Article

TRANSACTIONS OF THE INSTITUTE OF MEASUREMENT & CONTROL

Transactions of the Institute of Measurement and Control
1–13
© The Author(s) 2018
Reprints and permissions:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/0142331217747343
journals.sagepub.com/home/tim
SAGE

# Minimalistic approach for monocular SLAM system applied to micro aerial vehicles in GPS-denied environments

Sarquis Urzua[1], Rodrigo Munguía[1], Emmanuel Nuño[1] and Antoni Grau[2]

**Abstract**

In this work, a novel monocular simultaneous localization and mapping (SLAM) system with application to micro aerial vehicles is proposed. The main difference with respect to previous approaches is that a barometer is used as a unique sensory aid for incorporating altitude information into the system in order to recover an absolute metric scale. First, an observability analysis of a simplified model of a monocular SLAM system is developed. From this analysis, several theoretical results are derived. Among others, one important result is related to the fact that the metric scale can become observable when measurements of altitude are included in the system. In this case, sufficient conditions for observability are presented. The design of the proposed method is based on these theoretical results. Simulations and experiments with real data are presented to validate the proposed approach. The results confirm that the metric scale can be retrieved by including altitude measurements in the system. It is also shown that the proposed method can be practically implemented, using low-cost sensors, to perform visual-based navigation in GPS-denied environments.

**Keywords**

Monocular SLAM, barometer-aided, micro aerial vehicles, GPS-denied, visual-based navigation

## Introduction

State estimation of vehicle position is a fundamental necessity for any application involving autonomous micro aerial vehicles. Monocular simultaneous localization and mapping (SLAM) deals with the way in which a mobile robot can operate in an a-priori unknown environment by using an onboard monocular camera to simultaneously build a map of its surroundings, which is used to track its position. Some examples of SLAM systems are those of Dehghan and Moradi (2016) and Ihemadu et al. (2015). In the case of micro aerial vehicles, monocular SLAM techniques represent an excellent alternative, especially owing to several limitations regarding the design of the platform, mobility and payload capacity that impose considerable restrictions on the available computational and sensing resources.

In particular, and compared with other kinds of visual configuration (e.g. stereo vision), the use of monocular vision has some advantages in terms of weight, space, power efficiency or scalability. For example, in stereo rigs, the fixed base-line between cameras can limit the operation range. Conversely, the use of monocular vision introduces some technical challenges. First, depth information cannot be retrieved in a single frame; hence, robust techniques for recovering the depth of the features are required. Another challenging aspect of working with monocular sensors has to do with the impossibility of directly recovering a metric scale of the world. If no additional information is used, and a single camera is used as the sole source of data to the system, only the map and trajectory can be recovered, without metric information (Davison et al., 2007).

In this work, this latter issue is addressed in the context of monocular SLAM navigation for micro aerial vehicles. A common approach for setting the metric scale in some early monocular SLAM methods was to use a pattern with known dimensions at the initialization stage (e.g. Davison, 2003; Eade and Drummond, 2006; Munguia and Grau, 2007). With the subsequent use of monocular SLAM methods in different kinds of robotics application, such as micro aerial vehicles, the necessity of using better approaches to incorporate metric information into the system has become more evident. Also, there are other kinds of robotic application where the problem of the metric scale is involved. For instance, Zhuang et al. (2015) propose a metric scale coordination technique for solving the problem of variable object scales in 3D object detection.

[1] Department of Computer Science, CUCEI, University of Guadalajara, México
[2] Department of Automatic Control, Technical University of Catalonia UPC, Spain

**Corresponding author:**
Rodrigo Munguía, Department of Computer Science, CUCEI, University of Guadalajara, Blvd. Marcelino Garca Barragn 1421, C.P. 44430, Guadalajara, México.
Email: rodrigo.munguia@upc.edu

### Related work

In the case of monocular SLAM with application to aerial vehicles, different approaches have been followed for recovering the metric scale of the world. Mirzaei and Roumeliotis (2008) retrieved the monocular scale factor from a feature pattern with known dimensions. In experiments presented in Forster et al. (2013) and Weiss et al. (2011), the map is initially set by hand, by aligning first estimates with the ground-truth to determine the scale of the environment. Celik and Somani (2013) addressed the problem of scale recovery for environments formed by corridors, such as those commonly found in office buildings. In this case, several assumptions are made about the structure of the environment, such as the flatness of the floor. Also, it is assumed that the relative altitude of the micro aerial vehicle from the floor is determined by using an ultrasonic range sensor, as is the distance from the micro aerial vehicle to the wall of the corridor. Another approach for recovering the metric scale consists of integrating inertial measurements from an inertial measurement unit (accelerometer or gyroscope). In particular, Nützi et al. (2011) explicitly considered the scale in the system state, which was estimated through an extended Kalman filter (EKF). The filter makes use of an innovation error defined by the difference between the unscaled acceleration (obtained from monocular vision) and the measured acceleration in the vertical axis (obtained from an inertial measurement unit). Wang et al. (2014) also follow the same approach. The potential problem with this approach is that the acceleration obtained from an inertial measurement unit has a dynamic bias that is difficult to estimate. This bias introduces, at the same time, a bias in the estimated scale. Also, in this kind of setup, a precise calibration for the alignment of the camera and the inertial measurement unit is required. Chowdhary et al. (2013) propose an EKF-based method to fuse measurements from inertial sensors, a monocular downward-facing camera and a range sensor (sonar altimeter). In this approach, it is assumed that landmarks lie on a plane (flat-terrain assumption). Because the range sensor is aligned with the camera, it is assumed that the range readings provide reliable information about the depth of the features. The metric scale is recovered through the range and bearing measurements. In this sense, this method can be considered as being like a range-and-bearing SLAM scheme, instead of a truly bearing-only (monocular) SLAM system. However, the efficacy of this method depends on the assumption of a flat terrain, and is also limited by the operation range of the sonar.

### Objectives and contributions

In this work, a novel monocular SLAM system with application to a micro aerial vehicle is proposed. The main difference with respect to previous approaches is that a barometer is used as a unique sensory aid for incorporating altitude information in the system to recover an absolute metric scale.

An observability analysis for this kind of system was developed. From this analysis, several theoretical results were derived. In this sense, one central contribution of this work is to show that, under certain conditions, the incorporation of altitude measurements in the system is sufficient to recover the metric scale in estimates.

The design of the proposed method is based on these results. The proposed method is useful for performing visual-based navigation in fully GPS-denied environments or as a backup system in periods where a GPS signal is not available.

To the best of our knowledge, this is the first monocular SLAM approach that relies on this minimalistic barometer–camera setup. Some of the main features that highlight the applicability of the proposed method are as follows:

- It differs from such methods as those of Celik and Somani (2013) or Chowdhary et al. (2013); in the proposed method, there is no need to make assumptions about the structure of the environment or about the limitations regarding the operation range of a sonar device.
- Compared with the methods of Nützi et al. (2011) and Wang et al. (2014), in the proposed method there is no need for an extensive pre-calibration routine for aligning the inertial measurement unit and the camera.
- According to Hopkins et al. (2010), the dynamic error bias of a barometer is smaller than the bias of an accelerometer. Actually, the barometer is commonly used as an augmentation sensor to limit errors in inertial navigation systems.
- The architecture of the proposed method is based on a well-known methodology, EKF-SLAM.

## Problem description

For the sake of clarity, the problem to be addressed in this work is introduced in a simplified two-degrees-of-freedom context. However, it is important to note that this simplification is still representative of all the major aspects of the full problem. Later, the proposal will be extended to the six-degrees-of-freedom problem. Let us consider the following unconstrained model, $\dot{x}_r = f(x_r, u)$, of a camera $C_s$ attached to a micro aerial vehicle (see Figure 1)

