High-frequency MAV state estimation using low-cost inertial and optical flow measurement units

Angel Santamaria-Navarro, Joan Solà and Juan Andrade-Cetto

***Abstract*— This paper develops a simple and low-cost method for 3D, high-rate vehicle state estimation, specially designed for free-flying Micro Aerial Vehicles (MAVs). We fuse observations from inertial measurement units and the recently appeared low-cost optical flow smart cameras. These smart cameras integrate a sonar altimeter, a triaxial gyrometer and an optical flow sensor, and directly provide metric ego-motion information in the form of body velocities and altitude. Compared to state-of-the-art visual-inertial odometry methods, we are able to drastically reduce the computational load in the main processor unit, and obtain an accurate estimation of the vehicle state at a high update rate of 100Hz. We thus extend the current use of these smart cameras from hovering purposes to odometry estimation. In order to propose a simple algorithmic solution, we investigate the performances of two Kalman filters, in the extended and error-state flavors, alongside a large number of algorithm variations, using simulations and real experiments with precise ground-truth. We observe that the marginal performance gain attained with these algorithm improvements does not pay for the effort of implementing them. We conclude that a classical EKF in its simplest form is sufficient for providing motion estimates that coherently exploit the available measurements.**

I. Introduction

Micro Aerial Vehicles (MAVs), and in particular quadrotor systems, have substantially gained popularity in the research community in recent years, motivated by their significant increase in maneuverability, together with a decrease in weight and cost. In order to perform high level tasks with MAVs, such as autonomous exploration or scene interaction, robust state estimation at a high update rate is crucial for robot navigation and agile flight control.

Low-cost, low complexity systems for rich and high-dynamics MAV state estimation are rare. A first family of localization systems for MAVs relies on external infrastructure (GPS, RF beacons, visual tags, IR cameras), *e.g.* [1], which may not always be practical. When using low-cost setups, these systems usually lack precision, dynamics, or both. A second family of methods embark all hardware and software for self-localization, thus not relying on any external setup. These include any kind of SLAM or odometry system drawn from other robotics fields [2], [3].

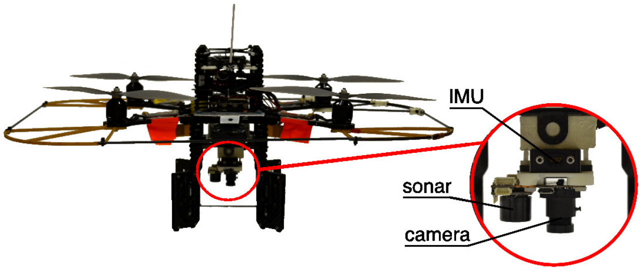

Fig. 1. Quadrotor used in the experiments with the IMU and optical flow measurement units attached below.

Combinations of IMU and visual sensors (especially in monocular mode) are becoming very popular, thanks to their low weight, power consumption and price. However, many of these systems are computationally expensive and require last-generation processors and complex algorithms that rely on a vast diversity of techniques that are often difficult to master. Their typical high precision outcome is attained by jointly estimating a subset of past camera poses and a number of landmarks in the environment, which are tracked in the image over relatively long periods. Key examples are [4], [5] using EKF, [6] using the multi-state constraint KF, [7] using bundle adjustment, or [8] using graphical models.

In order to reduce the computational burden and increase the update rate, several authors opt to exploit the image information only locally and in 2D. Ref. [9] proposes a speed-estimation module, which converts the pair camera-IMU into a metric body-speed sensor at 40Hz update rate, using optical flow information of at least 2 image features, within an EKF framework. In [10], optical flow information is fused with inertial measurements, however results are only shown in simulations. A similar approach is [11].

The solution we propose (Fig. 1) belongs to this last group of approaches. It uses exclusively low-cost sensors and low-complexity algorithms: As hardware, we use a MicroStrain IMU [12] and the recently appeared PX4 smart camera [13]. The PX4 integrates a low-cost sonar altimeter, a triaxial gyrometer and an optical flow sensor, and processes the data to deliver metric measurements of altitude and velocity at 100Hz. As software, we have developed two Kalman filters, in the extended (EKF) and error-state (ESKF) forms, for the sake of performance comparison. All the image processing for the optical flow is computed by the PX4. The overall system acts as an odometer that provides absolute altitude, velocity, orientation, angular rate, and acceleration, with respect to a precise gravity reference, at 100Hz data rate. XY position and yaw orientation are not observable, and

This work has been partially funded by the EU project ARCAS FP7-287617. and by the Spanish Ministry of Economy and Competitiveness projects PAU+ DPI-2011-27510 and ROBINSTRUCT TIN2014-58178-R.

The authors are with the Institut de Robòtica i Informàtica Industrial, CSIC-UPC, Llorens i Artigas 4-6, Barcelona 08028, Spain, asantamaria, jsola, cetto@iri.upc.edu.

The paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors. The material includes a video of the state estimation presented in the paper.

their output is the result of dead-reckoning, subject to drift. These modes can be observed by a higher level task, such as a visual servoing ([14], [15]), with a lower update rate.

To the knowledge of the authors, this work is the first example of the usage of such a low-cost visual-range-inertial sensor setup for 6DoF motion estimation, using also low complexity algorithms without sophisticated features. This sensor setup has the advantage of being simple, lightweight, low power and low cost, and is already included, with minor variations, in several commercial multi-copters as their basic instrumentation, being typically used as a means for automatic hovering. As such, a setup like this is already available to a huge amount of teams worldwide, both in research and industry. Many of these teams may not have the energy, the time or the interest of implementing complex, CPU-demanding motion estimation algorithms, yet they are not satisfied with a pure hovering solution.

When compared to other visual-intertial approaches, the optical-flow nature of the PX4, with a very narrow field of view of only 64×64 pixels or $1.6°$, represents an important limitation, in the sense that visual features are only exploited locally, and there is no notion of features being tracked over long periods of time. The number and length of feature tracks are key to the high precision attained by the aforementioned sophisticated methods, and in consequence, we cannot expect performances of our setup comparable to other visual-odometry systems. But importantly, the number and length of these feature tracks are responsible for their algorithmic complexity and CPU consumption.

Fortunately for us, the high filter update rate made possible by the PX4 contributes to increase the validity of linearizations, minimizing the accumulation of error and thus enabling simpler methods for an equivalent performance. In this scenario, the main aim of this paper is to show that, given a sensor setup with such capabilities, we are able to derive motion estimates that are useful in the mid term (a few minutes, *i.e.*, from a significant fraction to a full battery recharge), without the need to implement complicated algorithms.

Indeed, we explore a large number of algorithm improvements and variations defended in the literature, and show that their impact on system performance is marginal. These variations are shown in Tab. I, and are properly defined later in the text. They are summarized hereafter. First, we implement extended (EKF) and error-state (ESKF) Kalman filters [16]. Second, we express the orientation errors of ESKF both in local (LE) and global (GE) frames [17]. Note that in EKF the orientation error is additive and this distinction is irrelevant. Third, we compare different schemes for rotational motion integration of the quaternion [18], including forward zeroth-order (Q0F), backward zeroth-order (Q0B), and first-order (Q1). Fourth, we compare different integration forms and approximations of the system's transition matrix $(\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3)$ [5].

