in \mathbb{R}^3$ .

The camera is observing a target plane which contains visual features. We define  $\mathbf{P} \in \mathbb{R}^3$  to be the coordinate of a visual feature on the target plane expressed in the camera frame. Its corresponding coordinates projected on the spherical image plane of the camera are  $\mathbf{p} \in \mathbb{S}^2$ . The target plane is defined by its normal  $\boldsymbol{\eta} \in \mathbb{S}^2$  expressed in the inertial frame.

## 3. MEASUREMENT MODELS

This section discusses the employed visual and inertial sensors and their corresponding stochastic measurement models largely following the derivations of (1).

### 3.1 Accelerometer Model

The accelerometer output  $\mathbf{a}_m \in \mathbb{R}^3$  in the body frame is composed of the gravity vector  $\mathbf{g} = g \mathbf{e}_3 \in \mathbb{R}^3$  in the inertial frame, the true linear accelerations  $\mathbf{a} \in \mathbb{R}^3$  in the inertial frame, a slowly time-varying bias  $\mathbf{b}_a \in \mathbb{R}^3$  in the body frame and zero-mean, white, Gaussian noise  $\mathbf{n}_a \in \mathbb{R}^3$ . The bias  $\mathbf{b}_a$  is modeled as a random walk process governed by zero-mean, white, Gaussian noise  $\mathbf{n}_{b_a} \in \mathbb{R}^3$ . We can write:

$$\mathbf{a} = \mathbf{C}^T (\mathbf{a}_m - \mathbf{b}_a - \mathbf{n}_a) + \mathbf{g} \quad (1)$$

$$\dot{\mathbf{b}}_a = \mathbf{n}_{b_a} \quad (2)$$

The noises  $\mathbf{n}_a$  and  $\mathbf{n}_{b_a}$  are specified by the corresponding covariance matrices  $\mathbf{Q}_a$  and  $\mathbf{Q}_{b_a}$  respectively.

### 3.2 Gyroscope Model

The gyroscopes on the IMU measures the angular velocity  $\boldsymbol{\omega}_m \in \mathbb{R}^3$  in the body frame which is composed of the true angular speed  $\boldsymbol{\omega} \in \mathbb{R}^3$ , slowly time-varying bias  $\mathbf{b}_\omega \in \mathbb{R}^3$  in the body frame and zero-mean, white, Gaussian noise  $\mathbf{n}_\omega \in \mathbb{R}^3$ . As in the case of accelerometers, each bias is modeled as a random walk process. We can write:

$$\boldsymbol{\omega} = \boldsymbol{\omega}_m - \mathbf{b}_\omega - \mathbf{n}_\omega \quad (3)$$

$$\dot{\mathbf{b}}_\omega = \mathbf{n}_{b_\omega} \quad (4)$$

As in the case of accelerometers, gyroscope noises are specified by their corresponding diagonal covariance matrices  $\mathbf{Q}_\omega$  and  $\mathbf{Q}_{b_\omega}$ .

### 3.3 Camera Model

Since we assume that the camera is intrinsically calibrated, we can map camera pixel coordinates  $\boldsymbol{\phi} \in \mathbb{R}^2$  to their corresponding position  $\mathbf{P} \in \mathbb{R}^3$  in the camera frame up to a scaling factor  $\lambda$

$$\lambda \mathbf{P} = \text{cam}(\boldsymbol{\phi}, \boldsymbol{\theta}_C) \quad (5)$$

using some generic camera model  $\text{cam}$ , and the intrinsic camera calibration parameter vector  $\boldsymbol{\theta}_C$ . Since we know  $\mathbf{P}$  only up to scale, we simply normalize it to obtain the feature vector  $\mathbf{p} \in \mathbb{S}^2$  on the spherical image plane with focal length  $f = 1$

$$\mathbf{p} = \frac{\mathbf{P}}{|\mathbf{P}|}. \quad (6)$$

From now on, we consider the camera to be a bearing vector sensor, i.e. it directly outputs feature vectors  $\mathbf{p}$ .