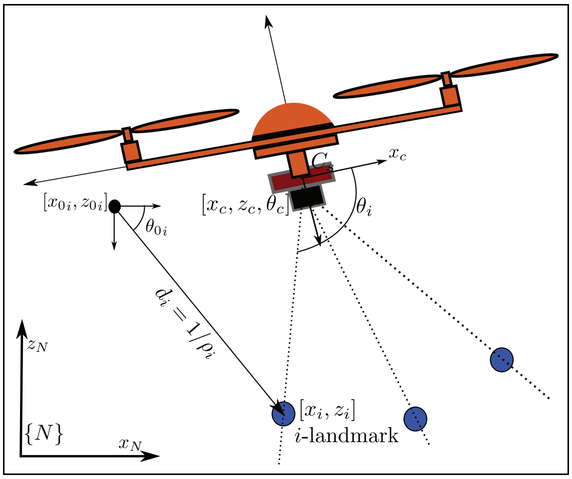

**Figure 1.** System parametrization.

$$\begin{aligned} \dot{x}_c = v_x \quad \dot{z}_c = v_z \quad \dot{\theta}_c = \omega_c \\ \dot{v}_x = V_x \quad \dot{v}_z = V_z \quad \dot{\omega}_c = \Omega \end{aligned} \tag{1}$$

where $x_r = [x_c, z_c, \theta_c, v_x, v_z, \omega_c]^{\mathrm{T}}$ is the vector state of camera $C_s$. Let $[x_c, z_c, \theta_c]$ represent the position and orientation of the camera, and $[v_x, v_z, \omega_c]$ their first derivatives. In this model, an unknown input $u = [V_x, V_z, \Omega]^{\mathrm{T}}$ of linear and angular accelerations is assumed, with zero-mean and known-covariance Gaussian processes. Also it is assumed that the camera $C_s$ is capable of detecting and tracking 2D feature points. The measurement process is modelled by equations of the form

$$\begin{aligned} h_{\theta i}(x) &= \operatorname{arctan2}\left(\frac{z_c - z_i}{x_c - x_i}\right) - \theta_c \\ x_i &= (1/\rho_i)\cos(\theta_{0i}) + x_{0i} \\ z_i &= (1/\rho_i)\sin(\theta_{0i}) + z_{0i} \end{aligned} \tag{2}$$

where $[x_i, z_i]$ is the Euclidean position of an $i$th feature coded by its inverse form. The state of an $i$th feature $y_i$ is defined by $y_i = [x_{0i}, z_{0i}, \theta_{0i}, \rho_i]^{\mathrm{T}}$, where $[x_{0i}, z_{0i}]$ is the position of the camera $C_s$ when the feature was first detected, $\theta_{0i}$ is the first bearing measurement and $\rho_i = 1/d_i$ is the inverse of the feature depth $d_i$ (see Figure 1).

The system state $x$, to be estimated, is composed by the camera state $x_r$ and is augmented with the state $y_i$ of every feature contained in the map

$$x = [x_r, y_1, y_2, \ldots, y_n]^{\mathrm{T}} \tag{3}$$

If no other metric information is included in the system, the estimates will converge to an unknown metric scale. This fact is formalized in Civera et al. (2007), where the state vector is split into a metric parameter $s$, unobservable with only monocular measurements, and a dimensionless map and camera part

$$x_s = [s, \Pi_{x_c}, \Pi_{z_c}, \theta_c, \Pi_{v_x}, \Pi_{v_z}, \Pi_{\omega_c}, \Pi_{y_1}, \ldots]^{\mathrm{T}} \tag{4}$$

where the notation $\Pi_{\{\text{variable}\}}$, is used to indicate the corresponding dimensionless version of a state variable. For example, $[\Pi_{x_c}, \Pi_{z_c}]$ represents the dimensionless position of the camera. Camera measurements will reduce the scene geometry uncertainty, but not the uncertainty in the metric parameter $s$. The mapping from the state vector $x_s$ to the metric geometry is a nonlinear computation involving the dimensionless geometry and the parameters $s$ and $\Delta t$ (time)

$$\begin{cases} x_c = s\Pi_{x_c} \quad z_c = s\Pi_{z_c} \quad v_x = s\Pi_{v_x}\Delta t \\ v_z = s\Pi_{v_z}\Delta t \quad \omega_c = s\Pi_{\omega_c}\Delta t \\ y_i = [s\Pi_{x_{0i}}, s\Pi_{z_{0i}}, \theta_i, \Pi_{\rho_i}/s] \end{cases} \tag{5}$$

## Observability analysis

In this section, the observability of the system (equation (3)) is studied in terms of the metric parameter $s$, and the dimensionless camera part and map. When a system is fully observable, the lower bound of the error in our estimate of its state will depend only on the noise parameters of the system and will not be reliant on initial information about the states. This has important consequences in the context of SLAM.

For the sake of simplicity, the analysis will be developed by using the two-degrees-of-freedom model. Nevertheless, we are interested in having a SLAM method that works in a real six-degrees-of-freedom scenario. In this case, to generalize the analysis results obtained with the two-degrees-of-freedom system, one can think in the following manner. Let us consider a micro aerial vehicle moving freely in any direction in $\mathbb{R}^3 \times SO(3)$. Now let us consider the plane defined by the 3D velocity vector of the camera–micro aerial vehicle system and the line that is parallel to the $z$-axis of the navigation frame, which crosses the origin of the camera–micro aerial vehicle system coordinate frame. Finally, let us assume that, for each instant of time, the two-degrees-of-freedom analysis is carried out over this vertical plane. Of course, in this case, the two-degrees-of-freedom analysis will not contribute to investigating the observability of the yaw of the vehicle. However, this fact does not represent a major problem since we are mainly focused on investigating the observability of the metric scale parameter.

From the previous section, let us recall that the state of an $i$ th feature $y_i$ is defined by $y_i = [x_{0i}, z_{0i}, \theta_i, \rho_i]$, where $[x_{0i}, z_{0i}]$ is the position of the camera $C_s$ when the feature was first detected, $\theta_i$ is the first bearing measurement and $\rho_i = 1/d_i$ is the inverse of the feature depth $d_i$. Because $[x_{0i}, z_{0i}, \theta_i]$ is directly given when the $i$ th feature is initialized, the following analysis will be focused on the observability of the state of the camera $C_s$ and the inverse depth of the features. Therefore, $x$ will be taken as $x = [x_r, \rho_1, \rho_2, \ldots, \rho_n]$.

Now, consider the previous system state in terms of the metric parameter $s$ and the dimensionless parameters. Substituting equation 5 into equations 1 and 2, and augmenting the system state with $s$, the following dynamics are obtained

$$\begin{aligned} &\dot{s} = 0, \qquad &&\dot{\Pi}_{x_c} = s\Pi_{v_x}\Delta t, \qquad &&\dot{\Pi}_{z_c} = s\Pi_{v_z}\Delta t \\ &\dot{\theta}_c = s\Pi_{\omega_c}\Delta t, &&\dot{\Pi}_{v_x} = 0, &&\dot{\Pi}_{v_z} = 0 \\ &\dot{\Pi}_{\omega_c} = 0, &&\dot{\Pi}_{\rho_i} = 0 \end{aligned} \tag{6}$$

In the system defined by equation 6, a constant-acceleration camera model is assumed, as well as a rigid scene (the feature map remains static) and a constant metric scale. The system output equations are

$$\begin{aligned} h_{\theta i}(x_s) &= \operatorname{arctan2}\left(\frac{s\Pi_{z_c} - z_i}{s\Pi_{x_c} - x_i}\right) - \theta_c \\ x_i &= (s/\Pi_{\rho_i})\cos(\theta_i) + s\Pi_{x_{0i}} \\ z_i &= (s/\Pi_{\rho_i})\sin(\theta_i) + s\Pi_{z_{0i}} \end{aligned} \tag{7}$$

Hence, for $n$ landmarks being measured by the camera, the system output is defined as $h = [h_{\theta 1}, \ldots, h_{\theta n}]^{\mathrm{T}}$, and the system state as $x_s = [s, \Pi_{x_c}, \Pi_{z_c}, \theta_c, \Pi_{v_x}, \Pi_{v_z}, \Pi_{\omega_c}, \Pi_{\rho_1}, \Pi_{\rho_2}, \ldots, \Pi_{\rho_n}]^{\mathrm{T}}$

Hermann and Krener (1977) demonstrated that a nonlinear system is *locally weakly observable* if the observability rank condition, $\operatorname{rank}(\mathcal{O}) = \dim(x)$, is verified. For the analysis presented in this section, the observability matrices $\mathcal{O}$ are computed as described in the appendix.

**Table 1.** Degree of observability obtained by testing the combination of state variables $[\Pi_{x_c}, \Pi_{z_c}, \Pi_{v_x}, \Pi_{v_z}]$ equal to zero; rank$(\mathcal{O}) = 8$ denotes the maximum degree of observability for the system state defined in equation 9. In this table the number one is set for nonzero values $(1 : = \{x \in \mathbb{R} : x \neq 0\})$.

| | | | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| $\Pi_{v_z}$ | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 |
| $\Pi_{v_x}$ | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 | 0 | 0 | 1 | 1 |
| $\Pi_{z_c}$ | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 |
| $\Pi_{x_c}$ | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 |
| rank$(\mathcal{O})$ | 6 | 8 | 7 | 8 | 6 | 8 | 8 | 8 | 6 | 8 | 7 | 8 | 6 | 8 | 8 | 8 |

The following result is obtained for the system defined by equations 6 and 7:

- The maximum degree of observability is obtained with four landmarks. In this case, $\dim(x_s) = 11$, $\mathrm{rank}(\mathcal{O}) = 8$. As could be expected, the system is partially observable, in this case with three nonobservable modes. The observability is not increased by adding more landmarks. Because the metric scale is not observable when only angular measurements are available, one of those unobservable modes should correspond to the metric parameter $s$.

Now consider that measurements of the altitude of the camera–micro aerial vehicle are available. The additional system output equation for $h_{z_c}$ is

$$h_{z_c}(x_s) = s\Pi_{z_c} \tag{8}$$

Hence, for $n$ landmarks being measured by the camera, the system output is now defined as $h = [h_{z_c}, h_{\theta 1}, \ldots, h_{\theta n}]^{\mathrm{T}}$.