Other contributions of the paper follow. First, the observation model of the PX4 sensor, and its analytical derivation to get, together with the IMU, all the observable modes. Also, we propose algorithmic solutions to correct some limitations of the PX4 that make it produce, under particular circumstances, aberrant measurements.

TABLE I

KALMAN FILTERS AND ALGORITHM VARIATIONS

Filter type	Quat. error	Quat. integration	Trans. Mat. Trunc.
EKF	–	Q0F, Q0B, Q1	$\mathbf{F}_2$
ESKF	GE, LE	Q0F, Q0B, Q1	$\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3$

The remainder of this article is structured as follows. In the next section we provide a brief description of the system setup, with its sensor models and state parametrizations. The system kinematic equations are described in Sec. III, together with a brief note on observability. Sec. IV shows the EKF and ESKF filters. Simulation validations come in V, and real robot experiment results in Sec. VI. Finally, conclusions are given in Sec. VII.

II. System Setup

We consider a quadrotor equipped with a MicroStrain 3dm-gx3-25 IMU and a PX4 optical flow sensor, moving with respect to a global coordinate frame, assumed inertial. Our goal is to track the vehicle state, *i.e.*, the position, orientation and velocities of the platform, using the inertial measurements, and to correct these estimates with the PX4 readings, observing in turn the IMU biases for their proper compensation. The PX4 integrates a monocular camera looking downwards, an ultrasound range sensor in the same direction, and a triaxial gyrometer. It processes these data and delivers horizontal velocities $\mathbf{v}_{of}(x, y)$ in local (*i.e.*, body) frame, plus altitude $\mathbf{p}_{of}(z)$ with respect to a flat, horizontal ground. The PX4 can also provide the raw optical flow data, which can be used in critical situations to recover from aberrant PX4 readings, as we explain in section VI-A. In order for the PX4 to perform correctly, scene lighting must be adequate, and the ground is required to have good visual texture and sufficient ultrasound reflectivity.

The IMU and the PX4 optical flow sensors are rigidly attached together as shown in Fig. 1. Their respective frames are aligned with each other and with the frame of the MAV via careful mechanics, without further calibration.

A. State vectors: notation and conventions

We are interested in estimating a *true* state $\mathbf{x}_t$ defined by

$$\mathbf{x}_t = \begin{bmatrix} \mathbf{p}_t & \mathbf{v}_t & \mathbf{q}_t & \mathbf{a}_{bt} & \boldsymbol{\omega}_{bt} \end{bmatrix}^\top , \qquad (1)$$

where $\{\mathbf{p}, \mathbf{v}, \mathbf{q}\}$ are position, velocity, and orientation quaternion[1], all expressed in inertial frame, and $\{\mathbf{a}_b, \boldsymbol{\omega}_b\}$ are accelerometer and gyrometer biases. In EKF, we estimate $\mathbf{x}_t$ directly. In ESKF, we speak of true-, nominal- and error-state values, the true-state $\mathbf{x}_t$ being expressed as a

[1] We use the Hamilton convention for quaternions, which defines $ij = -ji = k$. This is the most widely used convention (*e.g.* Eigen, Ceres, ROS, and most literature), but it is in contrast with quite a few remarkable works on visual-inertial odometry (*e.g.* [5], [6], [18]) using the JPL convention, with $ji = -ij = k$. Because of the sign change, the Hamilton quaternion is right-handed, while the JPL is left-handed [18].

suitable composition (linear sum or quaternion product) of the *nominal*-state $\mathbf{x}$, which is only propagated in time, and the *error*-state $\delta\mathbf{x}$, which is estimated by the filter. They are related as follows,

$$\mathbf{x}_t = \mathbf{x} \oplus \delta\mathbf{x} \tag{2}$$

$$\mathbf{x} = \begin{bmatrix}\mathbf{p} \; \mathbf{v} \; \mathbf{q} \; \mathbf{a}_b \; \boldsymbol{\omega}_b\end{bmatrix}^\top \tag{3}$$

$$\delta\mathbf{x} = \begin{bmatrix}\delta\mathbf{p} \; \delta\mathbf{v} \; \delta\boldsymbol{\theta} \; \delta\mathbf{a}_b \; \delta\boldsymbol{\omega}_b\end{bmatrix}^\top \ . \tag{4}$$

where $\oplus$ denotes a generic composition. This composition is the trivial addition for all state variables (*e.g.*, $\mathbf{p}_t = \mathbf{p} + \delta\mathbf{p}$) except for the orientation. We use a minimal orientation error $\delta\boldsymbol{\theta} \in \mathbb{R}^3$, which may be defined in the global frame (GE) or in the local frame (LE), *i.e.*, its composition is computed, respectively, with a product *on the left or right hand sides* of the nominal quaternion,

$$\text{global error (GE):} \quad \mathbf{q}_t = \delta\mathbf{q} \otimes \mathbf{q} \ , \tag{5a}$$

$$\text{local error (LE):} \quad \mathbf{q}_t = \mathbf{q} \otimes \delta\mathbf{q} \ , \tag{5b}$$

where $\delta\mathbf{q} = \mathbf{q}\{\delta\boldsymbol{\theta}\}$ is the orientation error expressed in quaternion space,

$$\delta\mathbf{q} = \begin{bmatrix}\cos(||\delta\boldsymbol{\theta}||/2) \\ \frac{\delta\boldsymbol{\theta}}{||\delta\boldsymbol{\theta}||}\sin(||\delta\boldsymbol{\theta}||/2)\end{bmatrix} \approx \begin{bmatrix}1 \\ \delta\boldsymbol{\theta}/2\end{bmatrix} \ . \tag{6}$$

Finally, we note $\mathbf{R}\{x\}$ and $\mathbf{q}\{x\}$ the rotation matrix and quaternion equivalents to the generic orientation x. Likewise, we write $\mathbf{R} \triangleq \mathbf{R}\{\mathbf{q}\}$ and $\mathbf{R}_t \triangleq \mathbf{R}\{\mathbf{q}_t\}$.

B. Sensor Models

The IMU is composed of a 3-axis accelerometer and a 3-axis gyrometer. The accelerometer measures acceleration $\mathbf{a}_t$ and gravity $\mathbf{g}$ together, expressed in body frame. This measurement is affected by additive bias $\mathbf{a}_{bt}$ and noise $\mathbf{a}_n$. The gyrometer measures angular rates in body frame, likewise affected by additive bias and noise. The sensed values are thus,

$$\mathbf{a}_S = \mathbf{R}_t^\top(\mathbf{a}_t - \mathbf{g}) + \mathbf{a}_{bt} + \mathbf{a}_n \tag{7a}$$

$$\boldsymbol{\omega}_S = \boldsymbol{\omega}_t + \boldsymbol{\omega}_{bt} + \boldsymbol{\omega}_n \ . \tag{7b}$$

