

A Comparison of Scale Estimation Schemes for a Quadrotor UAV based on Optical Flow and IMU Measurements

Volker Grabe, Heinrich H. Bühlhoff, and Paolo Robuffo Giordano

Abstract—For the purpose of autonomous UAV flight control, cameras are ubiquitously exploited as a cheap and effective onboard sensor for obtaining *non-metric* position or velocity measurements. Since the metric scale cannot be directly recovered from visual input only, several methods have been proposed in the recent literature to overcome this limitation by exploiting independent ‘metric’ information from additional onboard sensors. The flexibility of most approaches is, however, often limited by the need of constantly tracking over time a certain set of features in the environment, thus potentially suffering from possible occlusions or loss of tracking during flight. In this respect, in this paper we address the problem of estimating the scale of the observed linear velocity in the UAV body frame from direct measurement of the instantaneous (and non-metric) optical flow, and the integration of an onboard Inertial Measurement Unit (IMU) for providing (metric) acceleration readings. To this end, two different estimation techniques are developed and critically compared: a standard *Extended Kalman Filter* (EKF) and a novel *nonlinear observer* stemming from the adaptive control literature. Results based on simulated and real data recorded during a quadrotor UAV flight demonstrate the effectiveness of the approach.

I. INTRODUCTION

Over the last years, Unmanned Aerial Vehicles (UAVs) and in particular quadrotors became a highly popular robotic platform. Being cheap and flexible, quadrotors can be used for many different applications such as exploration, mapping or inspection tasks. Recently, with the support of several EU-funded projects, UAVs started to enter also the area of service robotics [1], [2], thus opening a new wide range of possibilities and challenges.

Key to almost all applications involving high autonomy levels is a reliable (self-)motion estimation to allow for an effective flight control performance. Among the typical sensors used for this goal, cameras represent a lightweight, cheap, and flexible choice. While cameras are often installed on UAVs for the purpose of aerial photography or target localization, they can also be exploited to retrieve the UAV position or velocity w.r.t. the observed scene. Using a purely visual approach, however, any position or velocity information can only be retrieved up to an unknown scale factor — the well-known *scale-ambiguity* affecting perspective

projection systems [3]. Nevertheless, the actual metric scale can still be recovered by either exploiting some pre-known structure of the scene, or by fusing the visual information with independent metric measurements.

For instance, in the case of a known initial map, the scale can be recovered from the observation of a sufficiently large number of feature points at known locations [4]. Alternatively, when an initial metric map is not available, metric sensory information from, e.g., distance sensors or accelerometer readings can be exploited for this purpose. The commercially available AR.Drone uses ultrasonic distance and an air pressure sensor to estimate its height over ground. Other proposed frameworks exploiting the availability of an onboard IMU, however, are also dependent on SLAM techniques to build up and maintain a map of tracked features. In these cases, the main objective is to recover the global scale factor of the underlying map. This has been achieved for the popular Parallel Tracking and Mapping (PTAM) toolbox [5], [6], for a general position estimator [7], and for the recovery of the traveled trajectory using visual odometry [8]. Even more remarkably, in [9] it has been shown how to estimate in closed-form the absolute scale from IMU readings and observation of a *single* feature point over time.

A. Optical flow-based velocity estimation

All these approaches, however, highly depend on the possibility to continuously track or recognize individually identified features for an extended period of time. This assumption might be too restrictive when navigating with only on-board sensing/computational power in unknown or cluttered environments because of, e.g., unexpected occlusions or the heavy computations needed for feature matching. This is not the case, on the other hand, for those solutions based on an instantaneous optical flow decomposition as they allow for a self-motion estimation on a frame-to-frame basis, thus requiring the most minimal tracking capabilities. Motivated by these considerations, in [10], [11] we proposed, and experimentally tested on a quadrotor UAV, an algorithm for decomposing the (instantaneous) observed optical flow based on the so-called *continuous homography constraint*. In particular, assuming a predominant planar patch in the scene, we showed how to recover a scaled (non-metric) linear velocity which could also be used for closed-loop flight control purposes.

The goal of the present paper is to extend this latter approach to the online recovery of the (unknown) scale factor by exploiting the concurrent measurements of an onboard IMU. In particular, we propose, discuss and compare

V. Grabe is with the Max Planck Institute for Biological Cybernetics, Spemannstraße 38, 72076 Tübingen, Germany and the University of Zurich, Andreasstrasse 15, 8050 Zurich, Switzerland volker.grabe@tuebingen.mpg.de.

P. Robuffo Giordano is with CNRS at IRISA, Campus de Beaulieu, 35042 Rennes Cedex, France prg@irisa.fr.

H. H. Bühlhoff is with the Max Planck Institute for Biological Cybernetics, Spemannstraße 38, 72076 Tübingen, Germany, and with the Department of Brain and Cognitive Engineering, Korea University, Seoul, 136-713 Korea. E-mail: hbb@tuebingen.mpg.de.

two estimation schemes for achieving this goal: a classical Extended Kalman Filter (EKF), a common choice in most previous works on similar topics, and a novel nonlinear observer based on the theory of Persistency of Excitation (PE) stemming from the adaptive control community [12]. Both estimation schemes have their pros/cons: being the problem under consideration essentially nonlinear, the EKF requires a linearization step which may degrade the estimation performance. On the other hand, as it will be shown in the following, no linearization is needed in the PE case. Thus, the PE observer ought to perform better, in principle, than an EKF in all cases. However, the PE formulation assumes a *deterministic* dynamics, and copes with presence of noise in an indirect way. The EKF, in turn and despite its inherent approximations, is ‘noise-aware’, and thus more suited to deal with the typical noise levels of IMU and camera readings during a quadrotor flight.

Recently, a system capable of hovering and landing on a moving platform by exploiting optical flow was presented in [13]. This work did not consider the issue of determining the unknown scene scale factor since the proposed control approach could be implemented by only employing the scaled linear velocity. An optical flow-based approach combined with scale estimation was instead presented in [14] by developing a small sensor for on-board velocity estimation. However, the proposed metric scale estimation relies on a ground facing sonar, thus limiting the vehicle operation to near ground flights within the sonar range. Finally, in [15] the authors experimentally demonstrated the possibility of estimating the metric scale, sensor biases, and the IMU/camera relative pose via an EKF approach based on optical flow measurements. However, the system was mainly designed for serving as an initialization of the PTAM framework in near hovering mode rather than for closed-loop UAV control over an extended period of time.

With respect to this latter contribution, which shares some similarities with our work, we instead (i) propose and compare two independent estimation schemes for dealing with the scale estimation problem, (ii) provide a rigorous modeling of the complete system dynamics, including that of the scene scale factor, and (iii) rely on the continuous *homography constraint*, rather than on the *epipolar constraint*, for decomposing the perceived optical flow. In fact, in case of partly planar indoor or outdoor scenes, a homography-based approach should yield better results compared to an epipolar-based one (which becomes singular for a perfectly planar scene), and, additionally, it also remains valid in case of stationary flight (e.g., when moving close to hovering) [3].

The remainder of this paper is then structured as follows: in Sec. II we first review the self-motion estimation algorithms originally proposed in [10] and exploited also in this work. In Sec. III, we describe the equations of motion for the system under consideration. These are then exploited for estimating the metric scale of the scene by designing an Extended Kalman Filter (EKF) in Sec. IV, and the alternative PE nonlinear observer in Sec. V. Afterwards, in Sec. VI and VII we present the results of the conducted experi-

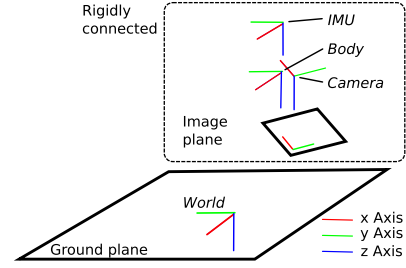


Fig. 1: Locations of the IMU (\mathcal{I}), camera (\mathcal{C}), body (\mathcal{B}) and world frame (\mathcal{W}) relative to each other. Frames \mathcal{I} , \mathcal{B} and \mathcal{C} are assumed to be rigidly linked to each other. The world frame \mathcal{W} is oriented horizontally with its z -axis pointing down, following the NED convention.

ments aimed at validating and comparing the two estimation schemes. Finally, we conclude the paper in Sec. VIII by giving a short overview of our future plans.