The following results are obtained for the system defined by equations 6, 7 and 8:

- In this case, the maximum degree of observability is obtained with three landmarks and $\dim(x_s) = 10$, $\mathrm{rank}(\mathcal{O}) = 8$. Hence, the system is still partially observable, but with two nonobservable modes. With measurement of altitude, one extra mode becomes observable.

Now let us consider a case where the orientation of the camera is fixed pointing to the same direction all along the trajectory. In this case, if the variables related with the orientation are removed, the system state is defined as

$$x_s = [s, \Pi_{x_c}, \Pi_{z_c}, \Pi_{v_x}, \Pi_{v_z}, \Pi_{\rho_1}, \Pi_{\rho_2}, \ldots, \Pi_{\rho_n}]^{\mathrm{T}} \tag{9}$$

Also note that, with this new system state, $\theta_c$ is removed from equation 7. With the foregoing modification, the following results are obtained:

- The maximum degree of observability is obtained with three landmarks, but in this case the system becomes observable, that is, $\dim(x_s) = 8$, $\mathrm{rank}(\mathcal{O}) = 8$.
- Considering the last results, it can be deduced that $\theta_c$ and $\Pi_{\omega_c}$ are the unobservable modes. But more importantly, it is shown that the metric scale becomes observable with the inclusion of altitude measurements.

For this scenario (system state defined by equation 9), some extra results are obtained by assuming different conditions in the vehicle state and by investigating their effects regarding observability (see Table 1):

- Movement of the vehicle along the vertical axis $(\Pi_{v_z} \neq 0)$ is a sufficient (not necessary) condition for full observability.
- A vertical position different from the origin, together with movement along the horizontal axis $(\Pi_{z_c} \wedge \Pi_{v_x} \neq 0)$ is a sufficient (not necessary) condition for full observability.
- The worst-case scenario of observability (rank $(\mathcal{O}) = 6$) is obtained when there is no movement at all $(\Pi_{v_z} = \Pi_{v_x} = 0)$.
- The horizontal position $(\Pi_{x_c})$ has no effect on observability.

## Method description

This section presents a six-degrees-of-freedom altitude-aided monocular SLAM method. The proposed method is motivated by the theoretical results obtained in the previous section. In this sense, as has previously been seen, it is difficult to recover the metric scale using monocular vision. However, it was found that the metric scale can become observable by assuming that the orientation of the camera is already known, together with the inclusion of altitude measurements in the system.

### *Assumptions and remarks*

As shown in Figure 2, the platform considered in this work is a quadrotor moving freely in any direction in $\mathbb{R}^3 \times SO(3)$. However, it is important to note that the proposed monocular SLAM method could be applied to any other kind of platform. The proposed method is mainly intended for local autonomous vehicle navigation. In this case, the local tangent frame is used as the navigation reference frame. Thus, the initial position of the vehicle defines the origin of the navigation coordinates frame. The navigation system follows the NED (north, east, down) convention. The magnitudes expressed in the navigation and in the camera frame are denoted by the

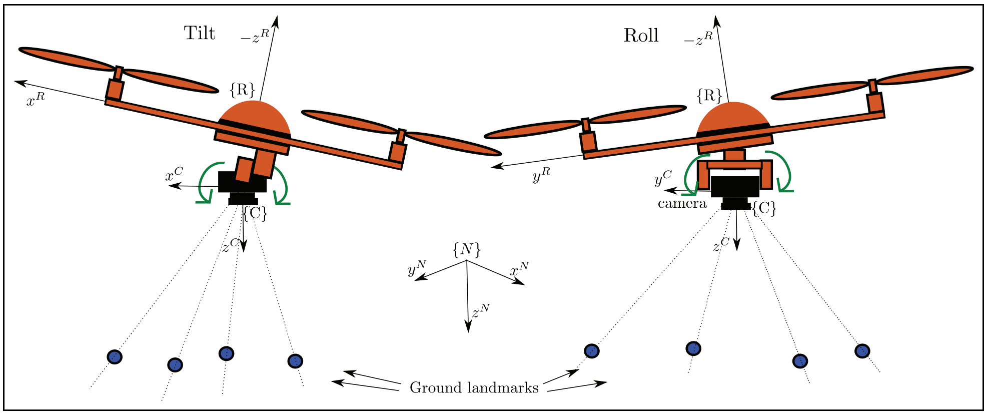

**Figure 2.** Coordinate systems: the local tangent frame is used as the navigation reference frame ($^{\mathrm{N}}$). In this work, the monocular camera is mounted over a servo-controlled gimbal that counteracts changes in attitude of the quadcopter.

superscripts $^{\mathrm{N}}$ and $^{\mathrm{C}}$, respectively. All the coordinate systems are right-handedly defined.

In the previous section, it has been demonstrated that the orientation of the camera is unobservable when only an angular sensor (i.e. a camera) is used. Fortunately, for such applications as aerial vehicles, attitude estimation is handled well by available systems (e.g. Euston et al., 2008; Munguia and Grau, 2014). In this sense, one approach that can be used to address the lack of observability in orientation consists of fusing measurements from an inertial measurement unit. Nevertheless, in this work, it is assumed that a monocular camera is mounted over a servo-controlled gimbal. This kind of accessory, used mainly for stabilizing video capture, has become very common in aerial applications. In our case, the gimbal is configured to counteract changes in attitude of the quadcopter; therefore, it stabilizes the orientation of the camera toward the ground (see Figure 2). With this assumption, the system state can be simplified by removing the variables related to orientation, and the problem is focused on position estimation.

Also, in the previous section, the system state has been split into a metric parameter $s$ and another part with the dimensionless map and camera, in order to show explicitly that the metric scale of the system is observable when altitude measurements are included in the system. By knowing this fact, hereinafter, the metric parameter will again be considered to be implicit into the system variables.

In this work, a barometric sensor is available for measuring atmospheric pressure. It is also assumed that the location of the origin of the camera frame with respect to other elements of the quadcopter (e.g. a barometer) is known and fixed. In this case, the position of the origin of the vehicle can be computed from the estimated location of the camera.

A standard monocular camera is considered. In this case, a central-projection camera model is assumed. The image plane, where a noninverted image is formed, is located in front of the camera's origin. The camera frame $^{\mathrm{C}}$ is right-handed, with the $z$-axis pointing to the field of view.

The $\mathbb{R}^3 \Rightarrow \mathbb{R}^2$ projection of a 3D point located at $p^{\mathrm{N}} = (x, y, z)^{\mathrm{T}}$ to the image plane $p = (u, v)$ is defined by

$$u = \frac{x'}{z'}, \qquad v = \frac{y'}{z'} \tag{10}$$

Let $u$ and $v$ be the coordinates of the image point $p$ expressed in pixel units, and

$$\begin{bmatrix} x' \\ y' \\ z' \end{bmatrix} = \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix} p^{\mathrm{C}} \tag{11}$$

Let $p^{\mathrm{C}}$ be the same 3D point $p^{\mathrm{N}}$, but expressed in the camera frame $^{\mathrm{C}}$ by $p^{\mathrm{C}} = R^{\mathrm{NC}} p^{\mathrm{N}}$. Let $R^{\mathrm{NC}}$ be the rotation matrix that allows transformation from the navigation frame $^{\mathrm{N}}$ to the camera frame $^{\mathrm{C}}$. Also, it is fulfilled that $R^{\mathrm{NC}} = (R^{\mathrm{CN}})^{\mathrm{T}}$, and $R^{\mathrm{CN}}$ is known by the use of the gimbal.

Inversely, a directional vector $w^{\mathrm{C}} = [w_x^{\mathrm{C}}, w_y^{\mathrm{C}}, w_z^{\mathrm{C}}]^{\mathrm{T}}$ can be computed from the image point coordinates $u$ and $v$ as

$$w^{\mathrm{C}}(u, v) = \left[\frac{u_0 - u}{f}, \frac{v_0 - v}{f}, 1\right]^{\mathrm{T}} \tag{12}$$

Vector $w^{\mathrm{C}}$ points from the camera optical centre position to the 3D point location; it can be expressed in the navigation frame by $w^{\mathrm{N}} = R^{\mathrm{CN}} w^{\mathrm{C}}$.

The distortion caused by the camera lens is considered through the model described in Bouguet (2008). By using the former model (and its inverse form), undistorted pixel coordinates $(u, v)$ can be obtained from $(u_d, v_d)$, and vice versa. In this case, it is assumed that the intrinsic parameters of the camera are already known: focal length $f$, principal point $(u_0, v_0)$, and radial lens distortion $k_1, \ldots, k_n$.