The PX4 optical flow sensor outputs the altitude from the ground, and a 2D linear velocity vector in the body XY plane, all affected by additive Gaussian noise. This can be expressed in compact form as

$$\begin{bmatrix}\mathbf{p}_{of}(z) \\ \mathbf{v}_{of}(x) \\ \mathbf{v}_{of}(y)\end{bmatrix} = \mathbf{S}\begin{bmatrix}\mathbf{p}_t \\ \mathbf{R}_t^\top \mathbf{v}_t\end{bmatrix} + \mathbf{n}_{of} \ , \tag{8}$$

where $\mathbf{S} = \begin{bmatrix}\mathbf{0}_{3\times 2} & \mathbf{I}_{3\times 3} & \mathbf{0}_{3\times 1}\end{bmatrix}$ is a 3×6 matrix that selects the respective rows. The measurement noise $\mathbf{n}_{of}$ is assumed to be Gaussian with zero mean and covariances matrix $\mathbf{N}_{of}$. The planar velocity measurement $\mathbf{v}_{of}$ is computed directly by the PX4 from its own optical flow $\boldsymbol{\Phi}_{of}$, angular rates $\boldsymbol{\omega}_{of}$, and altitude h_{of} measurements, according to

$$\mathbf{v}_{of} = \frac{-\boldsymbol{\Phi}_{of} \cdot h_{of}}{\triangle t \cdot f} + \mathbf{S}'\left(\boldsymbol{\omega}_{of} \times \begin{bmatrix}0 & 0 & h_{of}\end{bmatrix}^\top\right) \ , \tag{9}$$

where f is the focal length of the PX4's camera, expressed in pixels, and $\mathbf{S}' = [\ \mathbf{I}_{2x2} \ \ \mathbf{0}_{2x1}]$ is also a selection matrix.

III. System kinematics for EKF and ESKF

A. Continuous time

As it is common practice in the literature of IMU navigation (*e.g.* [4]), we express the true dynamics of our system with the following set of equations,

$$\dot{\mathbf{p}}_t = \mathbf{v}_t \tag{10a}$$

$$\dot{\mathbf{v}}_t = \mathbf{R}_t(\mathbf{a}_S - \mathbf{a}_{bt} - \mathbf{a}_n) + \mathbf{g} \tag{10b}$$

$$\dot{\mathbf{q}}_t = \frac{1}{2}\mathbf{q}_t \otimes \begin{bmatrix}0 & \boldsymbol{\omega}_S - \boldsymbol{\omega}_{bt} - \boldsymbol{\omega}_n\end{bmatrix}^\top \tag{10c}$$

$$\dot{\mathbf{a}}_{bt} = \mathbf{a}_w \tag{10d}$$

$$\dot{\boldsymbol{\omega}}_{bt} = \boldsymbol{\omega}_w \ , \tag{10e}$$

where true acceleration $\mathbf{a}_t$ and angular rates $\boldsymbol{\omega}_t$ have been substituted by those isolated from the IMU model (7). The biases have been considered non-constant, and in such case two small-energy random walks with white Gaussian generator noises $\mathbf{a}_w$ and $\boldsymbol{\omega}_w$ are used as dynamic models.

The system above, after integration to discrete time (see below), is estimated by an EKF. For the ESKF, we isolate the nominal-state kinematics by simply removing all error, noise and perturbation terms. The error-state kinematics are obtained by considering the composition of each system equation, solving for the error-state, rearranging some cross-products and simplifying all second order terms, yielding,

$$\dot{\delta\mathbf{p}} = \delta\mathbf{v} \tag{11a}$$

$$\dot{\delta\mathbf{v}} = -\lfloor\mathbf{R}(\mathbf{a}_S - \mathbf{a}_b)\rfloor_\times\delta\boldsymbol{\theta} - \mathbf{R}\delta\mathbf{a}_b - \mathbf{R}\mathbf{a}_n \tag{11b}$$

$$\dot{\delta\boldsymbol{\theta}} = -\mathbf{R}\,\delta\boldsymbol{\omega}_b \ - \ \mathbf{R}\boldsymbol{\omega}_n \tag{11c}$$

$$\dot{\delta\mathbf{a}}_b = \mathbf{a}_\omega \tag{11d}$$

$$\dot{\delta\boldsymbol{\omega}}_b = \boldsymbol{\omega}_\omega \ , \tag{11e}$$

which, remarkably, is a time-variant linear system. This system is valid for the globally-defined angular error, (GE, 5a). For a local definition, (LE, 5b), we just need to substitute equations (11b,11c) with,

$$\dot{\delta\mathbf{v}} = -\mathbf{R}\lfloor\mathbf{a}_S - \mathbf{a}_b\rfloor_\times\delta\boldsymbol{\theta} - \mathbf{R}\delta\mathbf{a}_b - \mathbf{R}\mathbf{a}_n \tag{11f}$$

$$\dot{\delta\boldsymbol{\theta}} = -\lfloor\boldsymbol{\omega}_S - \boldsymbol{\omega}_b\rfloor_\times\delta\boldsymbol{\theta} - \delta\boldsymbol{\omega}_b - \boldsymbol{\omega}_n \ . \tag{11g}$$

B. Computing the integral to discrete time

The integral of continuous differential equations of the type $\dot{\mathbf{x}} = f(\mathbf{x}, \mathbf{u})$ from times $(k\text{–}1)\Delta t$ to $k\Delta t$ can be approximated in a number of ways. We want to evaluate the impact of these approximations on final performance, and thus we perform these integrals with different degrees of accuracy. For all the variables except the quaternion, we integrate the linearized system, $\dot{\mathbf{x}} = \mathbf{A}\,\mathbf{x} + \mathbf{B}\,\mathbf{u}$, with $\mathbf{A} = \partial f/\partial\mathbf{x}$, $\mathbf{B} = \partial f/\partial\mathbf{u}$, into the discrete-time $\mathbf{x}_k \approx \mathbf{F}\,\mathbf{x}_{k-1} + \mathbf{B}\,\mathbf{u}\Delta t$, and we truncate the Taylor series expressing the transition matrix $\mathbf{F} = e^{\mathbf{A}\Delta t}$ at different orders, $N \in \{1, 2, 3\}$, obtaining the different approximations $\mathbf{F}_N$ in Tab. I,

$$\mathbf{F}_N \triangleq \sum_{n=0}^{N<\infty}\frac{1}{n!}\mathbf{A}^n\Delta t^n \ . \tag{12}$$

For the quaternion, we contemplate the Q0F, Q0B, and Q1 integrators [18] of the differential equation (10c),

$$\text{Q0F:}\quad \mathbf{q}_k \approx \mathbf{q}_{k-1} \otimes \mathbf{q}\{\boldsymbol{\omega}_{k-1}\Delta t\} \tag{13a}$$

$$\text{Q0B:}\quad \mathbf{q}_k \approx \mathbf{q}_{k-1} \otimes \mathbf{q}\{\boldsymbol{\omega}_{k}\Delta t\} \tag{13b}$$

$$\text{Q1:}\quad \mathbf{q}_k \approx \mathbf{q}_{k-1} \otimes \left(\mathbf{q}\{\bar{\boldsymbol{\omega}}\Delta t\} + \frac{\Delta t^2}{24}\begin{bmatrix} 0 \\ \boldsymbol{\omega}_{k-1} \times \boldsymbol{\omega}_k \end{bmatrix}\right) \tag{13c}$$

where $\boldsymbol{\omega}_k \triangleq \boldsymbol{\omega}_{S,k} - \boldsymbol{\omega}_{bt,k}$, $\boldsymbol{\omega}_{bt,k} \approx \boldsymbol{\omega}_{bt,k-1}$ and $\bar{\boldsymbol{\omega}} \triangleq (\boldsymbol{\omega}_{k-1} + \boldsymbol{\omega}_k)/2$.