II. SELF-MOTION ESTIMATION FROM OPTICAL FLOW

The approach adopted in this paper for decomposing the perceived optical flow in order to recover the scaled linear velocity is taken from [10], [11]. For our experimental validation, as explained in the previous section, we relied on an improved version of the *continuous homography constraint* [11], [16]. We believe this choice has the important advantage over other possibilities (e.g., epipolar constraint) of being well-conditioned when observing planar scenes such as, e.g., during take-off or landing when using a downfacing camera, or when flying in GPS denied environments made of large planar regions. We thus refer the reader to [10], [11] for all the details. We nevertheless note that the scale estimation schemes proposed in the next Sect. IV and Sect. V are in general independent from how the scaled linear velocity is actually retrieved — therefore, any other choice yielding comparable results would have been appropriate.

As for the actual implementation, in brief: from consecutive frames, we first compute an optical flow field through detection and tracking of FAST corners using a pyramidal version of the Lukas-Kanade algorithm. This flow field is further derotated using angular velocity measurements from the onboard IMU [10]. We then identify a suitable subset of features belonging to a common planar region [11]. This set of flow vectors is used to recover the continuous homography matrix which is further decomposed into the scaled linear velocity $\frac{v}{d}$ and the normal vector of the underlying plane n by means of a modified version of the 4-point algorithm [16].

III. EQUATIONS OF MOTION

We now proceed to describe the equations of motion of our system (UAV + camera + IMU) necessary for then designing the scale estimation schemes presented in Sects. IV–V. In the following, we will denote with \mathcal{B} , \mathcal{C} , \mathcal{I} and \mathcal{W} the body, camera, IMU and inertial world frame, respectively. The origin of frame \mathcal{B} is assumed to be located at the quadrotor barycenter, while frames \mathcal{C} and \mathcal{I} are supposed to be rigidly attached to \mathcal{B} , see Fig. 1.

Throughout the text, left superscripts will be exploited to indicate the frames where quantities are expressed in. The symbol ${}^{\mathcal{X}}\mathbf{R}_{\mathcal{Y}} \in SO(3)$ will be used to denote the rotation matrix from frame \mathcal{X} to frame \mathcal{Y} , and ${}^{\mathcal{Z}}\mathbf{p}_{\mathcal{X}\mathcal{Y}} \in \mathbb{R}^3$ to represent the vector from the origin of frame \mathcal{X} to the origin of frame \mathcal{Y} and expressed in frame \mathcal{Z} . We also introduce the following quantities instrumental for the next developments: $\mathbf{g} \in \mathbb{R}^3$ as the gravity vector, and ${}^{\mathcal{I}}\mathbf{f} \in \mathbb{R}^3$, ${}^{\mathcal{I}}\boldsymbol{\omega} \in \mathbb{R}^3$ as the specific acceleration and angular velocity at the origin of \mathcal{I} .

Define ${}^{\mathcal{C}}\mathbf{v} = {}^{\mathcal{C}}\mathbf{R}_{\mathcal{W}} {}^{\mathcal{W}}\dot{\mathbf{p}}_{\mathcal{WC}}$ as the camera linear velocity in camera frame, and ${}^{\mathcal{B}}\mathbf{v} = {}^{\mathcal{B}}\mathbf{R}_{\mathcal{W}} {}^{\mathcal{W}}\dot{\mathbf{p}}_{\mathcal{WB}}$ as the body linear velocity in body frame. Since ${}^{\mathcal{B}}\mathbf{R}_{\mathcal{C}}$, ${}^{\mathcal{B}}\mathbf{R}_{\mathcal{I}}$, ${}^{\mathcal{B}}\mathbf{p}_{\mathcal{BC}}$ and ${}^{\mathcal{B}}\mathbf{p}_{\mathcal{BI}}$ are assumed constant, from standard kinematics the following relationships hold

$${}^{\mathcal{B}}\mathbf{v} = {}^{\mathcal{B}}\mathbf{R}_{\mathcal{C}} ({}^{\mathcal{C}}\mathbf{v} + [{}^{\mathcal{C}}\boldsymbol{\omega}]_{\times} {}^{\mathcal{C}}\mathbf{p}_{\mathcal{CB}}) = {}^{\mathcal{B}}\mathbf{R}_{\mathcal{C}} {}^{\mathcal{C}}\mathbf{v} + [{}^{\mathcal{B}}\boldsymbol{\omega}]_{\times} {}^{\mathcal{B}}\mathbf{p}_{\mathcal{CB}}, \quad (1)$$

$$\begin{aligned} {}^{\mathcal{C}}\dot{\mathbf{v}} &= {}^{\mathcal{C}}\mathbf{R}_{\mathcal{I}} ({}^{\mathcal{I}}\mathbf{a} + [{}^{\mathcal{I}}\dot{\boldsymbol{\omega}}]_{\times} {}^{\mathcal{I}}\mathbf{p}_{\mathcal{IC}} + [{}^{\mathcal{I}}\boldsymbol{\omega}]_{\times}^2 {}^{\mathcal{I}}\mathbf{p}_{\mathcal{IC}}) - [{}^{\mathcal{C}}\boldsymbol{\omega}]_{\times} {}^{\mathcal{C}}\mathbf{v} = \\ &= {}^{\mathcal{C}}\mathbf{R}_{\mathcal{I}} {}^{\mathcal{I}}\mathbf{a} + [{}^{\mathcal{C}}\dot{\boldsymbol{\omega}}]_{\times} {}^{\mathcal{C}}\mathbf{p}_{\mathcal{IC}} + [{}^{\mathcal{C}}\boldsymbol{\omega}]_{\times}^2 {}^{\mathcal{C}}\mathbf{p}_{\mathcal{IC}} - [{}^{\mathcal{C}}\boldsymbol{\omega}]_{\times} {}^{\mathcal{C}}\mathbf{v}, \end{aligned} \quad (2)$$

$${}^{\mathcal{C}}\boldsymbol{\omega} = {}^{\mathcal{C}}\mathbf{R}_{\mathcal{I}} {}^{\mathcal{I}}\boldsymbol{\omega} \quad (3)$$

$${}^{\mathcal{C}}\dot{\boldsymbol{\omega}} = {}^{\mathcal{C}}\mathbf{R}_{\mathcal{I}} {}^{\mathcal{I}}\dot{\boldsymbol{\omega}} \quad (4)$$

where ${}^{\mathcal{I}}\mathbf{a} = {}^{\mathcal{I}}\mathbf{R}_{\mathcal{W}} {}^{\mathcal{W}}\ddot{\mathbf{p}}_{\mathcal{WI}}$ is the linear acceleration experienced by the IMU. We note that ${}^{\mathcal{I}}\mathbf{a} = {}^{\mathcal{I}}\mathbf{f} + {}^{\mathcal{I}}\mathbf{g}$ and ${}^{\mathcal{I}}\mathbf{g} = {}^{\mathcal{I}}\mathbf{R}_{\mathcal{W}} [0, 0, g]^T$ in case of a horizontal orientation of the world frame, see Fig. 1.