## System state

The system state to be estimated is

$$x = [x_r, y_1, y_2, \ldots, y_n]^{\mathrm{T}} \tag{13}$$

where $x_r = [r^{\mathrm{N}}, v^{\mathrm{N}}]^{\mathrm{T}}$ represents the state of the camera–quadcopter system. At the same time, $r^{\mathrm{N}} = [x_c, y_c, z_c]$ denotes the position of the camera expressed in the navigation frame and $v^{\mathrm{N}} = [v_x, v_y, v_z]$ denotes the linear velocity of the vehicle expressed in the navigation frame. In equation 13, $y_i$ represents the location of the $i$ th feature point, parametrized in its inverse-depth form

$$y_i = [r_i, \theta_i, \phi_i, \rho_i]^{\mathrm{T}} \tag{14}$$

where $r_i = [x_{0_i}, y_{0_i}, z_{0_i}]$ are the coordinates of the centre of the camera when the feature is observed for the very first time; $\theta_i$ and $\phi_i$ are azimuth and elevation, respectively; and $\rho_i = 1/d_i$ is the inverse of the feature depth $d_i$.

## Prediction

The architecture of the system is defined by the typical loop of predictions and updates of the standard EKF-SLAM, where the EKF propagates the vehicle state as well as the feature estimates. The interested reader is referred to the works of Bailey and Durrant-Whyte (2006) and Durrant-Whyte and Bailey (2006) for a good review of the EKF-SLAM methodology.

The system state $x$ is taken a step forward by the following discrete model

$$\begin{cases} r^{\mathrm{N}}_{k|k-1} = r^{\mathrm{N}}_{k-1|k-1} + v^{\mathrm{N}}_{k-1|k-1}\Delta t \\ v^{\mathrm{N}}_{k|k-1} = v^{\mathrm{N}}_{k-1|k-1} + V^{\mathrm{N}} \\ y_{1_{k|k-1}} = y_{1_{k-1|k-1}} \\ \vdots \\ y_{n_{k|k-1}} = y_{n_{k-1|k-1}} \end{cases} \tag{15}$$

At every step, it is assumed that there is an unknown linear velocity with zero-mean acceleration and known-covariance Gaussian processes $\sigma_a$, producing an impulse of linear velocity: $V^{\mathrm{N}} = \sigma_a^2 \Delta t$. Note that in equation (15) it is assumed that the map features $y_i$ remain static (rigid-scene assumption).

The state covariance matrix $P$ takes a step forward by

$$P_{k|k-1} = \nabla F_x P_{k-1|k-1} \nabla F_x^{\mathrm{T}} + \nabla F_u Q \nabla F_u^{\mathrm{T}} \tag{16}$$

where $Q$ and the Jacobians $\nabla F_x$ and $\nabla F_u$ are defined as

$$\nabla F_x = \begin{bmatrix} \dfrac{\partial f_v}{\partial x_r} & 0_{6\times n} \\ 0_{n\times 6} & I_{n\times n} \end{bmatrix}, \qquad \nabla F_u = \begin{bmatrix} \dfrac{\partial f_v}{\partial u} & 0_{6\times n} \\ 0_{n\times 3} & 0_{n\times n} \end{bmatrix} \\ Q = \begin{bmatrix} U & 0_{3\times n} \\ 0_{n\times 3} & 0_{n\times n} \end{bmatrix} \tag{17}$$

and $\partial f_v / \partial x_r$ represents the derivatives of the prediction model (equation 15) with respect to the robot state $x_r$ and where $\partial f_v / \partial u$ represents the derivatives of the prediction model with respect to the system input $u$. Uncertainties are incorporated into the system by means of the process noise covariance matrix $U = \sigma_a^2 I_{3\times 3}$, through parameter $\sigma_a^2$.

## Feature initialization

When a feature is detected for the first time from a monocular camera (bearing sensor), only information about the light reflected from the feature can be retrieved. In this work, the feature initialization process is conducted by means of the well-known method described in Montiel et al. (2006). In this case, new features are incorporated into the system state by assuming a hypothetical initial inverse-depth Gaussian prior on $\rho_i \sim N(\rho_0, \sigma_{\rho_0})$, which is applied to cover, with 95% probability, the range of depths from the closest possible depth to infinity (see Civera et al., 2007).

The initialization function $y_{id_{\mathrm{new}}} = f_{id}(x, u_i, v_i, d_i)$ used for computing new ID features is:

$$y_{id_{\mathrm{new}}} = [r_0, \theta_0, \phi_0, \rho_0]^{\mathrm{T}} \tag{18}$$

with $[r_0, \theta_0, \phi_0]$ calculated as in equation (14); $\rho_0$ is a free parameter. The system state $x$ is augmented with: $x_{\mathrm{new}} = [x_r, y_1, y_2, \ldots, y_n, y_{id_{\mathrm{new}}}]^{\mathrm{T}}$. The new covariance matrix $P_{\mathrm{new}}$ is computed by

$$P_{\mathrm{new}} = \nabla J \begin{bmatrix} P_{\mathrm{old}} & 0 & 0 \\ 0 & R & 0 \\ 0 & 0 & \sigma^2_{\rho_0} \end{bmatrix} \nabla J^{\mathrm{T}} \tag{19}$$

where $\nabla J$ is the Jacobian computed from the initialization function $f_{id}$ with respect to the system state $x$, and $R$ is the measurement noise covariance matrix.

## Visual updates

The active search technique (Davison and Murray, 1998) is used for tracking visual features $y_i$ in each frame. Assuming that, for the current frame, $n$ visual measurements are available, respectively, for features $y_1, y_2, \ldots, y_n$, the filter is updated with the Kalman update equations as

$$\begin{cases} x_{k|k} = x_{k|k-1} + K_k(z_k - h_k) \\ P_{k|k} = P_{k|k-1} - K_k S_k K_k^{\mathrm{T}} \\ K_k = P_{k|k-1} \nabla H_i^{\mathrm{T}} S_k^{-1} \\ S_k = \nabla H_i P_{k|k-1} \nabla H_i^{\mathrm{T}} + R_i \end{cases} \tag{20}$$

where $z = [z_{uv_1}, z_{uv_2}, \ldots, z_{uv_n}]^{\mathrm{T}}$ is the current measurement vector, $h = [h_1, h_2, \ldots, h_n]^{\mathrm{T}}$ is the current prediction measurement vector, $K$ is the Kalman gain and $S$ is the innovation covariance matrix. The following measurement prediction model $(u, v) = h_i(x)$ is used.

Each feature $y_i$ models a 3D point $p^{\mathrm{N}}$ located at

$$p^{\mathrm{N}} = r_i + \frac{1}{\rho_i} m(\theta_i, \phi_i) \tag{21}$$

where $m(\theta_i, \phi_i)$ is the unit vector defined by the pair of azimuth–elevation angles. The 3D point $p^{\mathrm{N}}$ is expressed in the camera frame $^{\mathrm{C}}$ by $p^{\mathrm{C}} = R^{\mathrm{NC}} p^{\mathrm{N}}$, where $R^{\mathrm{NC}}$ is the rotation matrix that transforms from the navigation frame $^{\mathrm{N}}$ to the camera frame $^{\mathrm{C}}$; $R^{\mathrm{NC}} = (R^{\mathrm{CN}})^{\mathrm{T}}$ and $R^{\mathrm{CN}}$ is known, assuming that the camera is pointing to the ground. Finally, the predicted image coordinates $(u, v)$ are computed from $p^{\mathrm{C}}$ using equations (10) and (11).

In equation (20), $\nabla H = [\nabla H_1, \nabla H_2, \ldots, \nabla H_n]^{\mathrm{T}}$ is the Jacobian formed by the partial derivatives of the measurement prediction model $h(x)$ with respect to the state $x$

$$\nabla H_i = \left[\frac{\partial h_i}{\partial x_r}, \ldots, 0_{2\times 3}, \ldots, \frac{\partial h_i}{\partial y_i}, \ldots, 0_{2\times 3}, \ldots\right] \tag{22}$$

where $\partial h_i/\partial x_r$ represents the partial derivative of the measurement prediction model $h_i$ with respect to the robot state $x_r$, and $\partial h_i/\partial y_i$ represents the partial derivative of $h_i$ with respect to feature $y_i$. Note that $\partial h_i/\partial y_i$ has only a nonzero value at the location (index) of the observed feature $y_i$. Let $R_i = (I_{2n\times 2n})\sigma_{uv}^2$ be the measurement noise covariance matrix.

### Altitude updates

Measurements of altitude can be inferred from measurements of atmospheric pressure. The proposed method is mainly intended for local autonomous vehicle navigation. Hence, the altitude or height of the micro aerial vehicle above a local ground location is computed from the change in pressure between the ground and the altitude of interest. The following formula can be used to compute the local altitude from barometric data

$$z_a = \left(1 - \left(\frac{B}{B_g}\right)^{\frac{K_R L_0}{Mg}}\right)\frac{T}{L_0} \tag{23}$$

where $B$ is the current barometric pressure measurement, $B_g$ is the barometric pressure at the initial position (home position), $K_R = 8.3144621$ N-m/(mol-K) is the universal gas constant, $L_0 = -0.0065$ K/m is the rate of temperature decrease in the lower atmosphere, $M = 0.0289644$ kg/mol is the standard molar mass of atmospheric air, $g = 9.80665$ m/s $^2$ is the acceleration of free fall and $T$ is the temperature at the flight location in kelvins. It is important to note that equation (23) provides the relative altitude of the vehicle with respect to its initial position (which is defined by $B_g$). At the initial altitude, $z_a = 0$; if the micro aerial vehicle moves below its initial altitude, $z_a$ will be negative.