C. *Discrete time*

For convenience, we start with the integration of the error state. For space reasons, we show here only the case with first-order Taylor truncation of the transition matrix ($\mathbf{F}_1$), zeroth-order backward quaternion integration (Q0B), and global angular error definition (GE), to integrate the error-state equations (11),

$$\delta\mathbf{p} \leftarrow \delta\mathbf{p} + \Delta t\, \delta\mathbf{v} \tag{14a}$$

$$\delta\mathbf{v} \leftarrow \delta\mathbf{v} - \lfloor \mathbf{R}(\mathbf{a}_S - \mathbf{a}_b) \rfloor_\times \Delta t\, \delta\boldsymbol{\theta} - \mathbf{R}\Delta t\, \delta\mathbf{a}_b + \mathbf{v}_i \tag{14b}$$

$$\delta\boldsymbol{\theta} \leftarrow \delta\boldsymbol{\theta} - \mathbf{R}\Delta t\, \delta\boldsymbol{\omega}_b + \boldsymbol{\theta}_i \tag{14c}$$

$$\delta\mathbf{a}_b \leftarrow \delta\mathbf{a}_b + \mathbf{a}_i \tag{14d}$$

$$\delta\boldsymbol{\omega}_b \leftarrow \delta\boldsymbol{\omega}_b + \boldsymbol{\omega}_i \ , \tag{14e}$$

where "$\leftarrow$" stands for "gets updated with", *i.e.*, $x \leftarrow f(x, \bullet)$ is equivalent to $x_k = f(x_{k-1}, \bullet)$. Here, $\mathbf{v}_i$, $\boldsymbol{\theta}_i$, $\mathbf{a}_i$ and $\boldsymbol{\omega}_i$ are random impulses applied to the velocity, orientation and bias estimates, modeled with Gaussian processes. Their mean is zero, and their covariance matrices are obtained by integrating the variances of the IMU measurement noises, $\mathbf{a}_n$, $\boldsymbol{\omega}_n$, and the IMU bias random walks, $\mathbf{a}_\omega$, $\boldsymbol{\omega}_\omega$, over the time step Δt,

$$\mathbf{V}_i = \sigma^2_{a_n}\, \Delta t^2\, \mathbf{I}_3 \ \ [m^2/s^2] \tag{15a}$$

$$\boldsymbol{\Theta}_i = \sigma^2_{\omega_n}\, \Delta t^2\, \mathbf{I}_3 \ \ [rad^2] \tag{15b}$$

$$\mathbf{A}_i = \sigma^2_{a_w}\, \Delta t\, \mathbf{I}_3 \ \ [m^2/s^4] \tag{15c}$$

$$\boldsymbol{\Omega}_i = \sigma^2_{\omega_w}\, \Delta t\, \mathbf{I}_3 \ \ [rad^2/s^2] \ , \tag{15d}$$

where $\sigma_{a_n}[m/s^2]$, $\sigma_{\omega_n}[rad/s]$, $\sigma_{a_w}[m/s^2\sqrt{s}]$ and $\sigma_{\omega_w}[rad/s\sqrt{s}]$ are to be determined from the information in the IMU datasheet, from real measurements, or –preferably as a last resort– via filter tuning.

Integration of the EKF equations (10) and the nominal state of the ESKF follow exactly the same scheme. We use 2nd order Taylor truncation ($\mathbf{F}_2$), and show here the expression with Q0B quaternion integration (we removed all subscripts $\bullet_t$ for clarity),

$$\mathbf{p} \leftarrow \mathbf{p} + \mathbf{v}\,\Delta t + \frac{1}{2}(\mathbf{R}(\mathbf{a}_S - \mathbf{a}_b) + \mathbf{g})\,\Delta t^2 \tag{16a}$$

$$\mathbf{v} \leftarrow \mathbf{v} + (\mathbf{R}\,(\mathbf{a}_S - \mathbf{a}_b) + \mathbf{g})\,\Delta t + \mathbf{v}_i \tag{16b}$$

$$\mathbf{q} \leftarrow \mathbf{q} \otimes \mathbf{q}\{(\boldsymbol{\omega}_S - \boldsymbol{\omega}_b)\Delta t + \boldsymbol{\theta}_i\} \tag{16c}$$

$$\mathbf{a}_b \leftarrow \mathbf{a}_b + \mathbf{a}_i \tag{16d}$$

$$\boldsymbol{\omega}_b \leftarrow \boldsymbol{\omega}_b + \boldsymbol{\omega}_i \ . \tag{16e}$$

D. *Observability, and marginally-observable modes*

The observability analysis of the system needs the evaluation of the rank and continuous symmetries of the observability matrix defined from the Lie derivatives [19]. Following this work, we detect three continuous symmetries,

$$\boldsymbol{\omega}_s^1 = [1, 0, 0, 0, \cdots, 0] \tag{17}$$

$$\boldsymbol{\omega}_s^2 = [0, 1, 0, 0, \cdots, 0] \tag{18}$$

$$\boldsymbol{\omega}_s^3 = [-p_y, p_x, 0, -v_y, v_x, 0, -\frac{q_z}{2}, -\frac{q_y}{2}, \frac{q_x}{2}, \frac{q_w}{2}, 0, \cdots] \tag{19}$$

corresponding to the non-observable modes of XY translation, and rotation around the direction of gravity (*i.e.*, the *yaw* angle), see [19]. All other modes, including all biases, are observable as long as the maneuvers performed span the observable directions. This means that the limitations on maneuverability imposed by the MAV dynamics have a negative impact on the observability of certain modes, in particular on the accelerometer bias in the XY axes, and the gyrometer bias in the Z axis. We explore these marginally-observable modes experimentally in section V.

IV. Extended and Error-State Kalman Filters

A. *Extended Kalman filter*

The development of an EKF from equations (16) and (8) is standard procedure and we do not include it here for space reasons. The transition matrix $\mathbf{F}_2$ is the Jacobian of (16).

