Chapter 2
Monocular Pose Estimation for an Unmanned Aerial Vehicle Using Spectral Features

Gastón Araguás, Claudio Paz, Gonzalo Perez Paina and Luis Canali

Pose estimation of Unmanned Aerial Vehicles (UAV) using cameras is currently a very active research topic in computer and robotic vision, with special application in GPS-denied environments. However, the use of visual information for ego-motion estimation presents several difficulties, such as features search, data association (feature correlation), inhomogeneous features distribution in the image, etc. We propose a visual position and orientation estimation algorithm based on the discrete homography constraint, induced by the presence of planar scenes, and the so-called spectral features in the image. Our approach has the following unique characteristics: it selects the appropriate distribution of the features in the image, it does not need either initialization process or search for features, and it does not depend on the presence of corner-like features in the scene. The position and orientation estimation is made using a down-looking monocular camera rigidly attached to a quadrotor. It is assumed that the floors over which the quadrotor flights are planar, and therefore two consecutive images are related by a homography induced by the floor plane. This homography constraint is more appropriate than the well-known epipolar constraint, which vanishes for a zero translation and loses rank in the case of planar scenes. The

G. Araguás (✉) · C. Paz · G.P. Paina · L. Canali
Centro de Investigación en Informática para la Ingeniería (CIII), Facultad Regional Córdoba, Universidad Tecnológica Nacional, Córdoba, Argentina
e-mail: garaguas@frc.utn.edu.ar

C. Paz
e-mail: cpaz@frc.utn.edu.ar

G.P. Paina
e-mail: gperez@frc.utn.edu.ar

L. Canali
e-mail: lcanali@frc.utn.edu.ar

N. Nedjah et al. (eds.), *Designing with Computational Intelligence*,
Studies in Computational Intelligence 664,
DOI 10.1007/978-3-319-44735-3_2

pose estimation algorithm is tested in a simulated dataset, and the robustness of the spectral features is evaluated in different conditions using a conveyor belt.

2.1 Introduction

In the last years quadrotors have gained popularity in entertainment, aero-shooting and many other civilian or military applications, mainly due to their low cost and great controllability. Between other tasks, they are a good choice for operation at low altitude, in cluttered scenarios or even for indoor applications. Such environments limit the use of GPS or compass measurements which are indeed excellent options for attitude determination in wide open outdoor areas [1, 12]. These constraints have motivated, over the last years, the extensive use of on-board cameras as a main sensor for state estimation [5, 14, 16]. In this context, we present a new approach to estimate the ego-motion of a quadrotor in indoor environments for smooth flights, using a down-looking camera for translation and rotation calculation. As a continuation of the work presented in [3], we propose the utilization of a fixed number of patches distributed on each image of the sequence to determine the ego-motion of the camera, based on the plane-induced homography that relates the patches in two consecutive frames.

A number of spatial and frequency domain approaches have been proposed to estimate the image-to-image transformation, between two views of a planar scene, most of them limited to similarities. Spatial domain methods need corresponding points, lines, conics, etc. [7, 9, 10], whose identification in many practical situations is non-trivial, thereby limiting their applicability. Scale, rotation, and translation invariant features have been popular facilitating recognition under these transformations. Geometry of multiple views of the same scene has been a subject of extensive research over the past decade. Important results relating corresponding entities such as points and lines can be found in [7, 9]. Recent work has also focused on more complex entities such as conics and higher-order algebraic curves [10]. However, these approaches depend on extracting corresponding entities such as points, lines or contours and do not use the abundant information present in the form of the intensity values in the multiple views of the scene. Frequency domain methods are in general superior to methods based on spatial features because the entire image information is used for matching. They also avoid the crucial issue regarding the selection of the best features.

Our work proposes the use of a fixed number of patches distributed on each image of the sequence to determine the pose change of a moving camera. The pose of the camera (and UAV) is estimated trough dead-reckoning, performing a time integration of ego-motion parameters determined between frames. We concentrate in the XY-position and the orientation estimation in order to fuse these parameters with the on-board IMU and altimeter sensors measurements. The camera ego-motion is estimated using the homography induced by the (assumed to be flat) floor, and the corresponding points needed to estimate the homography are obtained on the

frequency domain. A point in the image is represented by the spectral information of an image patch, which we call spectral feature [2, 3]. The correspondence between points in two consecutive frames is determined by means of the phase correlation between each spectral feature pair. These kind of features perform better than the interest points based on the image intensity when observing a floor with homogeneous texture. Moreover, since their position in the image plane is previously selected, they are always well distributed.