Assume now presence of a planar scene with plane equation ${}^{\mathcal{C}}\mathbf{n}^T {}^{\mathcal{C}}\mathbf{P} + d = 0$, where ${}^{\mathcal{C}}\mathbf{P} \in \mathbb{R}^3$ is a generic point on the plane seen by the camera, ${}^{\mathcal{C}}\mathbf{n} \in \mathbb{S}^2$ is the unit normal vector to the plane, and $d \in \mathbb{R}$ the plane distance from the camera optical center. We then have (see, e.g., [17])

$${}^{\mathcal{C}}\dot{\mathbf{n}} = -[{}^{\mathcal{C}}\boldsymbol{\omega}]_{\times} {}^{\mathcal{C}}\mathbf{n} \quad (5)$$

$$\dot{d} = {}^{\mathcal{C}}\mathbf{v}^T {}^{\mathcal{C}}\mathbf{n}. \quad (6)$$

Finally, according to this notation, the decomposition of the optical flow summarized in the previous Section allows to directly measure the scaled linear velocity ${}^{\mathcal{C}}\tilde{\mathbf{v}} = {}^{\mathcal{C}}\mathbf{v}/d$. The estimation schemes presented in the next Sections are then meant to recover the (unmeasurable) value of the plane distance d and to consequently reconstruct the linear velocity vector ${}^{\mathcal{C}}\mathbf{v}$.

IV. SCALE ESTIMATION BASED ON THE EXTENDED KALMAN FILTER

We start designing an EKF for estimating the distance to the planar scene d . We will adopt the discrete version of the EKF, and let index $k \in \mathbb{N}$ denote the k -th iteration step. For clarity, we will also adopt the convention of appending a right subscript m to all those quantities directly available through one of the onboard sensors, e.g., specific force ${}^{\mathcal{I}}\mathbf{f}_m$ and angular velocity ${}^{\mathcal{I}}\boldsymbol{\omega}_m$ from the IMU, and scaled linear velocity ${}^{\mathcal{C}}\tilde{\mathbf{v}}_m = ({}^{\mathcal{C}}\mathbf{v}/d)_m$ from the camera.

A. Definitions

The EKF state vector \mathbf{x} consists of the camera linear velocity in camera frame ${}^{\mathcal{C}}\mathbf{v}$ and the camera distance to the

planar scene d :

$$\mathbf{x} = \begin{bmatrix} {}^{\mathcal{C}}\mathbf{v} \\ d \end{bmatrix}, \quad {}^{\mathcal{C}}\mathbf{v} \in \mathbb{R}^3, d \in \mathbb{R}. \quad (7)$$

B. Prediction

We rewrite (2) in terms of the measurements ${}^{\mathcal{I}}\mathbf{f}_m$ and ${}^{\mathcal{I}}\boldsymbol{\omega}_m$ obtained from the IMU in frame \mathcal{I} :

$$\begin{aligned} {}^{\mathcal{C}}\dot{\mathbf{v}} &= {}^{\mathcal{C}}\mathbf{R}_{\mathcal{I}} ({}^{\mathcal{I}}\mathbf{a} + [{}^{\mathcal{I}}\dot{\boldsymbol{\omega}}]_{\times} {}^{\mathcal{I}}\mathbf{p}_{\mathcal{IC}} + [{}^{\mathcal{I}}\boldsymbol{\omega}_m]_{\times}^2 {}^{\mathcal{I}}\mathbf{p}_{\mathcal{IC}}) - [{}^{\mathcal{C}}\boldsymbol{\omega}_m]_{\times} {}^{\mathcal{C}}\mathbf{v} \\ &\approx {}^{\mathcal{C}}\mathbf{R}_{\mathcal{I}} ({}^{\mathcal{I}}\mathbf{f}_m + {}^{\mathcal{I}}\mathbf{g} + [{}^{\mathcal{I}}\boldsymbol{\omega}_m]_{\times}^2 {}^{\mathcal{I}}\mathbf{p}_{\mathcal{IC}}) - [{}^{\mathcal{C}}\boldsymbol{\omega}_m]_{\times} {}^{\mathcal{C}}\mathbf{v}. \end{aligned} \quad (8)$$

Since no direct measurement of ${}^{\mathcal{I}}\dot{\boldsymbol{\omega}}$ is possible in our setup, and ${}^{\mathcal{I}}\boldsymbol{\omega}$ is usually a noisy signal, in (8) we approximate ${}^{\mathcal{I}}\dot{\boldsymbol{\omega}} \approx \mathbf{0}$ rather than attempting to recover ${}^{\mathcal{I}}\dot{\boldsymbol{\omega}}$ via a numerical differentiation. Consequently, using (6) and (8) the following equations govern the predicted state $\hat{\mathbf{x}}[k]_{k-1}$:

$${}^{\mathcal{C}}\hat{\mathbf{v}}[k]_{k-1} = {}^{\mathcal{C}}\hat{\mathbf{v}}[k-1] + T {}^{\mathcal{C}}\dot{\mathbf{v}}[k] \quad (9)$$

$$\hat{d}[k]_{k-1} = \hat{d}[k-1] + T {}^{\mathcal{C}}\hat{\mathbf{v}}[k]^T {}^{\mathcal{C}}\mathbf{n}[k] \quad (10)$$

where T denotes the sampling time of the filter.

Although most quantities derived in the following steps are time varying, from now on, for the sake of exposition clarity, we will omit the time dependency $[k]$ wherever possible. In order to compute the predicted covariance matrix of the system uncertainty $\Sigma[k]_{k-1} \in \mathbb{R}^{4 \times 4}$, we first define the Jacobian matrix $\mathbf{G}[k]_{k-1} \in \mathbb{R}^{4 \times 4}$

$$\begin{aligned} \mathbf{G} &= \begin{bmatrix} \frac{\partial {}^{\mathcal{C}}\hat{\mathbf{v}}[k]_{k-1}}{\partial {}^{\mathcal{C}}\hat{\mathbf{v}}[k-1]} & \frac{\partial {}^{\mathcal{C}}\hat{\mathbf{v}}[k]_{k-1}}{\partial \hat{d}[k-1]} \\ \frac{\partial \hat{d}[k]_{k-1}}{\partial {}^{\mathcal{C}}\hat{\mathbf{v}}[k-1]} & \frac{\partial \hat{d}[k]_{k-1}}{\partial \hat{d}[k-1]} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{I}_3 - T[{}^{\mathcal{C}}\boldsymbol{\omega}]_{\times} & \mathbf{0}_{3 \times 1} \\ T {}^{\mathcal{C}}\mathbf{n}^T & 1 \end{bmatrix}. \end{aligned} \quad (11)$$