To fuse the altitude measurements into the EKF-SLAM, a loosely coupled approach is used. In other words, the SLAM algorithm takes the altitude computed directly by equation (23) as a high-level input.

In this case, the altitude of the micro aerial vehicle, $z^{\mathrm{N}}$, measured (in the navigation frame) as $z_a$ can be modelled by

$$z_a = z^{\mathrm{N}} + \mathrm{x}_z + v_z \tag{24}$$

where $\mathrm{x}_z$ is an additive error (bias) and $v_z$ is a Gaussian white noise with a power spectral density $\sigma_z^2$. According to Hopkins et al. (2010), the dynamic error bias of a barometer is smaller than the bias of an accelerometer. Moreover, equation (23) also provides compensation for variations in temperature during the flight. Therefore, in this work, the bias $\mathrm{x}_z$ is neglected; however, in future work, it could be of interest to modify the approach in order to estimate $\mathrm{x}_z$ dynamically. Additionally, it is important to note that $v_z$ is assumed to have a Gaussian distribution; although this is not necessarily true, good experimental results were found by using this approach.

Measurements of altitude are incorporated into the system by means of the EKF standard update equations defined in equation (20). However, in this case, the measurement model $h_a(x)$ is simply defined by

$$h_a = z^{\mathrm{N}} \tag{25}$$

where $z^{\mathrm{N}}$ is taken directly from the system state $x$. For altitude updates, $\nabla H = [0, 0, 1, 0, 0, 0, 0_{1\times n6}]^{\mathrm{T}}$, where $n$ is the number of visual features and $R_i = \sigma_z^2$ is the uncertainty associated with altitude measurements.

### System initialization

A short initial period of time $t \in [0, T]$ is used for the system initialization task. During this period, the vehicle is assumed to be still. The barometric pressure at the initial position, $B_g$, is determined by averaging the readings of the barometer

$$B_g = \frac{1}{T}\int_0^T B\mathrm{d}t \tag{26}$$

The initial position $B_g$ will be used in equation (23) to compute the relative altitude of the vehicle. As mentioned in the section entitled [Assumtions]'Assumptions and remarks', the initial position of the vehicle defines the origin of the navigation coordinate frame. In this case, the system state vector is simply initialized as

$$x = [0_{1\times 3}, 0_{1\times 3}]^{\mathrm{T}} \tag{27}$$

## Experimental results

In this section, results obtained using synthetic data from simulations are presented, as well as results obtained from experiments with real data. The experiments are carried out to validate the performance of the proposed method.

### Simulations

To validate the proposed approach, the six-degrees-of-freedom system was simulated under different conditions. In this case, a quadrotor was commanded to fly over a surface composed of randomly distributed 3D points, following a circular pattern. During the flight, the altitude of the vehicle was varied to follow a sinusoidal pattern (see Figure 3). In simulations, the monocular camera was emulated by using the same parameter values of the camera employed in the experiments with real data. except that, in this case, a Gaussian noise with $\sigma_\theta = 1°$ was added to the angular measurements. Also, it is assumed that the camera is pointing to the ground and it is able to track, without error, all the landmarks inside its field of view. The altitude sensor was configured to emulate the behaviour of the Adafruit BMP183 barometric low-cost sensor. In this case, the measured altitude was corrupted with Gaussian noise with $\sigma_z = 0.25$ m.

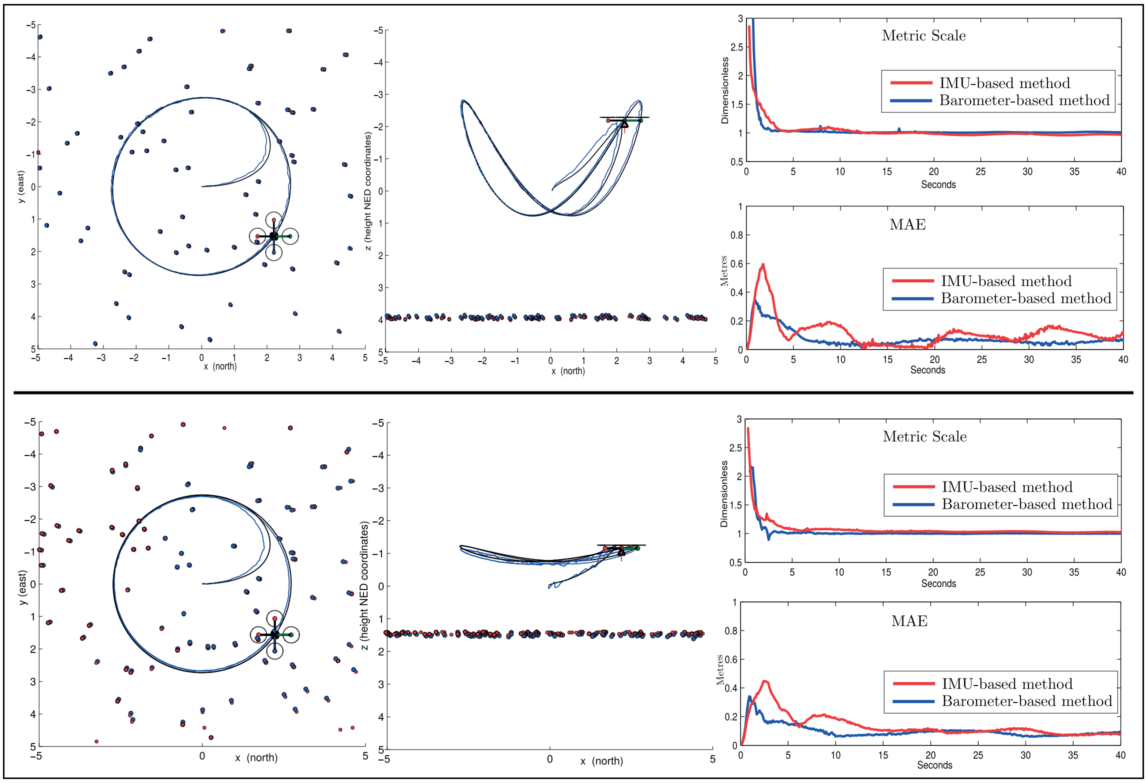

**Figure 3.** The proposed barometer-based approach was compared with an inertial-measurement-unit-based approach. Upper plots show results obtained for a flight with large variations in altitude. Lower plots show results obtained for a flight with small variations in altitude.
IMU: inertial measurement unit; MAE: mean absolute error; NED: north, east, down.

To obtain a better insight of the performance of the proposed method, a comparison with the approach used in Nützi et al. (2011) and Wang et al. (2014) is presented. In this case, the metric scale is explicitly estimated through an EKF that incorporates inertial measurements from an inertial measurement unit. For this purpose, the Invensense MPU-600 inertial measurement unit sensor was emulated in simulations. For this inertial-measurement-unit-based method, it is assumed that the alignment of the camera and the inertial measurement unit is perfectly known. In simulations, the hypothetical initial depth of features is set to $d_{\mathrm{ini}} = 0.5$ m and the dynamic error bias of the barometer and accelerometers is neglected.

Figure 3 shows the results obtained from simulating both methods: (i) the barometer-based method (proposed approach) and (ii) the inertial-measurement-unit-based method. Two experimental cases are presented: (i) a flight with large variations in altitude (upper plots) and (ii) a flight with small variations in altitude (lower plots). In experiments, the mean absolute error in position was computed, as well as the progression of the metric scale over time. In every case, the results were obtained by averaging 10 Monte Carlo executions.

Analysing the results closely, it can be appreciated that the convergence time of the inertial-measurement-unit-based method is longer than that obtained with the barometer-based method. It is important to note that convergence time obtained with the inertial-measurement-unit-based method is consistent with the results presented in Nützi et al. (2011). Also, note that the error in position is minimized as the metric scale converges to one. The simulation results suggest that the barometer-based approach should be a good alternative to the inertial-measurement-unit-based approach.