### 3.4 Optical Flow

Since the camera is fixed to the vehicle and the observed target point is considered to be stationary, the feature vector  $\mathbf{p}$  inherits the dynamics of the vehicle. The dynamics of the feature vector  $\mathbf{p}$  can be expressed as ((5)):

$$\dot{\mathbf{u}} = -[\boldsymbol{\omega}] \mathbf{p} - \pi_{\mathbf{p}} \frac{\mathbf{v}_b}{|\mathbf{P}|} \quad (7)$$

where  $\mathbf{v}_b$  is the translational velocity of the camera expressed in the body frame<sup>1</sup> and the projection operator

<sup>1</sup> In the actual hardware implementation the camera origin does not coincide with the body frame origin. There we replace  $\mathbf{v}_b$  by the camera velocity  $\mathbf{v}_c$  using the relation  $\mathbf{v}_c = \mathbf{v}_b + [\boldsymbol{\omega}] \mathbf{r}_b^c$  where  $\mathbf{r}_b^c$  is the baseline from the body frame- to the camera frame origin.

is defined as  $\pi_{\mathbf{p}} = (\mathbf{I}_3 - \mathbf{p}\mathbf{p}^T)$ . The vector  $\mathbf{u} \in \mathbb{R}^3$  is the optical flow corresponding to the feature vector  $\mathbf{p}$  and can be extracted from two consecutive images using a wide range of algorithms, such as Lucas-Kanade (6). We model

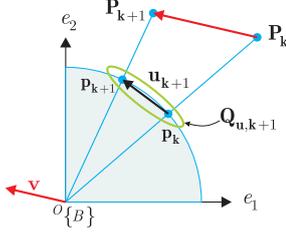


Fig. 3. Schematic representation of the optical flow computation as seen from the body frame  $\{B\}$ . The camera, moving with velocity  $\mathbf{v}$ , observes a visual feature  $\mathbf{P}$  whose projection on the spherical image plane is  $\mathbf{p}$ . The optical flow  $\mathbf{u}_{k+1}$  is calculated from the projected features at  $t_k$  and  $t_{k+1}$ . Its covariance ellipse is  $\mathbf{Q}_{\mathbf{u},k+1}$ .

the measured optical flow  $\mathbf{u}_m$  as being corrupted by zero-mean Gaussian noise  $\mathbf{n}_u \in \mathbb{R}^3$  specified by the covariance matrix  $\mathbf{Q}_u$ :

$$\mathbf{u} = \mathbf{u}_m - \mathbf{n}_u. \quad (8)$$

Since the camera output is not continuous, the optical flow vector  $\mathbf{u}_{k+1}$  at time  $t_{k+1}$  is approximated by  $\mathbf{u}_{k+1} \approx \frac{\mathbf{p}_{k+1} - \mathbf{p}_k}{t_{k+1} - t_k}$ , as depicted in Fig. 3. Since the feature vectors during both time steps are on the unit sphere, the noise of the optical flow vector is constrained to the surface of the sphere. Therefore, we approximate the covariance matrix  $\mathbf{Q}_{\mathbf{u},k+1}$  as a flat ellipsoid tangent to the unit sphere in direction of the average feature vector  $\frac{(\mathbf{p}_k + \mathbf{p}_{k+1})}{2}$ , as depicted in Fig. 3.

### 3.5 Target Plane Model

Since we assume that the target plane contains the origin of the inertial frame, it is sufficient to describe the plane by its normal vector  $\boldsymbol{\eta}$  expressed in the inertial frame. Since the normal vector is of unit length, it has two degrees of freedom. We can therefore parametrize the normal vector using two angles  $\alpha$  and  $\beta$  as:

$$\boldsymbol{\eta}(\alpha, \beta) = \begin{bmatrix} \sin(\alpha) \cos(\beta) \\ -\sin(\beta) \\ \cos(\alpha) \cos(\beta) \end{bmatrix} \quad (9)$$

For the sake of readability, we replace  $\boldsymbol{\eta}(\alpha, \beta)$  by  $\boldsymbol{\eta}$  from now on. We consider the angles to be slowly time-varying and model them as a random walk processes

$$\dot{\alpha} = n_\alpha \quad (10)$$

$$\dot{\beta} = n_\beta \quad (11)$$

with noises specified by their corresponding covariance terms  $Q_\alpha$  and  $Q_\beta$ ;