B. *Error-state Kalman filter*

a) Nominal-state propagation: The nominal state is propagated according to (16) with all perturbation impulses $\{\mathbf{v}_i, \boldsymbol{\theta}_i, \mathbf{a}_i, \boldsymbol{\omega}_i\}$ set to zero.

b) Error-state KF prediction: The error-state system (14) can be posed as

$$\delta\mathbf{x} \leftarrow \mathbf{F}_N\, \delta\mathbf{x} + \mathbf{i} \tag{20}$$

where $\mathbf{i} \sim \mathcal{N}\{0, \mathbf{Q}\}$ is a vector of Gaussian random impulses with zero mean and covariances matrix $\mathbf{Q}$,

$$\mathbf{Q} = \mathrm{diag}(\mathbf{0}_{3\times3}, \mathbf{V}_i, \boldsymbol{\Theta}_i, \mathbf{A}_i, \boldsymbol{\Omega}_i)\,. \tag{21}$$

The ESKF time-update is applied only to the covariance,

$$\mathbf{P} \leftarrow \mathbf{F}_N\, \mathbf{P}\, \mathbf{F}_N^\top + \mathbf{Q} \ , \tag{22}$$

with $\mathbf{F}_N$ computed according to (12) with $N \in \{1, 2, 3\}$, and where $\mathbf{A}$ is the system matrix derived from (11),

$$\mathbf{A} = \begin{bmatrix} 0 & \mathbf{I} & 0 & 0 & 0 \\ 0 & 0 & -\lfloor \mathbf{R}(\mathbf{a}_S - \mathbf{a}_b) \rfloor_\times & -\mathbf{R} & 0 \\ 0 & 0 & 0 & 0 & -\mathbf{R} \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{23}$$

which may accept some variations depending on the algorithm options, especially GE *vs.* LE.

c) Error-state KF correction: Our PX4 sensor output depends on the system state as shown in (8). This is written in function form as,

$$\mathbf{v}_{of} = h(\mathbf{x}_t) + \mathbf{n}_{of}. \tag{24}$$

Based on this, the ESKF update is written,

$$\mathbf{K} = \mathbf{P}\mathbf{H}^\top(\mathbf{H}\mathbf{P}\mathbf{H}^\top + \mathbf{N}_{of})^{-1} \tag{25a}$$

$$\hat{\delta\mathbf{x}} = \mathbf{K}(\mathbf{v}_{of} - h(\mathbf{x})) \tag{25b}$$

$$\mathbf{P} \leftarrow \mathbf{P} - \mathbf{K}(\mathbf{H}\mathbf{P}\mathbf{H}^\top + \mathbf{N}_{of})\mathbf{K}^\top \tag{25c}$$

$$\mathbf{x} \leftarrow \mathbf{x} \oplus \hat{\delta\mathbf{x}}, \tag{25d}$$

where $\mathbf{H} = \partial h(\mathbf{x}_t)/\partial\delta\mathbf{x}$ is the observation matrix, derived from (8) and (3–5a) for GE (a slightly different expression is obtained for LE using with (5b)),

$$\mathbf{H} = \mathbf{S}\begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3x3} & \mathbf{0}_{3x3} & \mathbf{0}_{3x6} \\ \mathbf{0}_{3x3} & \mathbf{R}^\top & \mathbf{R}^\top\lfloor\mathbf{v}\rfloor_\times & \mathbf{0}_{3x6} \end{bmatrix}, \tag{26}$$

and $\oplus$ is the same composition operator explained in Sec. II-A, paying special attention to the options in (5).

V. Validation with Simulations

In order to study the performances and limitations of the proposed setup, we first present experiments with synthetic data under realistic flight conditions. Quadrotors are typically equipped with four aligned coplanar propellers. Motion control is achieved by altering the rotation speed of these propellers, thereby changing its torque load and thrust lift characteristics. With this actuation technique, a quadrotor becomes an under-actuated vehicle with only 4 DOF. Translational accelerations are achieved by tilting the rotor plane in the desired direction. We generate a ground truth trajectory that accounts for these motion constraints. We subsequently generate corrupted PX4 and IMU measurements, with noise and bias characteristics similar to those of the real units (see next section for noise values). Both sensors deliver data synchronously at 100Hz.

Fig. 2 shows the results with simulated data using an ESKF with GE, Q0B and $\mathbf{F}_1$. The 60 s trajectory encompasses an initial phase of hovering (phase A); a forward movement along the x axis (B); a backward movement (C) to the initial position; a second hovering phase (A); and finally left (D) and then right (E) movements along the y axis. The true trajectory finishes precisely at the starting position. X-Y-Z magnitudes are colored R-G-B, and $\pm 3\sigma$ estimated error bounds are also plotted.

The xy position is not observable, and its estimated error increases with time (top frame in Fig. 2). However, the drift is small, a few centimeters after the whole one-minute flight. Altitude and velocity estimates converge quickly thanks to the PX4 measurements. Notice the transient increase of the position and velocity errors (phases B and D). This is due to the uncertainty in yaw (blue track of third plot in Fig 2, not observable) which produces position errors perpendicular to the displacement vector. These errors decrease as the MAV returns to the origin (C and E).

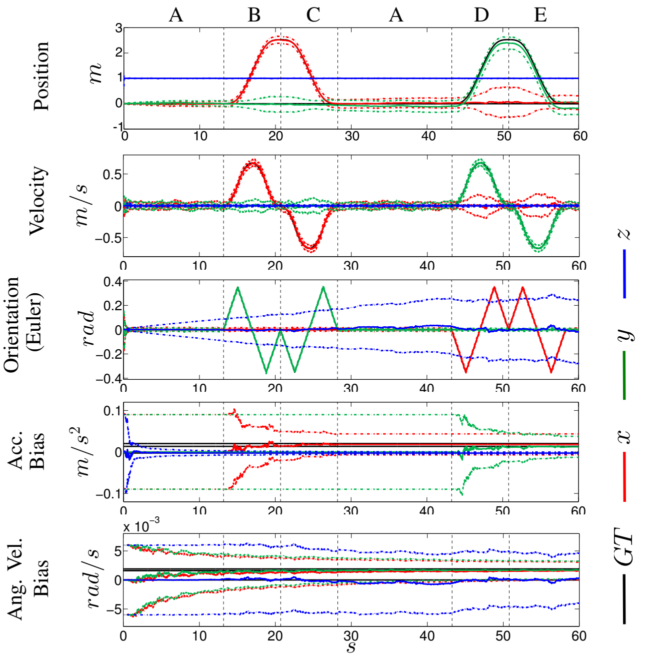

Fig. 2. Trajectories of all the estimated states, with their $\pm 3\,\sigma$ bounds, for the simulation experiment. The vertical sections indicate the hovering (A), forward (B), backward (C), left (D) and right (E) maneuvers.

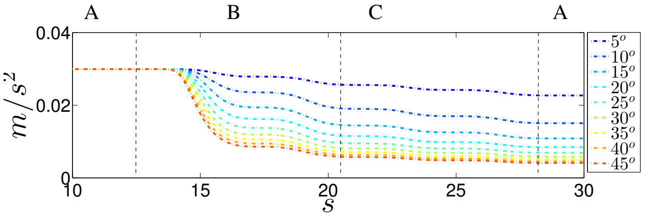

Fig. 3. Evolution of the estimated error of the accelerometer bias in the x (y) axis for different *pitch* (*roll*) angles.

One of the most interesting aspects is the marginal observability of some of the biases. On one hand, an acceleration bias becomes observable when the quadrotor orientation produces a non-null projection of the gravity vector over its axis. This makes the accelerometer's z-bias readily observable, but x- and y-biases require tilting the rotor plane. This is observable in Fig. 2, plot 'Acc. Bias'. This is a critical issue due to the quadrotor's restricted dynamics, seeking always horizontality and thus making these observations difficult. To further investigate this effect, the impact of a rotation over y (resp. x) on the observability of $\mathbf{a}_b(x)$ (resp. $\mathbf{a}_b(y)$) is illustrated in Fig. 3. A quadrotor can easily tilt to 20^o, but holding this tilt for a period of seconds represents a sustained high acceleration that is usually not desirable. Otherwise, this bias should be estimated beforehand and set up in the filter.