Matrix $\Sigma[k-1]$ from the previous step is then propagated as:

$$\Sigma[k]_{k-1} = \mathbf{G}\Sigma[k-1]\mathbf{G}^T + \mathbf{R}. \quad (12)$$

Here, matrix $\mathbf{R} \in \mathbb{R}^{4 \times 4}$ is obtained from

$$\mathbf{R} = \mathbf{V} \begin{bmatrix} \text{cov}({}^{\mathcal{I}}\mathbf{f}_m) & \mathbf{0}_{3 \times 3} \\ \mathbf{0}_{3 \times 3} & \text{cov}({}^{\mathcal{I}}\boldsymbol{\omega}_m) \end{bmatrix} \mathbf{V}^T \quad (13)$$

where

$$\begin{aligned} \mathbf{V} &= \begin{bmatrix} \frac{\partial {}^{\mathcal{C}}\hat{\mathbf{v}}[k]_{k-1}}{\partial {}^{\mathcal{I}}\mathbf{f}_m} & \frac{\partial {}^{\mathcal{C}}\hat{\mathbf{v}}[k]_{k-1}}{\partial {}^{\mathcal{I}}\boldsymbol{\omega}_m} \\ \frac{\partial \hat{d}[k]_{k-1}}{\partial {}^{\mathcal{I}}\mathbf{f}_m} & \frac{\partial \hat{d}[k]_{k-1}}{\partial {}^{\mathcal{I}}\boldsymbol{\omega}_m} \end{bmatrix}, \quad \mathbf{V} \in \mathbb{R}^{4 \times 6} \\ &= \begin{bmatrix} T {}^{\mathcal{C}}\mathbf{R}_{\mathcal{I}} & T({}^{\mathcal{C}}\mathbf{R}_{\mathcal{I}}\mathbf{M} + [{}^{\mathcal{C}}\mathbf{v}]_{\times} {}^{\mathcal{C}}\mathbf{R}_{\mathcal{I}}) \\ \mathbf{0}_{1 \times 3} & \mathbf{0}_{1 \times 3} \end{bmatrix} \end{aligned} \quad (14)$$

$$\mathbf{M} = ({}^{\mathcal{I}}\boldsymbol{\omega}_m^T {}^{\mathcal{I}}\mathbf{p}_{\mathcal{IC}}) \mathbf{I}_3 + {}^{\mathcal{I}}\boldsymbol{\omega}_m {}^{\mathcal{I}}\mathbf{p}_{\mathcal{IC}}^T - 2 {}^{\mathcal{I}}\mathbf{p}_{\mathcal{IC}} {}^{\mathcal{I}}\boldsymbol{\omega}_m^T, \quad (15)$$

and $\text{cov}({}^{\mathcal{I}}\mathbf{f}_m) \in \mathbb{R}^{3 \times 3}$, $\text{cov}({}^{\mathcal{I}}\boldsymbol{\omega}_m) \in \mathbb{R}^{3 \times 3}$ are the covariance matrixes of the accelerometers/gyroscopes sensors in the IMU.

C. Update

Whenever a new scaled visual velocity estimate $z_m[k] = {}^c\tilde{v}_m = ({}^c\mathbf{v}[k]/d[k])_m$ becomes available from the optical flow decomposition, the predicted state $\hat{\mathbf{x}}[k]_{k-1}$ is updated to produce the estimated state $\hat{\mathbf{x}}[k]$. Let $\hat{z}[k]_{k-1}$ be the predicted scaled visual velocity estimation based on the predicted state $\hat{\mathbf{x}}[k]_{k-1}$

$$\hat{z}[k]_{k-1} = \frac{{}^c\hat{\mathbf{v}}[k]_{k-1}}{\hat{d}[k]_{k-1}}. \quad (16)$$

The kalman gain $\mathbf{K}[k] \in \mathbb{R}^{4 \times 3}$ is obtained as

$$\mathbf{K}[k] = \Sigma \mathbf{J}^T (\mathbf{J} \Sigma \mathbf{J}^T + \text{cov}(z_m))^{-1}, \quad (17)$$

where $\text{cov}(z_m) \in \mathbb{R}^{3 \times 3}$ is the covariance matrix of the scaled visual velocity measurement, and the Jacobian $\mathbf{J} \in \mathbb{R}^{3 \times 4}$, relating the predicted measurement $\hat{z}[k]_{k-1}$ to the predicted state $\hat{\mathbf{x}}[k]_{k-1}$, is given by

$$\begin{aligned} \mathbf{J} &= \begin{bmatrix} \frac{\partial \hat{z}[k]_{k-1}}{\partial {}^c\hat{\mathbf{v}}[k]_{k-1}} & \frac{\partial \hat{z}[k]_{k-1}}{\partial \hat{d}[k]_{k-1}} \end{bmatrix} \\ &= \begin{bmatrix} 1 & -\frac{{}^c\hat{\mathbf{v}}[k]_{k-1}}{\hat{d}^2[k]_{k-1}} \end{bmatrix}. \end{aligned} \quad (18)$$

Finally, the predicted state $\hat{\mathbf{x}}[k]_{k-1}$ is updated to the estimated state $\hat{\mathbf{x}}[k]$, together with matrix $\Sigma[k]$, as

$$\hat{\mathbf{x}}[k] = \begin{bmatrix} {}^c\hat{\mathbf{v}}[k] \\ \hat{d}[k] \end{bmatrix} = \begin{bmatrix} {}^c\hat{\mathbf{v}}[k]_{k-1} \\ \hat{d}[k]_{k-1} \end{bmatrix} + \mathbf{K}[k] (z_m[k] - \hat{z}[k]_{k-1}) \quad (19)$$

$$\Sigma[k] = (\mathbf{I}_4 - \mathbf{K}\mathbf{J})\Sigma[k]_{k-1}. \quad (20)$$

D. Discussion

We list here the quantities actually needed for implementing the proposed EKF. Apart from the estimated state $\hat{\mathbf{x}}[k]$, one needs knowledge of:

- 1 the constant IMU/camera rotation matrix ${}^{\mathcal{I}}\mathbf{R}_c$ and displacement vector ${}^{\mathcal{I}}\mathbf{p}_{IC}$;
- 2 the IMU angular velocity ${}^{\mathcal{I}}\boldsymbol{\omega}_m$;
- 3 the IMU linear acceleration ${}^{\mathcal{I}}\mathbf{a} = {}^{\mathcal{I}}\mathbf{f}_m + {}^{\mathcal{I}}\mathbf{g}$;
- 4 the plane normal ${}^c\mathbf{n}$;
- 5 the scaled camera linear velocity $z_m = ({}^c\mathbf{v}/d)_m$.

The quantities in item 1 are assumed to be known from a preliminary IMU/camera calibration phase while vector ${}^{\mathcal{I}}\boldsymbol{\omega}$ in item 2 is available directly from the IMU gyroscope readings.

Measurement of the linear acceleration ${}^{\mathcal{I}}\mathbf{a}$ in item 3 requires the specific acceleration ${}^{\mathcal{I}}\mathbf{f}_m$ (directly available through the IMU accelerometer readings) and knowledge of the gravity vector ${}^{\mathcal{I}}\mathbf{g}$ in IMU frame. An estimation of this latter quantity is also provided by standard IMUs in *near-hovering conditions*, a fact largely exploited when recovering the UAV attitude from onboard sensing, see, e.g., [18]. Alternatively, when the vehicle is undergoing large accelerations one can, for instance, obtain ${}^{\mathcal{I}}\mathbf{g}$ by assuming a *horizontal planar scene* (as often the case) and by exploiting knowledge of the plane normal ${}^c\mathbf{n}$ recovered by decomposing the homography matrix \mathbf{H} (item 4) [10].

Finally, vector z_m in item 5 is directly retrieved from the optical flow decomposition described in Sec. II.

V. SCALE ESTIMATION BASED ON A NONLINEAR OBSERVER

In this Section, we detail the derivations of an alternative *nonlinear observer* for retrieving the plane distance d based on the theory of Persistency of Excitation (PE) in the context of adaptive control [12]. The benefits of this estimation scheme w.r.t. the previous EKF lie in its cleaner design (it does not require any linearization or approximation as implicit in any EKF), easier interpretation of its convergence properties, and easiness of tuning (less parameters to be fixed). However, as opposed to the EKF, the design assumes a deterministic dynamics, and therefore does not take explicitly into account the noise inherent into the system (state transition and measurement). For the sake of exposition, the developments are here formulated in continuous time.

We start by recalling the Persistency of Excitation Lemma [12] upon which the next developments are based.