## 4. EXTENDED KALMAN FILTER

### 4.1 State Dynamics and State Vector Representation

Using the sensor models defined in (1) - (4) and the dynamics of the plane (10) - (11), the state dynamics can be written as (1):

$$\dot{\mathbf{r}}_i = \mathbf{v}_i \quad (12)$$

$$\dot{\mathbf{v}}_i = \mathbf{C}^T(\mathbf{a}_m - \mathbf{b}_a - \mathbf{n}_a) + \mathbf{g} \quad (13)$$

$$\dot{\mathbf{q}}_i^b = \frac{1}{2} \boldsymbol{\Omega}(\boldsymbol{\omega}_m - \mathbf{b}_\omega - \mathbf{n}_\omega) \mathbf{q}_i^b \quad (14)$$

$$\dot{\mathbf{b}}_\omega = \mathbf{n}_{b_\omega} \quad (15)$$

$$\dot{\mathbf{b}}_a = \mathbf{n}_{b_a} \quad (16)$$

$$\dot{\alpha} = n_\alpha \quad (17)$$

$$\dot{\beta} = n_\beta \quad (18)$$

The matrix  $\boldsymbol{\Omega}(\boldsymbol{\omega}) \in \mathbb{R}^{4 \times 4}$  relates the body angular rates to the corresponding quaternion rate by

$$\boldsymbol{\Omega}(\boldsymbol{\omega}) = \begin{bmatrix} 0 & -\boldsymbol{\omega}^T \\ \boldsymbol{\omega} & -[\boldsymbol{\omega}] \end{bmatrix}. \quad (19)$$

As a minimal state vector representation, we define the 18-element state vector

$$\mathbf{x} = [\mathbf{r}_i \ \mathbf{v}_i \ \mathbf{q}_i^b \ \mathbf{b}_\omega \ \mathbf{b}_a \ \alpha \ \beta]^T. \quad (20)$$

For brevity, we omit the discussion of the error dynamics and the derivation of the process noise covariance matrix and refer the reader to (1).

### 4.2 Measurement Equation

We now have to express the optical flow equation (7) using the estimated state vector  $\hat{\mathbf{x}}$ . The body angular rates  $\hat{\boldsymbol{\omega}}$  can be written as

$$\hat{\boldsymbol{\omega}} = \boldsymbol{\omega}_m - \hat{\mathbf{b}}_\omega. \quad (21)$$

The velocity in the body frame  $\hat{\mathbf{v}}_b$  can be written as

$$\hat{\mathbf{v}}_b = \hat{\mathbf{C}} \hat{\mathbf{v}}_i. \quad (22)$$

The computation of  $|\hat{\mathbf{P}}|$  is a bit more involved. By inspection of Fig. 1, we express the orthogonal distance of the camera to the target plane  $\hat{d}$  using the position of the camera in the inertial frame  $\hat{\mathbf{r}}_i$  and the normal vector of the plane  $\hat{\boldsymbol{\eta}}$  as

$$\hat{d} = |\hat{\mathbf{r}}_i^T \hat{\boldsymbol{\eta}}| \quad (23)$$

Equivalently, since we assume that the observed visual feature is lying on the plane, we can express  $\hat{d}$  as a function of the feature coordinate  $\hat{\mathbf{P}}$  in the body frame and the plane normal vector  $\hat{\boldsymbol{\eta}}$  in the world frame

$$\hat{d} = |\hat{\mathbf{P}}^T (\hat{\mathbf{C}} \hat{\boldsymbol{\eta}})| \quad (24)$$

Combining equation (6), (23) and (24), we can eliminate  $\hat{d}$  and solve for  $|\hat{\mathbf{P}}|$ :

$$|\hat{\mathbf{P}}| = \left| \frac{\hat{\mathbf{r}}_i^T \hat{\boldsymbol{\eta}}}{\mathbf{p}^T (\hat{\mathbf{C}} \hat{\boldsymbol{\eta}})} \right|. \quad (25)$$

Expanding equations (21), (22) and (25) into the optical flow equation (7), we can predict the measured optical flow as

$$\hat{\mathbf{u}} = -[(\boldsymbol{\omega}_m - \hat{\mathbf{b}}_\omega)] \mathbf{p} - \left| \frac{\mathbf{p}^T (\hat{\mathbf{C}} \hat{\boldsymbol{\eta}})}{\hat{\mathbf{r}}_i^T \hat{\boldsymbol{\eta}}} \right| \pi_{\mathbf{p}} (\hat{\mathbf{C}} \hat{\mathbf{v}}_i) \quad (26)$$