On the other hand, The gyroscope's *pitch* and *roll* biases are well observable thanks to the gravity vector. The *yaw* bias $\boldsymbol{\omega}_b(z)$ (blue in 'Ang. Vel' plot) becomes observable only after a very long period, by taking small profit of transient inclinations that bring the gravity vector away from the local z axis. To achieve a better performance in yaw-bias, it is advisable to use extra heading measurements such as a compass or other stable external references.

TABLE II

KF TUNING PARAMETERS (STD; σ SYMBOLS OMITTED)

Sensor noises				Initial std. dev.			
a_n	ω_n	z_{of}	v_{of}	δz	$\delta\phi, \delta\theta$	δa_b	$\delta\omega_b$
m/s^2	rad/s	m	m/s	m	rad	m/s^2	rad/s
0.4	0.005	0.05	0.1	0.05	0.05	0.02	0.004

VI. REAL MAV EXPERIMENTS

We present experiments with an ASCTEC PELICAN quadrotor. The IMU and the optical flow sensors deliver data asynchronously at 100Hz, and we associate each PX4 measurement with the closest IMU measurement. The sensors are attached below the platform (Fig. 1) using silicon damping links to reduce motor vibrations.

Tab. II shows all non-null sigma values for the filter tuning. All filter variants use the same parameters. For the IMU noises, we followed the datasheet to obtain $\sigma_{a_n} = 5.3\,mm/s^2$ and $\sigma_{\omega_n} = 3.6\,mrad/s$. However, empirical IMU data has revealed that the mechanical damping is not sufficient, and propeller vibrations inject much higher noises to the acceleration measurements. For this reason, the value of σ_{a_n} is significantly increased as shown in the table. We assume null bias random walks, as our flights are short enough. Initial uncertainties are mostly null, except for altitude $\sigma_{\delta z}$, initial roll $\sigma_{\delta\phi}$ and pitch $\sigma_{\delta\theta}$, accelerometer bias $\sigma_{\delta a_b}$, and gyrometer roll and pitch bias $\sigma_{\delta\omega_b}$.

A. Improving PX4 usability

The PX4 sensor has two key limitations that need to be addressed algorithmically in order to improve its robustness and usability. The first one is its inability to measure altitudes below 30 cm. For this, we divide each experiment in 4 phases depending on the robot state: landed (A), taking-off (B), flying (C) or landing (D). Thus during A, B and D intervals, the PX4 output is not reliable. We address this problem by reading status data of the robot to acknowledge whether it is in flying mode or not (*e.g.*, motors on/off). Before take-off (phase A), we overwrite the optical flow measurements by assuming that the MAV is on zero position, with zero velocity, and we set a small observation covariance. Thus, the bias uncertainties on the z acceleration and x and y angular velocities are reduced. If the robot is flying below the minimum altitude of $0.3m$ (phases B and D), detected with PX4 readings, we set high PX4 covariances because the measures are not trustable. Hence the filter is propagating the nominal state with the IMU dynamical model, and practically does not correct with the PX4 sensor. During regular flight (phase C) the observation covariances are set to those in Tab. II, thus allowing the PX4 to correct the estimations properly.

A last feature of our algorithm deals with the second PX4 data integrity issue. In some occasions, tilts over 20 degrees produce wrong sonar echoes, deriving in aberrant PX4 outputs. We detect altitude outliers using the modified Thompson Tau method [20]. Once an outlier is detected, the filter correction can be accomplished by using the optical flow measurement as a function of the IMU's gyrometer measurement $\boldsymbol{\omega}_S$ and the state variables,

$$\mathbf{\Phi}_{of} = \mathbf{S}'\left(\boldsymbol{\omega}_S \times \begin{bmatrix}0 & 0 & 1\end{bmatrix}^\top - \frac{\mathbf{R}\{\mathbf{q}\}^\top \mathbf{v}}{\mathbf{p}(z)}\right)\triangle t \cdot f + \mathbf{n}_\Phi \tag{27}$$

with $\mathbf{n}_\Phi$ a one-pixel Gaussian noise. For simplicity, we however re-used the observation model (8) by computing the velocity $\mathbf{v}_{of}$ with (9) substituting $h_{of} \leftarrow \mathbf{p}(z)$ and $\boldsymbol{\omega}_{of} \leftarrow \boldsymbol{\omega}_S$. In this case, to compensate for the correlation between the state and the measurement, we increased the sensor covariances $\mathbf{N}_{of}$ slightly. Fig. 4 shows the reconstructed $\mathbf{v}_{of}$ in front of typical outliers.

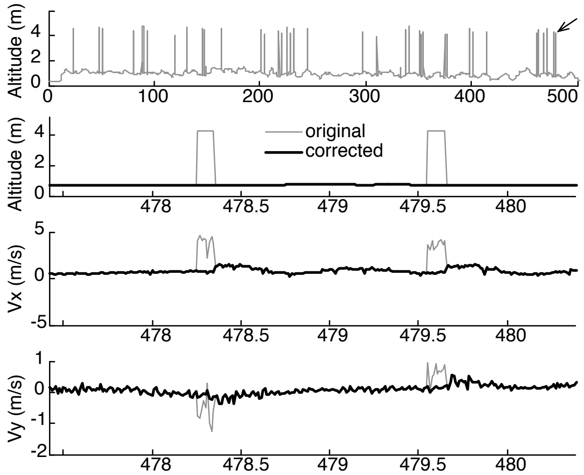

Fig. 4. Outlier detection and correction of PX4 measurements. Top: full 500 s altitude sequence with numerous outliers. Bottom plots: zoom of the arrowed outliers, showing original and corrected altitude and velocity measurements.

B. Benchmarking all filter variants against ground truth

We present the results in an indoor testbed equipped with an eight-camera Optitrack motion capture system [21], with a flying area of some 3×5 m. Our aim is at comparing the outcome of our different filter variants against ground truth. Fig. 5(a) shows the overhead plot of a 100m sample trajectory (using an ESKF with GE, Q1, and $\mathbf{F}_3$). Inspired by the methodologies in [7], [22], [23], we evaluate the odometry RMS errors as a function of the trajectory length, over 9 different trajectories. For each segment, the starting ground truth and estimation poses are aligned, and the error is evaluated at the end of the segment. Fig. 5(b) shows the negligible difference existing between a simple EKF and the aforementioned ESKF. Considering our flying zone, the statistic evaluation of these errors is only relevant for segment lengths up to around 10m (less than one turn to the arena), because after that the trajectory is rolled over itself and the nonlinear composition of angular and linear errors produces optimistic error evaluations. Below these 10m, odometry errors fit with the expected law of the square root of the length d of the evaluated segments, as shown in Fig. 6(a). This good fit indicates that the yaw error contribution to position error is small. The yaw error, Fig. 6(b), does not suffer from this nonlinear composition and fits the $\sqrt{d}$ better. At an average of $0.2°$ over the 10m, the induced linear error