Lemma 1: Consider the linear time-varying system

$$\begin{cases} \dot{\boldsymbol{\xi}} = \mathbf{W}\boldsymbol{\xi} + \boldsymbol{\Omega}^T(t)\mathbf{z}, & \boldsymbol{\xi} \in \mathbb{R}^n \\ \dot{\mathbf{z}} = -\boldsymbol{\Lambda}\boldsymbol{\Omega}(t)\mathbf{S}\boldsymbol{\xi}, & \mathbf{z} \in \mathbb{R}^p \end{cases} \quad (21)$$

where \mathbf{W} is an Hurwitz matrix, \mathbf{S} is an $n \times n$ symmetric positive definite matrix such that $\mathbf{W}^T\mathbf{S} + \mathbf{S}\mathbf{W} = -\mathbf{Q}$, with \mathbf{Q} symmetric positive definite, and $\boldsymbol{\Lambda}$ is a $p \times p$ symmetric positive definite matrix. If $\|\boldsymbol{\Omega}(t)\|$, $\|\dot{\boldsymbol{\Omega}}(t)\|$ are uniformly bounded and the *persistency of excitation* condition is satisfied, i.e., there exist two positive real numbers T and γ such that

$$\int_t^{t+T} \boldsymbol{\Omega}(\tau)\boldsymbol{\Omega}^T(\tau)d\tau \geq \gamma\mathbf{I} > 0, \quad \forall t \geq t_0, \quad (22)$$

then $(\boldsymbol{\xi}, \mathbf{z}) = (0, 0)$ is a globally exponentially stable equilibrium point of system (21).

In the context of range estimation from vision, extensions of the PE theory have been successfully applied to recover the depth of feature points from known camera motion in [19], [20], and the plane normal and distance by processing image moments in [17]. Roughly speaking, the PE Lemma can be exploited as follows: assume a vector $\mathbf{x} = [\mathbf{x}_1^T \ \mathbf{x}_2^T]^T \in \mathbb{R}^{n+p}$ can be split into a *measurable* component \mathbf{x}_1 and an *unmeasurable* component \mathbf{x}_2 . Defining an estimation vector $\hat{\mathbf{x}} = [\hat{\mathbf{x}}_1^T \ \hat{\mathbf{x}}_2^T]^T \in \mathbb{R}^{n+p}$, and the corresponding estimation error $\mathbf{e} = [\boldsymbol{\xi}^T \ \mathbf{z}^T]^T = [\mathbf{x}_1^T - \hat{\mathbf{x}}_1^T \ \mathbf{x}_2^T - \hat{\mathbf{x}}_2^T]^T$, the goal is to design an update rule for $\hat{\mathbf{x}}$ such that the closed-loop error dynamics matches formulation (21). When this manipulation is possible, Lemma 1 ensures global exponential convergence to 0 of the estimation error $\mathbf{e} = [\boldsymbol{\xi}^T \ \mathbf{z}^T]^T$, thus allowing to infer the unmeasurable value of \mathbf{x}_2 from knowledge of \mathbf{x}_1 . The PE condition (22) plays the role of an observability constraint: estimation of \mathbf{x}_2 is possible iff matrix $\boldsymbol{\Omega}(t) \in \mathbb{R}^{p \times n}$ is *sufficiently exciting* over time in the sense of (22). We finally note that, being $\boldsymbol{\Omega}(t)$ a generic *time-varying* quantity, the formulation (21) is not restricted to only span the class of linear systems, but it

can easily accommodate nonlinear terms as long as they are embedded in matrix $\Omega(t)$.

We now detail how to tailor (21) to the case under consideration. We start by defining $x_2 = 1/d$ and $x_1 = {}^c\tilde{v} = {}^c\mathbf{v}/d = {}^c\mathbf{v}x_2$ with, therefore, $n = 3$ and $p = 1$. Exploiting (6), the dynamics of x_2 is given by

$$\dot{x}_2 = -\frac{\dot{d}}{d^2} = -\frac{{}^c\mathbf{v}^T {}^c\mathbf{n}}{d^2} = -\frac{{}^c\tilde{v}^T {}^c\mathbf{n}}{d} = -x_2 x_1^T {}^c\mathbf{n}. \quad (23)$$

As for the dynamics of x_1 , using (2) we have

$$\begin{aligned} \dot{x}_1 &= {}^c\dot{\mathbf{v}}x_2 + {}^c\mathbf{v}\dot{x}_2 \\ &= {}^c\dot{\mathbf{v}}x_2 - {}^c\mathbf{v}x_2x_1^T {}^c\mathbf{n} \\ &= {}^c\dot{\mathbf{v}}x_2 - x_1x_1^T {}^c\mathbf{n} \\ &= ({}^c\mathbf{R}_I^T \mathbf{a} + [{}^c\boldsymbol{\omega}]_{\times}^2 {}^c\mathbf{p}_{IC} + [{}^c\dot{\boldsymbol{\omega}}]_{\times} {}^c\mathbf{p}_{IC})x_2 \\ &\quad - [{}^c\boldsymbol{\omega}]_{\times} x_1 - x_1x_1^T {}^c\mathbf{n} \\ &= \Omega^T(t)x_2 - [{}^c\boldsymbol{\omega}]_{\times} x_1 - x_1x_1^T {}^c\mathbf{n} \end{aligned} \quad (24)$$

with

$$\Omega^T(t) = {}^c\mathbf{R}_I^T \mathbf{a} + [{}^c\boldsymbol{\omega}]_{\times}^2 {}^c\mathbf{p}_{IC} + [{}^c\dot{\boldsymbol{\omega}}]_{\times} {}^c\mathbf{p}_{IC}. \quad (25)$$

We can then design the update rule for the estimated state \hat{x} as

$$\dot{\hat{x}}_1 = \Omega^T(t)\hat{x}_2 - [{}^c\boldsymbol{\omega}]_{\times} \hat{x}_1 - \hat{x}_1\hat{x}_1^T {}^c\mathbf{n} + \mathbf{K}_1 \boldsymbol{\xi} \quad (26)$$

$$\dot{\hat{x}}_2 = -\hat{x}_2\hat{x}_1^T {}^c\mathbf{n} + \mathbf{K}_2 \Omega(t)\boldsymbol{\xi}. \quad (27)$$

where $\mathbf{K}_1 > 0$ and $\mathbf{K}_2 > 0$ are symmetric and positive definite gain matrixes. With this choice, the dynamics of the estimation error $e = [\boldsymbol{\xi}^T \ z^T]^T$ becomes

$$\dot{\boldsymbol{\xi}} = -\mathbf{K}_1 \boldsymbol{\xi} + \Omega^T(t)z \quad (28)$$

$$\dot{z} = -\mathbf{K}_2 \Omega(t)\boldsymbol{\xi} - z\hat{x}_1^T {}^c\mathbf{n}. \quad (29)$$

It is easy to verify that, by letting $\mathbf{W} = -\mathbf{K}_1$, $\mathbf{A} = \mathbf{K}_2$ and $\mathbf{S} = \mathbf{I}_3$, the formulation (21) is almost fully recovered apart from the spurious scalar term $g(e, t) = -z\hat{x}_1^T {}^c\mathbf{n}$ in (29). Nevertheless, exponential convergence of the estimation error $e(t)$ to 0 can still be proven by resorting to Lyapunov theory and by noting that the spurious term $g(e, t)$ is a *vanishing* perturbation of an otherwise globally exponentially stable nominal system. We refer the reader to [20], [17] for an explicit proof of these facts.

We note that the design of observer (26–27) did not require any linearization step as for the previous EKF thanks to the more general class of (nonlinear) systems spanned by formulation (21). Instrumental, in this sense, is the choice of considering, as state variable x_2 , the *inverse* of the plane distance d : this manipulation allows to obtain linearity of the state equations in x_2 , thus ultimately making it possible to apply the PE theory to the case under consideration. Similar *inverse parameterizations* for the scene scale can also be found in other works dealing with the issue of range estimation from moving cameras, see, e.g., [20], [21].