Similarly, one can find the following expression for the measured optical flow:

$$\mathbf{u}_m = -[(\boldsymbol{\omega}_m - \mathbf{b}_\omega - \mathbf{n}_\omega)] \mathbf{p} + \frac{\mathbf{p}^T (\mathbf{C}_{q_i} \boldsymbol{\eta})}{\mathbf{r}_i^T \boldsymbol{\eta}} \pi_{\mathbf{p}} (\mathbf{C}_{q_i} \mathbf{v}_i) + \mathbf{n}_u. \quad (27)$$

In the case of the UKF in (1), the Jacobians of the measurement equation are not required to be derived due to the use of Sigma points. Since the EKF uses a linearized measurement error model to compute the Kalman gain, the measurement equations have to be linearized w.r.t state. To this end, we define the error of the measurement prediction, or innovation, as

$$\delta \mathbf{u} = \mathbf{u}_m - \hat{\mathbf{u}} \quad (28)$$

which can be linearized to

$$\delta \mathbf{u} = \mathbf{H} \delta \mathbf{x}, \quad (29)$$

where  $\mathbf{H}$  is the observation matrix. Using additive error terms for all states except for the rotation where we employ the relation  $\delta q = q \otimes \hat{q} \approx [1 \ \frac{1}{2} \delta \theta^T]^T$  using the small angle approximation, with  $\theta$  denoting the minimal representation of the error in attitude. This leads to  $\mathbf{C}_{\hat{\mathbf{q}}_i} \approx (\mathbf{I}_3 - [\delta \theta]) \mathbf{C}_{\hat{\mathbf{q}}_i}$ .

We obtain the measurement Jacobians:

$$\mathbf{H} = [\mathbf{H}_r \ \mathbf{H}_v \ \mathbf{H}_\theta \ \mathbf{H}_{b_\omega} \ \mathbf{H}_{b_a} \ H_\alpha \ H_\beta], \quad (30)$$

where

$$\mathbf{H}_r = -\frac{\mathbf{p}^T (\mathbf{C}_{\hat{\mathbf{q}}_i} \hat{\boldsymbol{\eta}})}{\hat{\mathbf{r}}_i^T \hat{\boldsymbol{\eta}}} \boldsymbol{\pi}_p (\mathbf{C}_{\hat{\mathbf{q}}_i} \hat{\mathbf{v}}_i) \hat{\boldsymbol{\eta}}^T, \quad (31)$$

$$\mathbf{H}_v = \frac{\mathbf{p}^T (\mathbf{C}_{\hat{\mathbf{q}}_i} \hat{\boldsymbol{\eta}})}{\hat{\mathbf{r}}_i^T \hat{\boldsymbol{\eta}}} \boldsymbol{\pi}_p \mathbf{C}_{\hat{\mathbf{q}}_i}, \quad (32)$$

$$\mathbf{H}_\theta = \frac{1}{\hat{\mathbf{r}}_i^T \hat{\boldsymbol{\eta}}} [\boldsymbol{\pi}_p (\mathbf{C}_{\hat{\mathbf{q}}_i} \hat{\mathbf{v}}_i) \mathbf{p}^T [\mathbf{C}_{\hat{\mathbf{q}}_i} \hat{\boldsymbol{\eta}}] + \mathbf{p}^T (\mathbf{C}_{\hat{\mathbf{q}}_i} \hat{\boldsymbol{\eta}}) \boldsymbol{\pi}_p [(\mathbf{C}_{\hat{\mathbf{q}}_i} \hat{\mathbf{v}}_i)]], \quad (33)$$

$$\mathbf{H}_{b_\omega} = -[\mathbf{p}], \quad (34)$$

$$\mathbf{H}_{b_a} = \mathbf{O}_{3 \times 3}, \quad (35)$$

$$H_\alpha = -\frac{\hat{\mathbf{r}}_i^T (\mathbf{C}_{\hat{\mathbf{q}}_i}^T \mathbf{p})^T}{(\hat{\mathbf{r}}_i^T \hat{\boldsymbol{\eta}})^2} \frac{\partial [\hat{\boldsymbol{\eta}}]}{\partial \hat{\beta}} \boldsymbol{\pi}_p (\mathbf{C}_{\hat{\mathbf{q}}_i} \hat{\mathbf{v}}_i) \cos(\hat{\beta}), \quad (36)$$

$$H_\beta = \frac{\hat{\mathbf{r}}_i^T (\mathbf{C}_{\hat{\mathbf{q}}_i}^T \mathbf{p})^T}{(\hat{\mathbf{r}}_i^T \hat{\boldsymbol{\eta}})^2} \begin{bmatrix} 0 & s & 0 \\ -s & 0 & -c \\ 0 & c & 0 \end{bmatrix} \boldsymbol{\pi}_p (\mathbf{C}_{\hat{\mathbf{q}}_i} \hat{\mathbf{v}}_i) \quad (37)$$

using  $s = \sin(\hat{\alpha})$  and  $c = \cos(\hat{\alpha})$ .