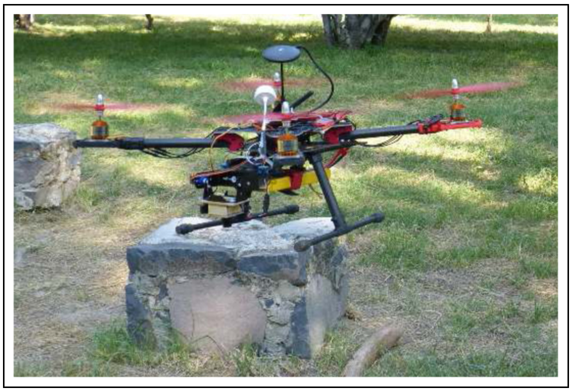

**Figure 4.** Radio-controlled quadrotor used for testing the proposed method.

## Experiments with real data

A custom-built quadrotor was used for experiments with real data (see Figure 4). The vehicle is equipped with: (i) an

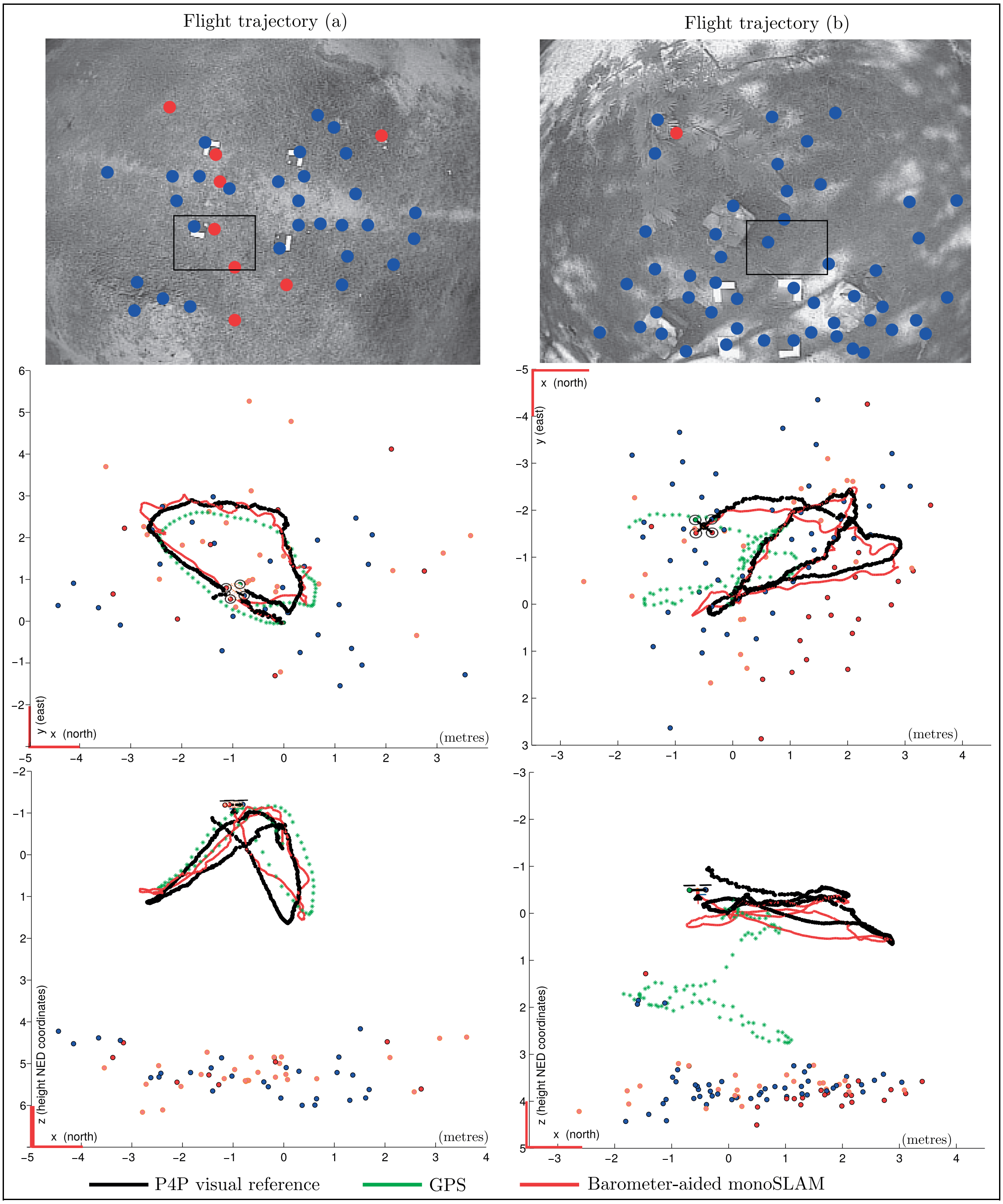

**Figure 5.** Estimated trajectory and map obtained for two different flight trajectories. The flight trajectory obtained with the proposed method is indicated in red. For comparison purposes, two external references of the flight trajectory were used: (i) GPS (green) and (ii) P4P visual reference (black).
NED: north, east, down.

Ardupilot unit as flight controller (Open Source Community, 2015a), (ii) a radio telemetry unit 3DR 915Mhz, (iii) a DX201 DPS camera with a wide angle lens, (iv) a 5.8 GHz video transmitter, (v) a NEO-M8N GPS unit and (vi) a low-cost Measurement Specialties MS5611-01BA03 barometer, which is included in the Ardupilot unit. The camera is mounted over a particularly low-cost gimbal, which is servo-controlled using standard servomotors.

In experiments, the quadrotor was manually radio-controlled. A custom-built C++ application running over a laptop was used to capture data from the quadrotor; the data were received via a MAVLINK protocol (Open Source

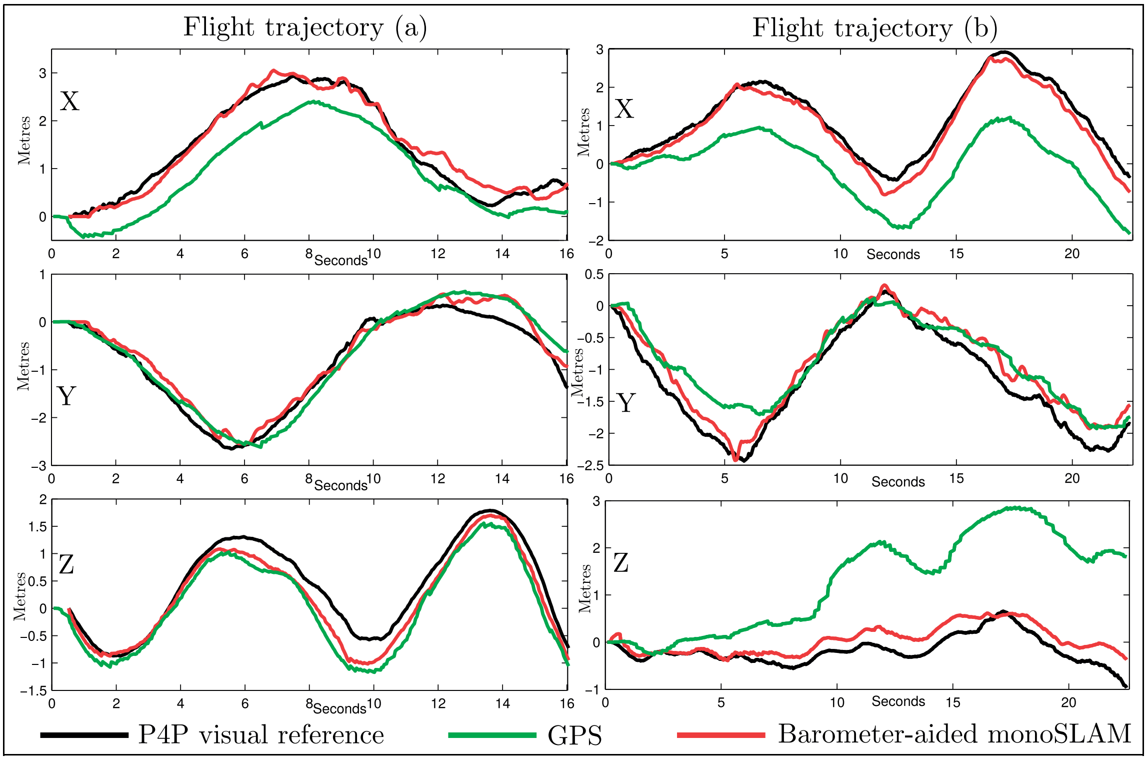

**Figure 6.** Estimated position in each coordinate frame: north (*x*), east (*y*) and down (*z*).

Community, 2015b) and the digitalized video signal transmitted from the quadrotor was also acquired. Data yielded by the barometer, as well as frames captured from the camera, were synchronized and stored in a dataset. The frames, with a resolution of 320 × 240 pixels in greyscale, were acquired at 26 fps. The quadrotor flights were conducted outdoors in open lawn areas. A MATLAB implementation of the proposed method was executed on the dataset offline to estimate the flight trajectory and the environment map.

Two different flight trajectories (*a* and *b*) were followed over two different test fields. Figure 5 shows the trajectory and estimated map for both cases. The upper plots show frames acquired during the flight. The middle plots show a zenithal (*x*–*y*) view of the maps and estimated trajectories. The lower plots show the sectional (*x*–*z*) view of the maps and estimated trajectories.

In the experiments, to show that the proposed method is able to work with real data for recovering the metric scale of the SLAM system, two external references of the flight trajectory were used: (i) the trajectory computed by filtering GPS measurements and (ii) the trajectory computed using a perspective four-point (P4P) technique (see the appendix for a description of the P4P technique). It is important to note that the objective of these experiments is only to obtain an insight into the ability of the proposed method for recovering the metric scale of the estimates, by visually comparing the amplitude of the signals obtained with the proposed method and the reference signals.