It is also worth analyzing, in our specific case, the meaning of the PE condition (22) necessary for obtaining a converging estimation. Being $\Omega^T(t) \in \mathbb{R}^3$ a vector, condition (22)

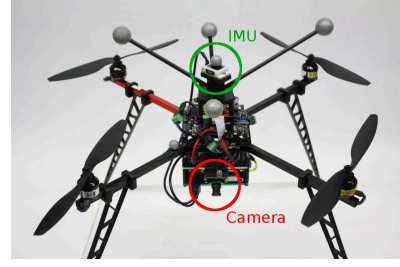


Fig. 2: Experimental setup with the highlighted location of IMU and camera. The x -axis of the body frame is oriented along the red metal beam of the frame.

requires that the norm of $\Omega^T(t)$ (i.e., at least one component) does not ultimately vanish over time. On the other hand, vector $\Omega^T(t)$ represents the camera linear acceleration through space w.r.t. the inertial world frame \mathcal{W} . Therefore, we recover the well-known condition that the estimation of d is possible if and only if the camera undergoes a physical acceleration, and, consequently, moving at constant velocity w.r.t. \mathcal{W} cannot allow the estimation to converge.

As done in the previous Sect. IV-D, we finally list the quantities necessary for implementing the proposed observer (26–27). In addition to the estimated state \hat{x} , these are:

- 1 the constant IMU/camera rotation matrix ${}^I\mathbf{R}_C$ and displacement vector ${}^I\mathbf{p}_{IC}$;
- 2 the IMU angular velocity ${}^I\boldsymbol{\omega}_m$;
- 3 the IMU angular acceleration ${}^I\dot{\boldsymbol{\omega}}$;
- 4 the IMU linear acceleration ${}^I\mathbf{a} = {}^I\mathbf{f}_m + {}^I\mathbf{g}$;
- 5 the plane normal ${}^c\mathbf{n}$;
- 6 the scaled camera linear velocity $x_1 = {}^c\tilde{v}_m = ({}^c\mathbf{v}/d)_m$.

Thus, the same considerations of Sect. IV-D apply also to the PE estimation scheme (26–27). In particular, as done for the EKF, in our implementation we will approximate ${}^I\dot{\boldsymbol{\omega}} \approx \mathbf{0}$ in (25) (item 3) because of lack of a direct measurement of the UAV angular acceleration.

VI. EXPERIMENTS

A. Experimental Setup

For our experiments, we used a quadrotor UAV from MikroKopter that was interfaced using the free *TeleKyb* software framework [22]. The location of all relevant sensors and frames can be found in Figs. 1–2. The quadrotor was equipped with an additional 3DM-GX3-25 IMU from MicroStrain (frame \mathcal{I}) to provide the measurements of the specific acceleration ${}^I\mathbf{f}_m$, of the angular velocity ${}^I\boldsymbol{\omega}_m$, and of the gravity vector ${}^I\mathbf{g}_m$ at 200 Hz. A precalibrated MatrixVision mvBlueFox camera (frame \mathcal{C}) captured the necessary image stream. The body frame \mathcal{B} was fixed at the center of the quadrotor metal frame. To obtain a reliable ground truth, a Vicon tracking system with sub-millimeter accuracy was used throughout our experiments.

B. Simulations

In order to illustrate the convergence of the proposed estimation schemes, we generated a synthetic acceleration

profile together with the corresponding (simulated) sensor readings. This resulted in a camera motion similar to the circular trajectory presented in the experimental results of the next Sec. VI-C. All generated sensor readings were perturbed with an additive zero mean gaussian noise with covariance matrixes taken from the real sensor characteristics: $0.00004 \mathbf{I}_3 \frac{m}{s^2}$, $0.00002 \mathbf{I}_3 \frac{rad}{s}$ and $0.00001 \mathbf{I}_3 \frac{1}{s}$ for $\mathcal{I}\mathbf{f}_m$, $\mathcal{I}\boldsymbol{\omega}_m$, and for the scaled linear velocity from optical flow $(^c\mathbf{v}/d)_m$, respectively. The same covariance matrixes were employed in the steps (13)–(17) of the EKF, while the gains of the nonlinear observer $\mathbf{K}_1, \mathbf{K}_2$ were manually tuned to $\mathbf{K}_1 = 10$ and $\mathbf{K}_2 = 70$.

To demonstrate the robustness to unknown initial estimates, the two filters were initialized with $\hat{d}(t_0) = -5 m$ and $^c\hat{\mathbf{v}}(t_0) = [0 \ 0 \ 0]^T m/s$, while the actual initial distance from the scene was $d(t_0) = -1 m$.

C. Recorded Data

To allow for a controlled and direct comparison of the two systems, we recorded data while flying along a circular trajectory of $2 m$ in diameter. The trajectory was chosen in order to have the UAV accelerating sinusoidally along the three cartesian directions. It was traveled once in about $10 s$, with a maximum speed of $0.6 \frac{m}{s}$. The height varied from $0.5 m$ to $1.5 m$ along the trajectory. The quadrotor relied on the Vicon tracking system in order to track the trajectory with a standard flight controller. On-board hardware was used to record vision and IMU data during flight. Afterwards, the two scale estimation approaches were tested offline by processing the collected data sets and by comparing the results against the Vicon ground truth.

Sensor offsets were calibrated before takeoff, and the covariance matrixes of $\mathcal{I}\mathbf{f}_m$, $\mathcal{I}\boldsymbol{\omega}_m$ and $(^c\mathbf{v}/d)_m$ were estimated over a period of $60 s$. Both filters were initialized close to the real initial state with $\hat{d}(t_0) = -1 m$ and $^c\hat{\mathbf{v}}(t_0) = [0 \ 0 \ 0]^T \frac{m}{s}$. The gains for the nonlinear observer were tuned to $\mathbf{K}_1 = 10$ and $\mathbf{K}_2 = 6$.

D. Comparison with Previous Work

To allow for a direct comparison with the approach presented in [15], we tried to reproduce a similar experiment consisting of small sinusoidal hand held motions. This resulted in a trajectory with an amplitude of about $0.1 \frac{m}{s}$ and a frequency of about $1 Hz$ at the height of $0.5 m$.

VII. RESULTS AND DISCUSSION

A. Simulations results

Figure 3a presents the results of the EKF and the nonlinear observer in estimating the distance d over time. By properly tuning the (two) gains of the PE observer, a faster convergence of the estimation error can be obtained w.r.t. the ‘fully-informed’ EKF, i.e., relying on the exact knowledge of the noise characteristics. This is also evident in Fig. 3b where the estimation error is explicitly shown. The nonlinear observer was able to compensate for the initial offset within $12 s$ while the EKF needed $27 s$. Measured after convergence, the EKF reaches a RMS error of $0.0042 m$ while we found a RMS

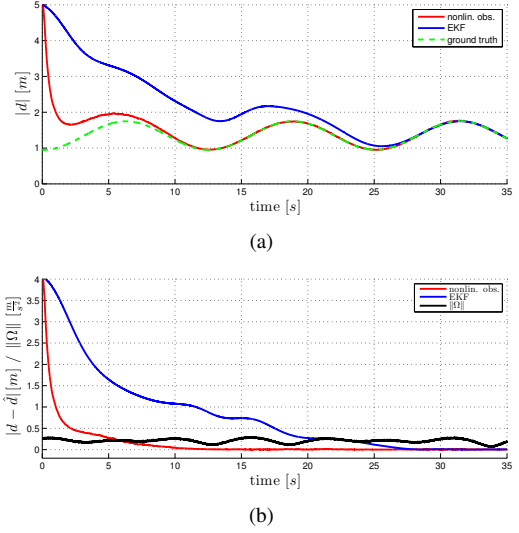


Fig. 3: Estimated plane distance d using a simulated dataset. (a) Estimated d and (b) the estimation error $d - \hat{d}$ for the two tested estimation approaches. For completeness, the norm of $\|\Omega(t)\|$ as defined in (25) is shown as well.