The transformation that relates two images taken from different views (with a moving camera) contains information about the spatial rotation and translation of the views, or the camera movement. Considering a downward-looking camera, and assuming that the floor is a planar surface, all the space points imaged by the camera are coplanar and there is a homography between the world and the image planes. Under this constraint, if the camera center moves, the images taken from different points of view are also related by a homography. The spatial transformation that relates both views can be completely determined from this homography between images.

The chapter is organized as follows: Sect. 2.2 details the homography-based pose estimation, with a review of the so-called plane-induced homography. In this section the homography decomposition used to obtain the translation and rotation of the camera is also presented; and in order to estimate the homography, the so-called spectral features are introduced in Sect. 2.3. The implementation details and the results are presented in Sect. 2.4, and finally Sect. 2.5 remarks the conclusions and future work.

2.2 Homography-Based Pose Estimation

The visual pose estimation is based on the principle that two consecutive images of a planar scene are related by a homography. The planar scene corresponds to the floor surface, which is assumed to be relatively flat, observed by the down-looking camera on the UAV. The spatial transformation of the camera, and therefore of the UAV, is encoded in this homography. Knowing the homography matrix that relates both images, the transformation parameters that describe the camera rotation and translation can be obtained.

In order to estimate the homography induced by the planar surface, a set of corresponding points on two consecutive images must be obtained. This process is performed selecting a set of features in the first image and finding the corresponding set of features in the second one. Then, the image coordinates of each feature in both images conform the set of corresponding image points needed to calculate the homography.

The image features used in our approach are the so-called spectral features, a Fourier domain representation of an image patch. Selecting a set of patches in both images (the same number, with the same size and position), the displacement between them is proportional to the phase shift between the associated spectral features, and

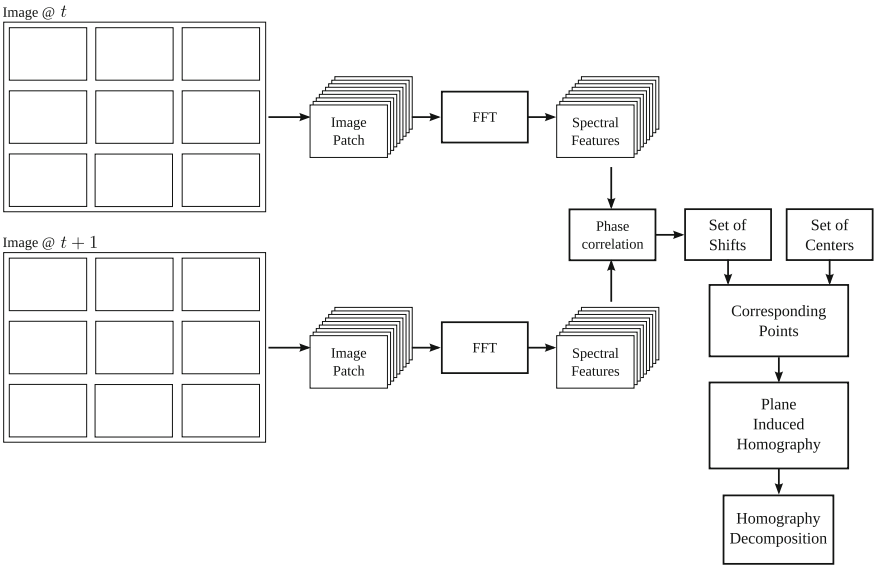

Fig. 2.1 Block diagram of the implemented visual pose estimation approach

can be obtained using the Fourier shift theorem. This displacement, in addition to the feature center, determines the correspondence between features in both images: that is, the set of corresponding points needed to estimate the homography.

In Fig. 2.1 a block diagram of the estimation process is shown. Here, as an example, nine spectral features in both images are used.

2.2.1 Review of Plane-Induced Homography

Given a 3D scene point $\mathbf{P}$, and two coordinate systems, CS_A and CS_B, the coordinates of the point $\mathbf{P}$ on each one can be denoted by $\mathbf{X}_A$ and $\mathbf{X}_B$ respectively. If $\boldsymbol{R}_A^B \in SO(3)$ is the rotation matrix that changes the representation of a point in CS_A to CS_B, and $\mathbf{T}_B \in \mathbb{R}^{3\times 1}$ is the translation vector of the origin of CS_A w.r.t CS_B (expressed in CS_B), then the representations of the point $\mathbf{P}$ relate each other as

$$\mathbf{X}_B = \boldsymbol{R}_A^B \mathbf{X}_A + \mathbf{T}_B. \tag{2.1}$$

We suppose now that the point $\mathbf{P}$ belongs to a plane π, denoted in the coordinate system CS_A by its normal $\mathbf{n}_A$ and its distance to the coordinate origin d_A. Therefore, the following plane equation holds

$$(\mathbf{n}_A)^T\mathbf{X}_A = d_A \qquad \Rightarrow \qquad \frac{(\mathbf{n}_A)^T\mathbf{X}_A}{d_A} = 1, \tag{2.2}$$