#### 4.3 Measurement Covariance Matrix

In Eq. (27) the true de-rotated optical flow measurement  $\mathbf{u}_m$  is perturbed by the pixel noise of the optical flow extraction  $\mathbf{n}_u$  and the gyroscope noise  $\mathbf{n}_\omega$  introduced via the de-rotation term  $[\boldsymbol{\omega}] \mathbf{p}$ . According to (1), the covariance matrix of the de-rotated optical flow is the sum of the gyroscope covariance  $\mathbf{Q}_\omega$  and the covariance of the optical flow algorithm  $\mathbf{Q}_u$ . If multiple features are extracted, the measurement covariance matrix is build by stacking the covariance matrices of the de-rotated optical flow vectors diagonally as

$$\mathbf{R} = \begin{bmatrix} \mathbf{Q}_{u,1} + \mathbf{Q}_\omega & \dots & \mathbf{O}_{3 \times 3} \\ \vdots & \ddots & \vdots \\ \mathbf{O}_{3 \times 3} & \dots & \mathbf{Q}_{u,N} + \mathbf{Q}_\omega \end{bmatrix}. \quad (38)$$

#### 4.4 Outlier Rejection

Before using the optical flow measurements in the update step of the EKF, we want to robustly detect and reject

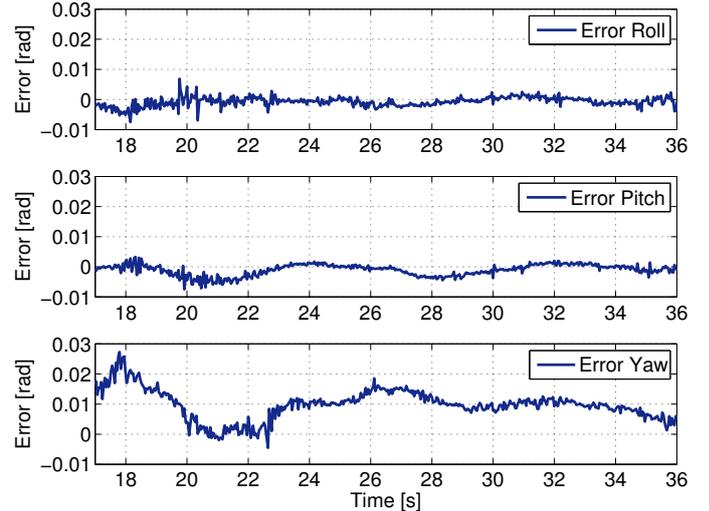


Fig. 4. The observability properties of the system are also reflected in the experimental results: The error in the estimation of observable modes of the system (roll and pitch) are significantly smaller than the errors in the unobservable mode (yaw). This result highlights the usability of the proposed approach for UAS attitude control.

outlier measurements. We follow the approach in (9) and employ a Mahalanobis-distance test to detect outlier measurements. We compute the Mahalanobis distance as

$$\chi^2 = (\mathbf{u}_m - \hat{\mathbf{u}})^T \mathbf{S}^{-1} (\mathbf{u}_m - \hat{\mathbf{u}}), \quad (39)$$

The matrix  $\mathbf{S}$  is the innovation covariance of the EKF as defined by eq. (77) in (8) and can be seen as a weighting term depending on the confidence in the current prediction. Measurements that exceed a certain threshold  $\chi_{th}^2$  are rejected and are not employed in the update step of the estimation scheme.

As in (1), to further increase robustness, each measurement is checked for its signal-to-noise ratio. Thus, each vector with a signal-to-noise ratio below a certain threshold  $\Psi_{th}^2$  was rejected as well:

$$\mathbf{u}_m^T (\mathbf{Q}_u + \mathbf{Q}_\omega)^{-1} \mathbf{u}_m < \Psi_{th}^2. \quad (40)$$

#### 4.5 Analysis of Extended Kalman Filter

The observability analysis, as conducted in (1), reveals that the position in direction of the normal of the plane is fully observable while the position in direction along the plane is not observable. The remaining states, namely the velocity and attitude with respect to the plane as well as the accelerometer and gyroscope biases in the body frame are also observable. However, these states are only observable under the condition that (1) the system is excited by accelerations on at least two independent axis and (2) there is at least one optical flow measurement with its feature vector  $\mathbf{p}$  not parallel to the velocity vector in the body frame. The requirement for the acceleration can be explained using an intuitive argument: If the system observes a constant optical flow, the observer cannot distinguish the ratio between velocity and distance.