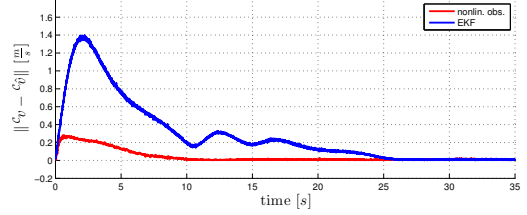


Fig. 5: Error of the linear velocity estimation for the two approaches w.r.t. the Vicon ground truth.

error of $0.0057 m$ for the nonlinear observer. As discussed in Sec. V, the norm of vector $\Omega(t)$ in (25) plays the role of an observability condition in affecting the filter convergence over time. This fact can be appreciated in Fig. 3b: phases with a small convergence speed correspond to relatively small values of $\|\Omega(t)\|$.

After estimating \hat{d} , one can retrieve the actual camera linear velocity from the optical flow measurement as $^c\hat{\mathbf{v}} = (^c\mathbf{v}/d)_m \hat{d}$. The resulting estimated linear velocity is shown in Fig. 4 while the estimation error relative to ground truth is presented in Fig. 5. The convergence behavior is similar compared to the estimation of d . From these plots we can conclude that, with the employed noise characteristics and gains, in a real scenario large velocities should be avoided until convergence is reached, and this process could be quickened by alternating acceleration phases. Again, after $30 s$, we computed the RMS error. For the EKF and the nonlinear observer, we found an error of $0.008 \frac{m}{s}$ and $0.010 \frac{m}{s}$ respectively.

B. Recorded Data

The picture is slightly different in the case of real sensory data, with the filters being initialized close to the real states. Figure 6a shows the estimated distance \hat{d} . Here, in presence

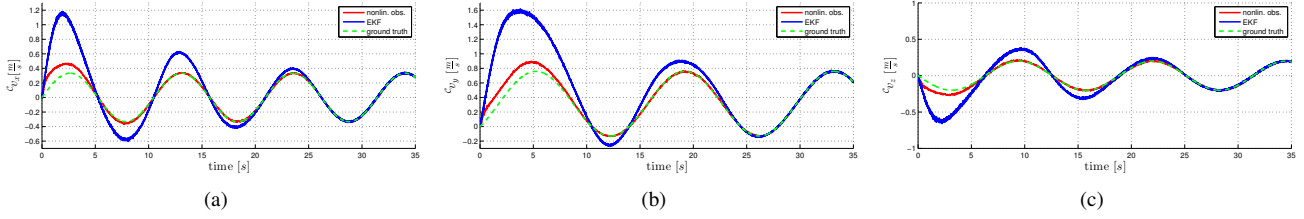


Fig. 4: Linear velocity estimate on simulated data in the (a) x , (b) y and (c) z axis.

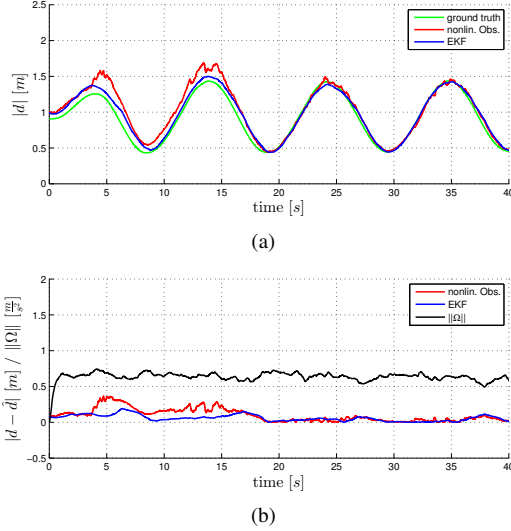


Fig. 6: Distance of the camera from the ground d as (a) estimated from a recorded dataset and (b) the corresponding estimation error compared to the ground truth as obtained from a Vicon tracking system. Additionally, the quantity $\|\Omega(t)\|$ is also shown.

of real noise, the EKF produces a smoother output compared to the nonlinear observer, for which noise corrupts the estimation error for the first 15 s as plotted in Fig. 6b. In fact, the noise naturally increases with the height of the UAV as the signal to noise ratio for the visual velocity estimate decreases due to the reduced apparent velocity. In our experiments, the EKF was able to cope with this noise more robustly than the nonlinear observer. Over the last 10 s however, we found a RMS error of approximately 0.03 m for the scale factor d with both approaches. As expected for a circular trajectory, $\|\Omega(t)\|$ kept a constant non-zero value over time, thus providing the two estimators with an adequate excitation level in the measured signals.

Similarly, the estimated metric linear velocities presented in Fig. 7 show an acceptable error for the control of a quadrotor right from the beginning in the case of the EKF. The nonlinear observer yielded a reliable output after 17 s as well.

C. Comparison with Previous Work

The results shown in Fig. 8 allow for a comparison of the PE estimator to the solution presented in [15]. By exploiting the PE scheme along a similar trajectory as the ones used in [15], we could obtain RMS values of

$[0.0074, 0.0095, 0.0114] \frac{m}{s}$ for the velocity estimation error in the three Cartesian directions, respectively. This yields an improvement of $[3.8, 3.7, 2.2]$ times w.r.t. what reported in [15]. Although the experimental conditions are obviously different, we believe these results still indicate the good potential of the proposed PE observer in dealing with scale estimation from vision.

As for our proposed EKF, we found a RMS error of $[0.0101, 0.0141, 0.0107] \frac{m}{s}$ respectively, which still corresponds to an average improvement of 2.5 times compared to the EKF used in [15]. This could be ascribed to the more accurate modeling of the system dynamics, including an explicit expression for \dot{d} , or to the choice of employing the continuous homography constraint instead of the epipolar one when dealing with a mostly planar scene.

VIII. CONCLUSIONS AND FUTURE WORK

A. Conclusions

In this paper, we stressed the need for reliable visual velocity estimation systems on UAVs to overcome the boundaries of protected lab environments. In particular, we proposed a solution based on direct optical flow decomposition in order to be highly independent from maps, known landmarks, or the need of extended tracking over time. We focused our attention on the estimation of the metric scale from the fusion of the scaled visual velocity obtained from optical flow with the high frequency readings of an onboard IMU. To this end, we discussed two estimation schemes, the former based on a classical EKF, and the latter proposing a novel PE nonlinear observer. Simulated and real experimental data were used to assess and compare the performance of both filters in ideal and real conditions. Compared to other state of the art solutions, we could obtain an improved estimation accuracy of about 3.5 times.

We found the PE nonlinear observer to yield good results in presence of well-characterized noise and poor knowledge of the actual initial state of the system. However, when it is more important to cope with various sources of complex noise rather than to achieve a fast convergence, the EKF might be the better choice.

B. Future Work

Currently, we are working towards an implementation of both scale estimation approaches in closed-loop control using onboard hardware only. In this scenario, the user will send a velocity command by means of wireless communication to the otherwise fully autonomously acting quadrotor system.