In this case, neither the GPS nor the P4P technique is considered, by any means, as a competing approach to the proposed method. In this sense, both, GPS and P4P make explicit use of a-priori known metric references. Instead, the GPS and P4P are only used as (imperfect) reference signals.

Figure 6 shows the progression over time for each estimated trajectory by (i) the proposed barometer-aided monocular SLAM method, (ii) GPS and (iii) P4P visual reference. A separate plot for each coordinate, north, east, and down (*x*, *y*, *z*), is presented. In this comparison, the results obtained using the proposed method are obtained by averaging 10 executions of each method. It is important to note that these averages are computed because the method is not deterministic, since the search for and detection of new visual features is conducted in a random manner over the images.

Based on the obtained results, it can be remarked that:

- *Flight trajectory (a)*. In general, a good concordance between the three estimates is obtained. For the *x*-coordinate only, a slightly major discrepancy is observed with the trajectory computed by the GPS. In this case an average of 13 satellites are available along the trajectory.
- *Flight trajectory (b)*. A better concordance is observed between the proposed method and the P4P visual reference. The error drift in GPS estimates (specifically in the vertical axis) probably comes from the lower availability of satellites, only eight in this case. It is important to recall that the GPS is affected by several sources of error. This can be especially problematic when a micro aerial vehicle is performing fine manoeuvres (Munguia et al., 2016).

Considering these results, it is shown that the proposed method is capable of working with real data obtained from low-cost sensors with the objective of estimating the flight trajectory of a micro aerial vehicle.

## Conclusions

In this work, an observability analysis has been conducted over a simplified two-degrees-of-freedom model of a monocular SLAM system. The results of the observability analysis confirm that the metric scale of the SLAM system is unobservable when only angular measurements are available. Additionally, it is shown that the modes corresponding to the orientation of the camera are also unobservable. Conversely, when measurements of altitude are included in the system, the metric scale can become observable. In this case, sufficient conditions for observability are presented.

Based on these theoretical results, a novel barometer-aided monocular SLAM method with application to micro aerial vehicles has been presented. To overcome the lack of observability of the camera orientation, the monocular camera is mounted over a servo-controlled gimbal in order to stabilize the orientation of the camera toward the ground. Hence, the problem is focused on position estimation. To overcome the lack of observability of the metric scale, a barometer is used to incorporate measurements of the altitude of the micro aerial vehicle in the filter.

Simulations and experiments with real data are carried out to validate the proposed approach. The results confirm that the actual metric scale can be retrieved by including altitude measurements in the system. It is also shown that the proposed method can be practically implemented by using low-cost sensors, to perform visual-based navigation in GPS-denied environments.

### Acknowledgements

The authors thank Jessica Davalos for her valuable support in correcting the language. The corresponding author also thanks his son Roderic Munguía for his contribution to this work.

### Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

### Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the PRO-SNI programme of the UdeG.

### ORCID iD

Rodrigo Munguía https://orcid.org/0000-0003-2282-2884

### References

Bailey T and Durrant-Whyte H (2006) Simultaneous localization and mapping (SLAM): Part II. *IEEE Robotics Automation Magazine* 13(3): 108–117.

Bouguet J-Y (2008) Camera calibration toolbox for MATLAB. Available at: http://www.vision.caltech.edu/bouguetj/calib_doc/ (accessed 12 December 2017).

Celik K and Somani AK (2013) Monocular vision SLAM for indoor aerial vehicles. *Journal of Electrical and Computer Engineering* 2013: 4.

Chatterjee C and Roychowdhury VP (2000) Algorithms for coplanar camera calibration. *Machine Vision and Applications* 12: 84–97.

Chowdhary G, Johnson EN, Magree D, et al. (2013) GPS-denied indoor and outdoor monocular vision aided navigation and control of unmanned aircraft. *Journal of Field Robotics* 30(3): 415–438.

Civera J, Davison AJ and Montiel JMM (2007) Dimensionless monocular SLAM. In: Mart J, Bened JM, Mendona AM, et al. (eds) *Pattern Recognition and Image Analysis. IbPRIA 2007* (*Lecture Notes in Computer Science*, vol. 4478). Berlin: Springer, pp. 412–419.

Davison A (2003) Real-time simultaneous localisation and mapping with a single camera. In: *Ninth IEEE international conference on computer vision*, Nice, France, 13–16 October 2003, vol. 2, pp. 1403–1410. Piscataway, NJ: IEEE.

Davison AJ and Murray DW (1998) Mobile robot localisation using active vision. In: *Proceedings of the 5th European conference on computer vision (ECCV '98)* (eds H Burkhardt and B Neumann), Freiburg, Germany, 2–6 June 1998, vol. II, pp. 809–825. London, UK: Springer-Verlag.

Davison A, Reid I, Molton N, et al. (2007) MonoSLAM: real-time single camera SLAM. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29(6): 1052–1067.

Dehghan SMM and Moradi H (2016) SLAM-inspired simultaneous localization of UAV and RF sources with unknown transmitted power. *Transactions of the Institute of Measurement and Control* 38(8): 895–907.

Durrant-Whyte H and Bailey T (2006) Simultaneous localization and mapping: Part I. *IEEE Robotics Automation Magazine* 13(2): 99–110.

Eade E and Drummond T (2006) Scalable monocular SLAM. In: *IEEE Computer Society conference on computer vision and pattern recognition*, New York, NY, 17–22 June 2006, vol. 1, pp. 469–476. Piscataway, NJ: IEEE.

Euston M, Coote P, Mahony R, et al. (2008) A complementary filter for attitude estimation of a fixed-wing UAV. In: *IEEE/RSJ international conference on intelligent robots and systems, IROS 2008*, Nice, France, 22–26 September 2008, pp. 340–345. Piscataway, NJ: IEEE.

Forster C, Lynen S, Kneip L, et al. (2013) Collaborative monocular SLAM with multiple micro aerial vehicles. In: *IEEE/RSJ international conference on intelligent robots and systems (IROS)*, Tokyo, Japan, 3–7 November 2013, pp. 3962–3970. Piscataway, NJ: IEEE.

Ganapathy S (1984) Decomposition of transformation matrices for robot vision. In: *IEEE international conference on robotics and automation*, Atlanta, GA, 13–15 March 1984, vol. 1, pp. 130–139. Piscataway, NJ: IEEE.

Hermann R and Krener AJ (1977) Nonlinear controllability and observability. *IEEE Transactions on Automatic Control* 22(5): 728–740.

Hopkins RE, Barbour NM, Gustafson DE, et al. (2010) Miniature inertial and augmentation sensors for integrated inertial/GPS based navigation applications. Technical report ADA581022, March 2010. Fort Belvoir, VA: Defense Technical Information Center.

Ihemadu OC, Naeem W and Ferguson S (2015) Optimizing the efficiency of the sub-map technique for large-scale simultaneous localization and mapping. *Transactions of the Institute of Measurement and Control* 37(3): 329–344.

Mirzaei F and Roumeliotis S (2008) A Kalman filter-based algorithm for IMU–camera calibration: observability analysis and performance evaluation. *IEEE Transactions on Robotics* 24(5): 1143–1156.

Montiel JMM, Civera J and Davison A (2006) Unified inverse depth parametrization for monocular SLAM. In: *Robotics: science and systems conference* (eds GS Sukhatme, S Schaal, W Burgard, et al.), Philadelphia, PA, 16–19 August 2006, pp. 16–19. Cambridge, MA: MIT Press.

Munguia R and Grau A (2007) Monocular SLAM for visual odometry. In: *IEEE international symposium on intelligent signal processing, WISP 2007*, Alcala de Henares, Spain, 3–5 October 2007. Piscataway, NJ: IEEE.

Munguia R and Grau A (2014) A practical method for implementing an attitude and heading reference system. *International Journal of Advanced Robotic Systems* 11(4): 62.

Munguia R, Urzua S, Bolea Y, et al. (2016) Vision-based SLAM system for unmanned aerial vehicles. *Sensors* 16(3): 372.

Nützi G, Weiss S, Scaramuzza D, et al. (2011) Fusion of IMU and vision for absolute scale estimation in monocular SLAM. *Journal of Intelligent & Robotic Systems* 61: 287–299.

Open Source Community (2015a) Ardupilot. Available at: http://ardupilot.com (accesed 12 December 2017).

Open Source Community (2015b) Ardupilot. Available at: http://qgroundcontrol.org/mavlink/start (accessed 12 December 2017).

Slotine JE and Li W (1991) *Applied Nonlinear Control*. Englewood Cliffs, NJ: Prentice-Hall.

Wang CL, Wang TM, Liang JH, et al. (2014) Bearing-only visual SLAM for small unmanned aerial vehicles in GPS-denied environments. *International Journal of Automation and Computing* 10(5): 387–396.

Weiss S, Scaramuzza D and Siegwart R (2011) Monocular-SLAM-based navigation for autonomous micro helicopters in GPS-denied environments. *Journal of Field Robotics* 28(6): 854–874.

Zhuang Y, Lin X, Hu H, et al. (2015) Using scale coordination and semantic information for robust 3-D object recognition by a service robot. *IEEE Sensors Journal* 15(1): 37–47.