However, while the attitude is observable with respect to the plane frame, this does not imply that it is fully observ-

able in the inertial frame. In fact, the rotation around the gravity vector in the inertial frame is not observable. This is due to the fact that the system has no measurement of an absolute, inertial yaw angle. Therefore, the yaw angle of the system in the inertial frame is not observable. The roll- and pitch angles of the attitude in the inertial frame are observable since the gravity vector acts as reference. If a virtual measurement of the yaw-angle is introduced as an additional measurement to the analysis, as suggested in (11), the unobservable manifold of the attitude in the inertial frame becomes observable. However, the use of an inertial heading reference vector (e.g. from a magnetometer) is not necessary if no globally consistent yaw angle is required. For example, in a wall-following scenario, the observer provides the relative attitude with respect to the wall, which is enough information for yaw-stabilization.

## 5. IMPLEMENTATION AND EVALUATION

### 5.1 Implementation Optical Flow Algorithm

Using a custom-made optical flow algorithm, salient features are detected by applying a FAST corner detector in each frame. For every detected corner, a cornerness function is applied to determine whether the corner is significant. Feature correspondences between consecutive frames are found using a patch-based correlation method. The candidate pixel with the patch with the lowest sum of absolute differences (SAD) then is chosen to be the position of the feature in the second frame. The output of the optical flow algorithm is depicted in Fig.5.

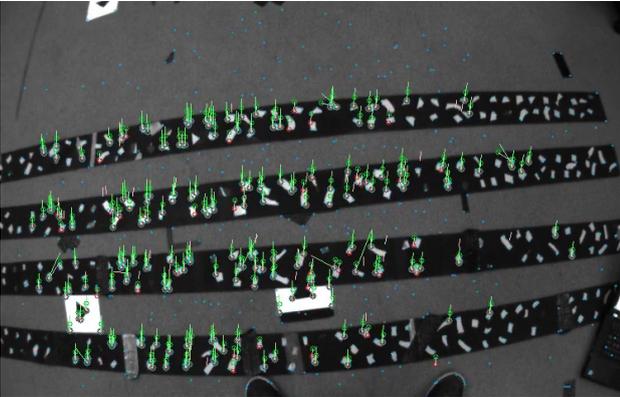


Fig. 5. Output of the optical flow algorithm taking 1ms per frame on a single core Intel i7-2720QM.

### 5.2 Implementation Extended Kalman Filter

The open-source implementation of the multi-sensor Extended Kalman Filter (12) was employed in this work. To this end, the state vector was extended to include the wall parameters. Additionally, the versatile structure of the filter enabled the straight-forward implementation of the measurement equations as well as their corresponding Jacobians.

### 5.3 Results

To validate the implemented EKF and to evaluate its performance, the filter was tested on data that was recorded