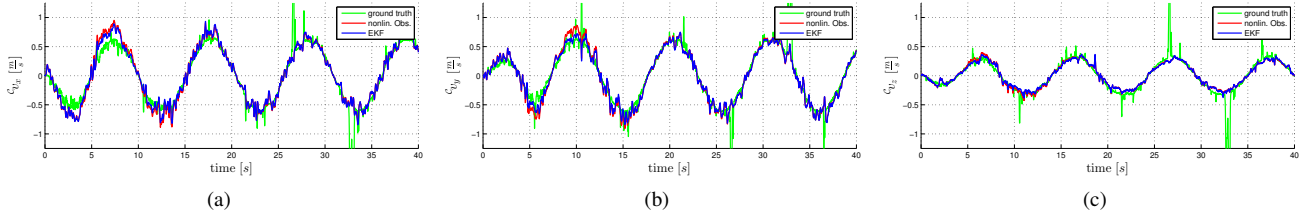


Fig. 7: Recovered linear velocity from a recorded dataset in the (a) x , (b) y and (c) z directions. The ground truth was obtained by numerically differentiating the Vicon position information.

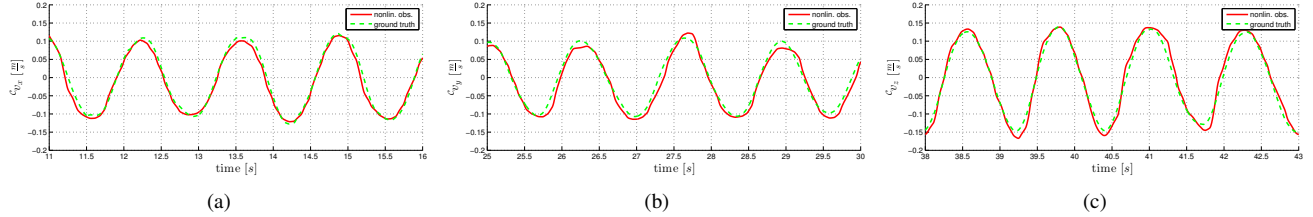


Fig. 8: Experimental results for the comparison with [15]. Results for the (a) x , (b) y and z directions are plotted against a ground truth obtained from the Vicon tracking system. We found a RMS value of $[0.0074, 0.0095, 0.0114] \frac{m}{s}$ for the Cartesian directions, respectively.

IX. ACKNOWLEDGMENTS

The authors like to thank Dr. Antonio Franchi and Martin Riedel for their valuable suggestions and contribution on the development of the underlying software framework.

REFERENCES

- [1] EU Collaborative Project ICT-248669, “AIRobots,” www.airobots.eu.
- [2] EU Collaborative Project ICT-287617, “ARCAS,” www.arcas-project.eu.
- [3] Y. Ma, S. Soatto, J. Kosecka, and S. S. Sastry, *An Invitation to 3-D Vision*. Springer, 2004.
- [4] K. E. Wenzel, A. Masselli, and A. Zell, “Automatic Take Off, Tracking and Landing of a Miniature UAV on a Moving Carrier Vehicle,” *Journal of Intelligent & Robotic Systems*, vol. 61, no. 1, pp. 221–238, 2010.
- [5] G. Klein and D. Murray, “Parallel Tracking and Mapping for Small AR Workspaces,” in *Proceedings of the International Symposium on Mixed and Augmented Reality*, Nara, Japan, Nov. 2007, pp. 225–234.
- [6] G. Nützi, S. Weiss, D. Scaramuzza, and R. Siegwart, “Fusion of IMU and Vision for Absolute Scale Estimation in Monocular SLAM,” in *Proceedings of the International Conference on Unmanned Aerial Vehicles*, Dubai, United Arab Emirates, 2010.
- [7] S. Weiss, M. W. Achtelik, M. Chli, and R. Siegwart, “Versatile distributed pose estimation and sensor self-calibration for an autonomous MAV,” in *Proceedings of the International Conference on Robotics and Automation*, Saint Paul, MN, USA, May 2012, pp. 31–38.
- [8] L. Kneip, A. Martinelli, S. Weiss, D. Scaramuzza, and R. Siegwart, “Closed-Form Solution for Absolute Scale Velocity Determination Combining Inertial Measurements and a Single Feature Correspondence,” in *Proceedings of the International Conference on Robotics and Automation*, Shanghai, China, 2011.
- [9] A. Martinelli, “Vision and IMU Data Fusion : Closed-Form Solutions for Attitude, Speed, Absolute Scale, and Bias Determination,” *Transactions on Robotics*, vol. 28, no. 1, pp. 44–60, 2012.
- [10] V. Grabe, Heinrich H. Bühlhoff, and P. R. Giordano, “On-board Velocity Estimation and Closed-loop Control of a Quadrotor UAV based on Optical Flow,” in *Proceedings of the International Conference on Robotics and Automation*, St. Paul, MN, USA, 2012.
- [11] V. Grabe, H. H. Bühlhoff, and P. Robuffo Giordano, “Robust Optical-Flow Based Self-Motion Estimation for a Quadrotor UAV,” in *Proceedings of the International Conference on Intelligent Robots and Systems*, Vilamoura, Portugal, 2012.
- [12] R. Marino and P. Tomei, *Nonlinear Control Design: Geometric, Adaptive and Robust*. Prentice Hall, 1995.
- [13] B. Herissé, T. Hamel, R. Mahony, and F.-X. Russotto, “Landing a VTOL Unmanned Aerial Vehicle on a Moving Platform Using Optical Flow,” *Transactions on Robotics*, vol. 28, no. 1, pp. 77–89, Feb. 2012.
- [14] D. Honegger, L. Meier, P. Tanskanen, and M. Pollefeys, “An Open Source and Open Hardware Embedded Metric Optical Flow CMOS Camera for Indoor and Outdoor Applications,” in *Proceedings of the International Conference on Robotics and Automation*, Karlsruhe, Germany, 2013.
- [15] S. Weiss, M. W. Achtelik, S. Lynen, M. Chli, and R. Siegwart, “Real-time Onboard Visual-Inertial State Estimation and Self-Calibration of MAVs in Unknown Environments,” in *Proceedings of the International Conference on Robotics and Automation*, Saint Paul, MN, USA, 2012, pp. 957–964.
- [16] Y. Ma, J. Kosecka, and S. Sastry, “Linear Differential Algorithm for Motion Recovery: A Geometric Approach,” *International Journal of Computer Vision*, vol. 36, no. 1, pp. 71–89, 2000.
- [17] P. Robuffo Giordano, A. De Luca, and G. Oriolo, “3D Structure Identification from Image Moments,” in *Proceedings of the International Conference on Robotics and Automation*, 2008, pp. 93–100.
- [18] R. Mahony, T. Hamel, and J.-M. Pfimlin, “Nonlinear Complementary Filters on the Special Orthogonal Group,” *Transactions on Automatic Control*, vol. 53, no. 5, pp. 1203–1218, 2008.
- [19] A. De Luca, G. Oriolo, and P. Robuffo Giordano, “On-Line Estimation of Feature Depth for Image-Based Visual Servoing Schemes,” in *Proceedings of the International Conference on Robotics and Automation*, 2007, pp. 2823–2828.
- [20] —, “Feature Depth Observation for Image-based Visual Servoing: Theory and Experiments,” *International Journal of Robotics Research*, vol. 27, no. 10, pp. 1093–1116, 2008.
- [21] J. Civera, A. J. Davison, and J. Montiel, “Inverse Depth Parametrization for Monocular SLAM,” *Transactions on Robotics*, vol. 24, no. 5, pp. 932–945, 2008.
- [22] V. Grabe, M. Riedel, Heinrich H. Bühlhoff, P. Robuffo Giordano, and A. Franchi, “The TeleKyb Framework for a Modular and Extendible ROS-based Quadrotor Control,” in *Proceedings of the European Conference on Mobile Robots*, Barcelona, Spain, 2013.