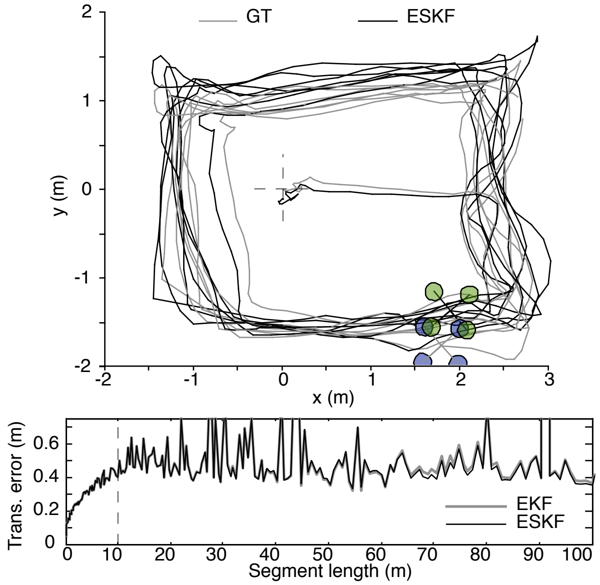

Fig. 5. Benchmark results. a) Top view of ground-truth (Optitrack system) vs. ESKF-estimated trajectory with a path length of 100m. b) Translation RMS error with respect to ground-truth, measured for different path segment lengths between 0.1m and 100m. After the first 10m, the trajectory rolls over itself and errors combine nonlinearly producing optimistic figures.

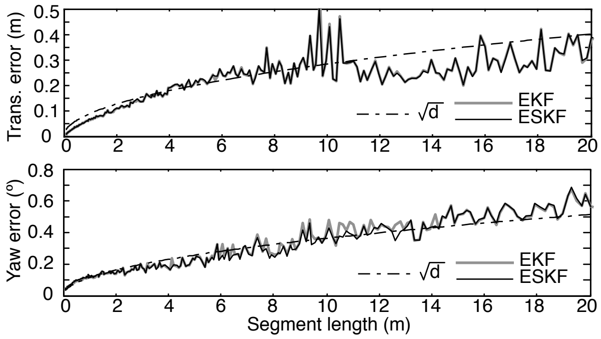

Fig. 6. Benchmark results. Translation and yaw RMS errors with respect to ground-truth, measured for different path segment lengths d between 0.1 m and 20 m.

would be of the order of only $10\text{m} \sim (0.2^\circ) = 3.5\,\text{cm}$, a small fraction of the 30cm total position drift.

To evaluate all 21 possible filter variants (shown previously in Tab. I), we identified 100 straight segments of 2m long over 9 paths. The position RMSEs at the end of the segments are shown in Tab. III. Notice that the performance of all filter variants is essentially equivalent.

C. *Outdoor arena experiments*

The outdoor scenario corresponds to a GPS-denied zone in which ground truth from external positioning systems is not available. Instead, we perform manual flights around some fixed obstacles, taking care to take-off and land on the same base point.

Fig. 7 shows the calibrated outdoor scenario with the take-off and landing platform in the center, surrounded by vertical cylindric obstacles. Fig. 8 shows the estimated trajectories of 15 flights of approximately two minutes and 70 m each (using an ESKF with GE, Q1 and $\mathbf{F}_3$). The quadrotor is driven manually around the obstacles, which results in different flight paths. The final estimated land points have a standard deviation of $\sigma = \begin{bmatrix}0.50 & 0.53 & 0.01\end{bmatrix}$ m from the center of the landing base. The orange ellipse in Fig. 8 corresponds to the zone of confidence of 95% probability of this landing point distribution. All true landings were done inside the base area which measures $0.4 \times 0.4\,m$. The orange ellipse is thus the composition of the landing error and the estimation error. The blue line corresponds to a sample flight whose specific results are shown in Fig. 9. Notice in the zoomed details of position and velocity plots how the estimation errors increase during take-off and landing periods due to the PX4 limitations explained in Sec. VI-A. The transition to observability B→C once the PX4 data is recovered is also visible.

TABLE III

POSITION RMS ERRORS (m) AFTER 2 M FLIGHT.

Quat. error	GE			LE		
Quat. int.	Q0F	Q0B	Q1	Q0F	Q0B	Q1
EKF $\mathbf{F}_2$	-	-	-	0.1984	0.2003	0.1993
ESKF $\mathbf{F}_1$	0.1986	0.2007	0.1996	0.1986	0.2005	0.1995
ESKF $\mathbf{F}_2$	0.1987	0.2007	0.1997	0.1987	0.2005	0.1996
ESKF $\mathbf{F}_3$	0.1987	0.2007	0.1997	0.1987	0.2005	0.1996

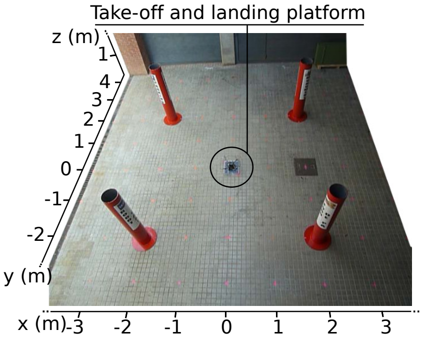

Fig. 7. Calibrated outdoor flying arena for real robot testings. The quadrotor takes off and lands at the same point in the middle of the field (base).

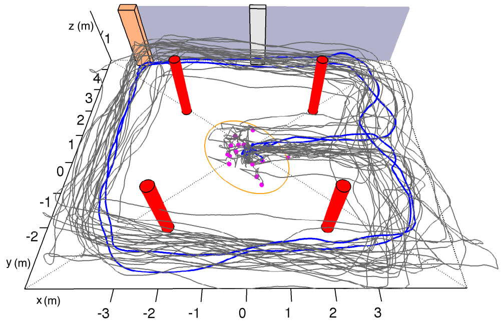

Fig. 8. Trajectory estimations for 15 different flights (2 minutes, 70 m each). A sample trajectory is shown in blue. The orange ellipse corresponds to the 95% confidence region for the landing point.

VII. CONCLUSIONS

This paper presents a simple method for rich and high-frequency vehicle state estimation fusing low-cost inertial and smart optical flow measurement units for a MAV, using low complexity algorithms without sophisticated features.

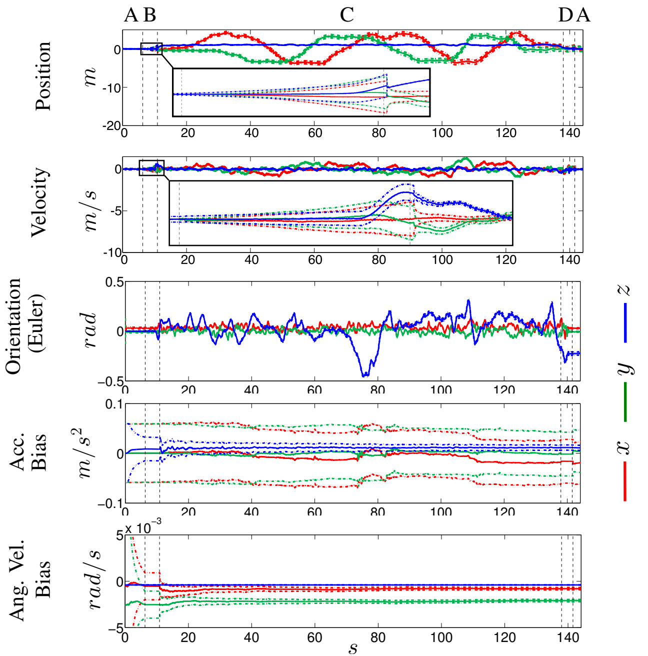

Fig. 9. Trajectories of all the estimated states, with their $\pm 3\,\sigma$ bounds, for the outdoors experiment. The vertical sections indicate the landed states (A), and the take-off (B), flight (C) and landing (D) maneuvers. Zooms are provided to appreciate the transition between phases B→C.