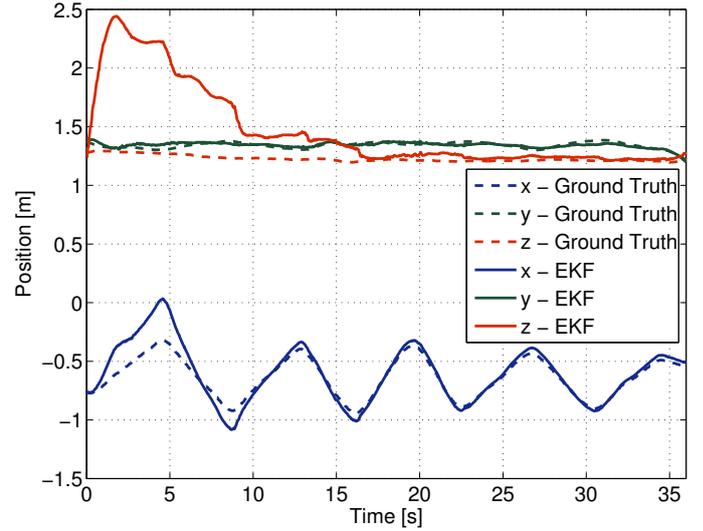


Fig. 6. Position estimate

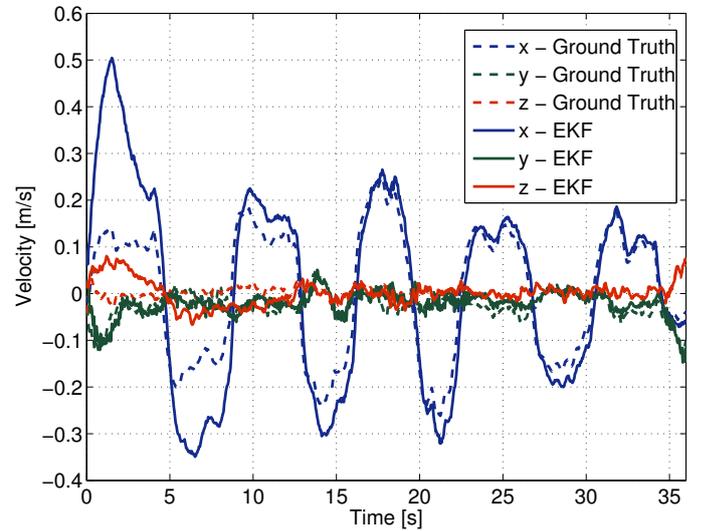


Fig. 7. Velocity estimate.

using the ASL visual-inertial SLAM sensor, with synchronized camera and IMU. The pose of the sensor was additionally tracked using a Vicon motion tracking system at 200 Hz. By fusing the pose estimate of a Vicon motion capture system in a EKF (12) using the same IMU data as used for the optical flow EKF, we obtain ground truth data for position, velocity, attitude and the IMU biases.

The experiment was performed using a down looking camera at an altitude of approximately 1.3m and performing a see-saw motion in x-direction parallel to the ground plane. For this experiment, the estimation of the planes normal vector was disabled. Instead it was assumed that it is horizontal, i.e. its normal vector is parallel to gravity.

The figures 6, 7, 8 and 9 show the evolution of the filter estimates of the position, velocity, attitude and the acceleration biases. The gyroscope biases converge similarly and are omitted for space reasons.

All the observable states of the EKF have converged after 17s and remain close to the ground truth. The filter estimate of the z position, which corresponds to the

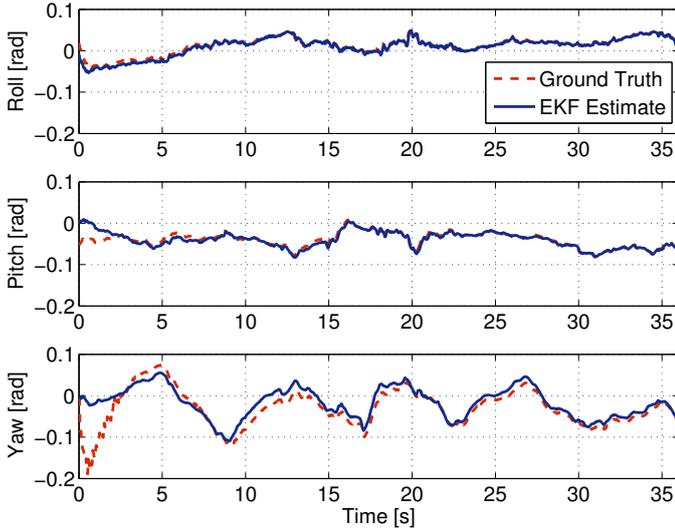


Fig. 8. Attitude estimate.