# Appendix

## *Computation of observability matrix*

In this appendix, for simplicity, the symbol $\Pi$ will be removed from the dimensionless variables. For instance, $\Pi_{x_c}$ will be treated as $x_c$.

For the system with angular measurements, which is defined by equations (6) and (7), the observability matrix $\mathcal{O}$ can be constructed from

$$\mathcal{O} = \left[ \frac{\mathcal{L}_f^0(h_{\theta 1})}{\partial x} \frac{\mathcal{L}_f^1(h_{\theta 1})}{\partial x} \cdots \frac{\mathcal{L}_f^0(h_{\theta n})}{\partial x} \frac{\mathcal{L}_f^1(h_{\theta n})}{\partial x} \right]^{\mathrm{T}} \tag{28}$$

where $\mathcal{L}_f^i(h)$ is the $i$th order Lie derivative (Slotine and Li, 1991) of the scalar field of the measurement $h$ with respect to the vector field $f$. In this case, for each $i$ angular measurement $y_i = h_{\theta i}(x)$, the observability matrix $\mathcal{O}$ is augmented with the zeroth-order and first-order Lie derivatives

$$\left[ \begin{array}{ccccccc|ccc} 0 & f_{x_c} & f_{z_c} & -1 & 0 & 0 & 0 & \ldots 0 \ldots & f_{\rho_i} & \ldots 0 \ldots \\ f_{2_s} & f_{2_{x_c}} & f_{2_{z_c}} & 0 & f_{2_{v_x}} & f_{2_{v_z}} & -d & \ldots 0 \ldots & f_{2_{\rho_i}} & \ldots 0 \ldots \end{array} \right] \tag{29}$$

where

$$\begin{aligned} f_{x_c} &= \frac{\partial(h_\theta(x))}{\partial x_c} = f(s, x_c, z_c, \rho_i) \\ f_{z_c} &= \frac{\partial(h_\theta(x))}{\partial z_c} = f(s, x_c, z_c, \rho_i) \\ f_{\rho_i} &= \frac{\partial(h_\theta(x))}{\partial \rho_i} = f(s, x_c, z_c, \rho_i) \\ f_{2_s} &= f(s, x_c, z_c, \rho_i, v_x, v_z, \omega_c) \\ f_{2_{x_c}} &= \frac{\partial(f_{x_c} s)}{\partial x_c} v_x + \frac{\partial(f_{z_c} s)}{\partial x_c} v_z \\ f_{2_{z_c}} &= \frac{\partial(f_{x_c} s)}{\partial z_c} v_x + \frac{\partial(f_{z_c} s)}{\partial z_c} v_z \\ f_{2_{v_x}} &= f(s, x_c, z_c, \rho_i) \\ f_{2_{v_z}} &= f(s, x_c, z_c, \rho_i) \\ f_{2_{\rho_i}} &= \frac{\partial(f_{x_c} s)}{\partial \rho_i} v_x + \frac{\partial(f_{z_c} s)}{\partial \rho_i} v_z \end{aligned}$$

Note that the extended part of the matrix (equation (29)) corresponds to the Lie derivatives computed with respect to the inverse depth of the $i$th landmark. In this case, a nonzero column $[f_{1_{\rho_i}}, f_{2_{\rho_i}}]^{\mathrm{T}}$ will be located only at the indexes corresponding to the $i$th landmark. For $n$ landmarks, an observability matrix with dimension $\mathcal{O}_{2n \times (7 + n)}$ will be constructed. In equation (29), note also the dependency of $f_{2_{x_c}}$, $f_{2_{z_c}}$ and $f_{2_{\rho_i}}$ on $v_x$ and $v_z$. If $v_x = v_z = 0$, then $f_{2_{x_c}} = f_{2_{z_c}} = f_{2_{\rho_i}} = 0$.

When altitude measurements $y_0 = h_{z_c}(x)$ are considered, the observability matrix $\mathcal{O}$ can be computed from

$$\mathcal{O} = \left[ \frac{\mathcal{L}_f^0(h_{z_c})}{\partial x} \frac{\mathcal{L}_f^1(h_{z_c})}{\partial x} \frac{\mathcal{L}_f^0(h_{\theta 1})}{\partial x} \frac{\mathcal{L}_f^1(h_{\theta 1})}{\partial x} \cdots \frac{\mathcal{L}_f^0(h_{\theta n})}{\partial x} \frac{\mathcal{L}_f^1(h_{\theta n})}{\partial x} \right]^{\mathrm{T}} \tag{30}$$

In this case, the observability matrix $\mathcal{O}$ is augmented with the zeroth-order and first-order Lie derivatives obtained from altitude measurements

$$\left[ \begin{array}{ccccccc|c} z_r & 0 & s & 0 & 0 & 0 & 0 & 0_{1 \times n} \\ 2v_z s & 0 & 0 & 0 & 0 & s^2 & 0 & 0_{1 \times n} \end{array} \right] \tag{31}$$

With the inclusion of altitude measurements, and considering $n$ landmarks, an observability matrix with dimension $\mathcal{O}_{(2 + 2n) \times (7 + n)}$ will be constructed. When variables related to the orientation of the camera are not considered (equation (9)), the fourth and seventh columns are removed from equations (29) and (31). In this case, an observability matrix with dimension $\mathcal{O}_{(2 + 2n) \times (5 + n)}$ is constructed. In this work, the MATLAB symbolic toolbox was used to compute the derivatives and the numerical solutions for each observability matrix.

## P4P reference trajectory

In this work, to obtain an independent trajectory reference for evaluating the performance of the proposal, the following methodology was used.

Four marks are placed in the ground, forming a square of known dimensions (see Figure 5). Each corner is a coplanar point with spatial coordinates $[x_i, y_i, 0]$, with $i \in 1, \ldots, 4$, and their corresponding four undistorted image coordinates $[u_i, v_i]$ with $i \in 1, \ldots, 4$. Then, for each frame, a perspective four-point (P4P) technique (Chatterjee and Roychowdhury, 2000), is applied iteratively to compute the relative position of the camera with respect to the known metric reference. At each frame, the image location of the four corners is provided by a simple tracking algorithm designed for this purpose.

The P4P technique used to estimate the camera position, defined by $R^{\mathrm{CN}}$ and $r^{\mathrm{N}}$, is based on the work of Ganapathy (1984). The following linear system is formed with the vector $b$ as unknown parameter

$$\begin{bmatrix} x_1 f & y_1 f & 0 & 0 & -u_1 x_1 & -u_1 y_1 & f & 0 \\ 0 & 0 & x_1 f & y_1 f & -v_1 x_1 & -v_1 y_1 & 0 & f \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ x_4 f & y_4 f & 0 & 0 & -u_4 x_4 & -u_4 y_4 & f & 0 \\ 0 & 0 & x_4 f & y_4 f & -v_4 x_4 & -v_4 y_4 & 0 & f \end{bmatrix} b = \begin{bmatrix} u_1 \\ v_1 \\ \vdots \\ u_4 \\ v_4 \end{bmatrix} \tag{32}$$

where

$$b = \begin{bmatrix} \frac{r_{11}}{r_3} & \frac{r_{12}}{r_3} & \frac{r_{21}}{r_3} & \frac{r_{22}}{r_3} & \frac{r_{31}}{r_3} & \frac{r_{32}}{r_3} & \frac{r_1}{r_3} & \frac{r_2}{r_3} \end{bmatrix}^{\mathrm{T}} \tag{33}$$

The linear system represented in equation (33) is solved for

$$b = \begin{bmatrix} b_1 & b_2 & b_3 & b_4 & b_5 & b_6 & b_7 & b_8 \end{bmatrix}^{\mathrm{T}}$$

The camera position is computed from

$$\begin{aligned} R^{\mathrm{CN}} &= \begin{bmatrix} r_3 b_1 & r_3 b_2 & (R_{21}R_{32} - R_{31}R_{22}) \\ r_3 b_3 & r_3 b_4 & (R_{31}R_{12} - R_{11}R_{32}) \\ r_3 b_5 & r_3 b_6 & (R_{11}R_{22} - R_{21}R_{12}) \end{bmatrix}, \\ r^{\mathrm{N}} &= \begin{bmatrix} r_3 b_7 & r_3 b_8 & r_3 \end{bmatrix}^{\mathrm{T}} \end{aligned} \tag{34}$$

where

$$r_3 = \sqrt{\frac{f^2}{b_1^2 + b_3^2 + f^2 b_5^2}} \tag{35}$$

In equation (34), the third column of matrix $R^{\mathrm{CN}}$ is formed by the combination of the values of the first and second columns of the same matrix. The results obtained with this procedure can be very noisy; for this reason, a simple low-pass filter is applied to obtain the flight trajectory. The P4P trajectory is computed with respect to the metric reference. Trajectories obtained through visual SLAM have their own reference frame. In experiments, both reference frames are aligned to make the trajectories coincident at the beginning. In other words, it is assumed that the initial position of the quadcopter is known.