A large amount of variants for nonlinear KF have been benchmarked, revealing that, for such a basic setup, with high input data rates, the plain EKF filter performs equivalently to other, more sophisticated options.

A new observation model is derived for this sensor combination, and the observability limitations of the system are pointed out. Namely, it is difficult to recover the IMU's xy accelerometer and yaw gyrometer biases because the quadrotor dynamics prevent from severe orientation changes.

Methods have also been proposed to overcome situations of poor PX4 data integrity, mainly due to low altitude or high tilt angles where the sonar data fails. These improvements have revealed crucial for improving the robustness of such a low-cost setup to acceptable levels.

We can think of three avenues for further research. First, a calibration warm-up period can be specially designed for quadrotors to compensate for the marginal observability. Second, some other validation experiments can be done using an IMU with lower cost than the Microstrain 3dm-gx3-25. Finally, the non-observable modes can be made observable by adding other sensor readings; for example, compass readings can easily avoid the yaw drift.

VIII. Acknowledgements

The authors would like to thank the unmanned systems unit from Eurecat Technology Center for giving us access to their indoor flying arena and facilities.

References

[1] H. Liu, H. Darabi, P. Banerjee, and J. Liu, "Survey of wireless indoor positioning techniques and systems," *Syst., Man, and Cyb., Part C: App. and Reviews, IEEE Trans. on*, vol. 37, no. 6, pp. 1067–1080, Nov 2007.

[2] J. Engel, J. Sturm, and D. Cremers, "Camera-based navigation of a low-cost quadrocopter," in *Proc.IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Vilamoura, Oct. 2012, pp. 2815–2821.

[3] R. G. Valenti, I. Dryanovski, C. Jaramillo, D. P. Strom, and J. Xiao, "Autonomous quadrotor flight using onboard RGB-D visual odometry," in *Proc.IEEE Int. Conf. Robotics Autom.*, Hong Kong, May 2014, pp. 5233–5238.

[4] C. Roussillon, A. Gonzalez, J. Solà, J. M. Codol, N. Mansard, S. Lacroix, and M. Devy, "RT-SLAM: A generic and real-time visual SLAM implementation," in *Computer Vision Systems*, ser. Lecture Notes in Computer Science, J. Crowley, B. Draper, and M. Thonnat, Eds. Springer Berlin Heidelberg, 2011, vol. 6962, pp. 31–40.

[5] M. Li and A. I. Mourikis, "High-precision, consistent EKF-based visual–inertial odometry," *Int. J. Robotics Res.*, vol. 32, no. 6, pp. 690–711, 2013.

[6] A. I. Mourikis and S. I. Roumeliotis, "A multi-state constraint Kalman filter for vision-aided inertial navigation," in *Proc.IEEE Int. Conf. Robotics Autom.*, Rome, Apr. 2007, pp. 3565–3572.

[7] K. Konolige, M. Agrawal, and J. Solà, "Large-scale visual odometry for rough terrain," in *Robotics Research*. Springer Berlin Heidelberg, 2011, pp. 201–212.

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

[9] S. Weiss, M. Achtelik, S. Lynen, M. Chli, and R. Siegwart, "Real-time onboard visual-inertial state estimation and self-calibration of mavs in unknown environments," in *Proc.IEEE Int. Conf. Robotics Autom.*, Saint Paul, May 2012, pp. 957–964.

[10] S. Omari and G. Ducard, "Metric visual-inertial navigation system using single optical flow feature," in *Proc.Eur. Control Conf.*, Zurich, Jul. 2013, pp. 1310–1316.

[11] M. Bloesch, S. Omari, P. Fankhauser, H. Sommer, C. Gehring, J. Hwangbo, M. Hoepflinger, M. Hutter, and R. Siegwart, "Fusion of optical flow and inertial measurements for robust egomotion estimation," in *Proc.IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Chicago, Sept 2014.

[12] H. Chao, C. Coopmans, L. Di, and Y.-Q. Chen, "A comparative evaluation of low-cost IMUs for unmanned autonomous systems," in *Proc.IEEE Int. Conf. Multisens. Fus. and Integ. for Int. Syst.*, Salt Lake City, Sep. 2010, pp. 211–216.

[13] D. Honegger, M. Lorenz, P. Tanskanen, and M. Pollefeys, "An open source and open hardware embedded metric optical flow CMOS camera for indoor and outdoor applications," in *Proc.IEEE Int. Conf. Robotics Autom.*, Karlsruhe, May. 2013, pp. 1736–1741.

[14] A. Santamaria-Navarro and J. Andrade-Cetto, "Uncalibrated image-based visual servoing," in *Proc.IEEE Int. Conf. Robotics Autom.*, Karlsruhe, May. 2013, pp. 5247–5252.

[15] A. Santamaria-Navarro, V. Lipiello, and J. Andrade-Cetto, "Task priority control for aerial manipulation," in *Proc.IEEE Int. Symp. Safe. Sec. Resc. Robotics.*, Toyako-cho, Oct. 2014, pp. 1–6.

[16] V. K. Madyastha, V. C. Ravindra, S. Mallikarjunan, and A. Goyal, "Extended Kalman filter vs. error state Kalman filter for aircraft attitude estimation," in *Proc.AIAA Guid., Nav., and Control Conf.*, Portland, Aug. 2011, pp. 6615–6638.

[17] M. Li and A. I. Mourikis, "Improving the accuracy of EKF-based visual-inertial odometry," in *Proc.IEEE Int. Conf. Robotics Autom.*, Saint Paul, May. 2012, pp. 828–835.

[18] N. Trawny and S. I. Roumeliotis, "Indirect Kalman filter for 3d attitude estimation," *University of Minnesota, Dept. of Comp. Sci. & Eng., Tech. Rep*, vol. 2, 2005.

[19] A. Martinelli, "Vision and imu data fusion: Closed-form solutions for attitude, speed, absolute scale, and bias determination," *IEEE Trans. Robotics*, vol. 28, no. 1, pp. 44–60, Feb 2012.

[20] J. M. Cimbala, "Outliers," Sep. 2011. [Online]. Available: https://www.mne.psu.edu/me345/Lectures/outliers.pdf

[21] N. Point, "Inc.: Optitrack-optical motion tracking solutions," 2009.

[22] S. Leutenegger, S. Lynen, M. Bosse, R. Siegwart, and P. Furgale, "Keyframe-based visual–inertial odometry using nonlinear optimization," *Int. J. Robotics Res.*, 2014.

[23] A. Geiger, P. Lenz, and R. Urtasun, "Are we ready for autonomous driving? the kitti vision benchmark suite," in *Proc.IEEE Conf. on Comp. Vision and Patt. Recog.*, Providence, Jun. 2012, pp. 3354–3361.