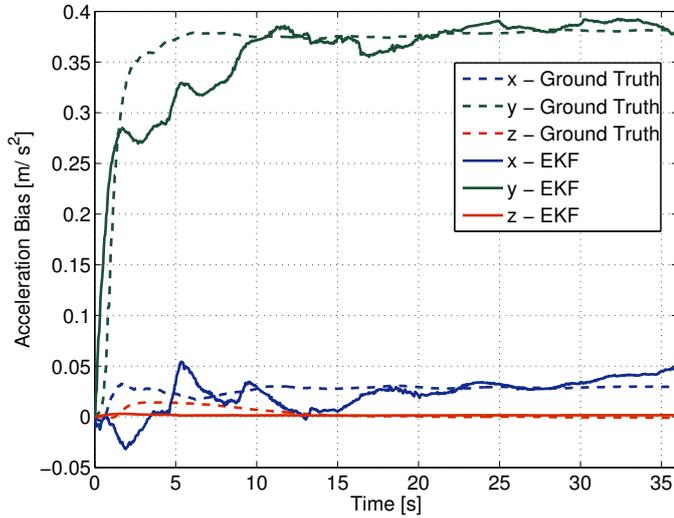


Fig. 9. Accelerometer bias estimate. distance to the ground, follows the ground truth after convergence with an RMS error of 4.6 cm. At an altitude of around 1.3 m, this is a relative error of less than 4%. The positions along the x and y axes are not observable. Nevertheless, the drift of the estimates is low. The RMS errors of the velocity estimates in x, y and z directions are 0.030, 0.016 and 0.006 m/s respectively. Apart from the unobservable rotation around the direction of gravity the filter is able to accurately (See 4) estimate the attitude of the IMU which is crucial for UAS stabilisation.

A video of the experiment can be found at <http://youtu.be/ieFseoBF1OU>.

## 6. CONCLUSIONS

In this paper, we presented an implementation of a state estimation framework that allows estimating the attitude, full metric velocity and the orthogonal metric distance of an IMU-camera system w.r.t. a plane. The filter relies on optical flow- as well as gyroscope and accelerometer measurements and does not require any range sensors. Experiments in simulation successfully demonstrate the performance of the filter as well as highlight the relevance

of the conclusions given by the observability analysis. The experimental results successfully show convergence of the filter in presence of substantial initial state offsets and sensor noises present during the flight of a UAV.

This generic estimation framework can be used as a vision-aided navigation system in a wide range of applications, on a variety of robotic platforms. Due to the minimal sensor configuration, which is both light-weight and low-cost, this framework is an appealing choice to be used on weight-sensitive robotic platforms such as UAVs.

## REFERENCES

- [1] S. Omari, G. Ducard, "Metric Visual-qInertial Navigation System Using Single Optical Flow Features", Proc. of the 2013 European Control Conference (ECC), Zurich, 2013.
- [2] M. Achtelik, A. Bachrach, R. He, S. Prentice, N. Roy, "Stereo vision and laser odometry for autonomous helicopters in GPS-denied indoor environments," Unmanned Systems Technology XI., 2009, Orlando.
- [3] B. Herisse, F-X. Russotto, T. Hamel, R. Mahony, "Hovering flight and vertical landing control of a VTOL Unmanned Aerial Vehicle using Optical Flow," Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2008, Nice, France.
- [4] V. Grabe, H.H. Buelthoff, P.R. Giordano, "On-board Velocity Estimation and Closed-loop Control of a Quadrotor UAV based on Optical Flow," Proc. of the IEEE/RSJ Int. Conf. on Robotics and Automation (ICRA), 2012, Saint-Paul, MN.
- [5] R. Mahony, P. Corke, T. Hamel, "Dynamic Image-Based Visual Servo Control Using Centroid and Optic Flow Features," J. Dyn. Sys., Meas., Control, Volume 130, Issue 1, 2008.
- [6] B.D Lucas, T. Kanade. "An iterative image registration technique with an application to stereo vision," Proc. of the 7th international joint conference on Artificial intelligence. 1981.
- [7] B. K. Horn, B.G. Schunck, "Determining optical flow," Artificial intelligence 17, No. 1, pp. 185-203, 1981.
- [8] J.L. Crassidis, "Sigma-point Kalman filtering for integrated GPS and inertial navigation," IEEE Transactions on Aerospace and Electronic Systems, 42.2, pp. 750-756, 2006.
- [9] F.M. Mirazei, S.I. Roumeliotis, "A Kalman filter-based algorithm for IMU-camera calibration," Proc. of the IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS), 2007, San Diego, CA.
- [10] R. Hermann, A. Krener, "Nonlinear controllability and observability," IEEE Trans. Autom. Control, vol. AC-22, No. 5, pp 728-740, 1977.
- [11] S. M. Weiss, "Vision Based Navigation for Micro Helicopters," Ph.D. dissertation, ETH Zurich, 2012.
- [12] S. Lynen, M.W. Achtelik, S. Weiss, M. Chli, R. Siegwart, "A Robust and Modular Multi-Sensor Fusion Approach Applied to MAV Navigation," Proc. of the IEEE/RSJ Conference on Intelligent Robots and Systems (IROS), 2013