Plugging (2.2) into (2.1) we have

$$\mathbf{X}_B = \left(\boldsymbol{R}_A^B + \frac{\mathbf{T}_B}{d_A}(\mathbf{n}_A)^T\right)\mathbf{X}_A = \boldsymbol{H}_A^B\mathbf{X}_A, \tag{2.3}$$

with

$$\boldsymbol{H}_A^B \doteq \left(\boldsymbol{R}_A^B + \frac{\mathbf{T}_B}{d_A}(\mathbf{n}_A)^T\right). \tag{2.4}$$

The matrix $\boldsymbol{H}_A^B$ is a *plane-induced homography*, in this case induced by the plane π. As can be seen, this matrix encodes the transformation parameters that relate both coordinates systems ($\boldsymbol{R}_A^B$ and $\mathbf{T}_B$), and the structure parameters of the environment ($\mathbf{n}_A$ and d_A).

Considering now a moving camera associated to the coordinate system CS_A at time t_A and by CS_B at time t_B, according to the central projection model the relations between the 3D points and their projections on the camera normalized plane are given by

$$\lambda_A\mathbf{x}_A = \mathbf{X}_A; \qquad\qquad \lambda_B\mathbf{x}_B = \mathbf{X}_B \tag{2.5}$$

where $\lambda_A \in \mathbb{R}^+$ and $\lambda_B \in \mathbb{R}^+$. Using (2.5) in Eq. (2.3) we have

$$\lambda_B\mathbf{x}_B = H_A^B\lambda_A\mathbf{x}_A \qquad \Rightarrow \qquad \mathbf{x}_B = \lambda H_A^B\mathbf{x}_A, \tag{2.6}$$

with $\lambda = \frac{\lambda_A}{\lambda_B}$. Given that both vectors $\mathbf{x}_B$ and $\lambda H_A^B\mathbf{x}_A$ have the same direction

$$\mathbf{x}_B \times \lambda H_A^B\mathbf{x}_A = \hat{\boldsymbol{x}}_B H_A^B\mathbf{x}_A = 0, \tag{2.7}$$

with $\hat{\boldsymbol{x}}_B$ the skew-symmetric matrix associated to $\mathbf{x}_B$. The Eq. (2.7) is known as the *planar epipolar restriction*, and holds for all 3D points belonging to the plane π. Assuming that the camera is pointing to the ground (downward-looking camera) and that the scene structure is approximately a planar surface, all the 3D points captured by the camera will fulfill this restriction.

The homography $\boldsymbol{H}_A^B$ represents the transformation of the camera coordinate systems between instant t_A and t_B, hence, it contains the information of the camera rotation and translation between these two instants. This homography can be estimated knowing at least four corresponding points of two images. In our case the correspondence between these points is calculated in the spectral domain, by means of the *spectral features*. The complete process is detailed in Sect. 2.3.

2.2.2 Homography Decomposition

Following [13] $\boldsymbol{H}$ can be decomposed in order to obtain a non-unique solution (exactly four different solutions) $\left\{\boldsymbol{R}_i, \mathbf{n}_i, \frac{\mathbf{T}_i}{d_i}\right\}$. Then, adding some extra data for disambiguation we can arrive to the appropriate $\left\{\boldsymbol{R}_A^B, \mathbf{n}_A, \frac{\mathbf{T}_B}{d_A}\right\}$ solution.

Normalization

Given that the planar epipolar constraint ensures equality only in the direction of both vectors (Eq. (2.7)), what is actually obtained after the homography estimation is $\lambda\boldsymbol{H}$, that is[1]

$$\boldsymbol{H}_\lambda = \lambda\boldsymbol{H} = \lambda\left(\boldsymbol{R} + \frac{\mathbf{T}}{d}\mathbf{n}^T\right). \tag{2.8}$$

The unknown factor λ included in $\boldsymbol{H}_\lambda$ can be found as follows. Consider the product

$$\boldsymbol{H}_\lambda^T\boldsymbol{H}_\lambda = \lambda^2\left(\boldsymbol{I} + \boldsymbol{Q}\right) \tag{2.9}$$

with $\boldsymbol{I}$ the identity, $\boldsymbol{Q} = \mathbf{a}\mathbf{n}^T + \mathbf{n}\mathbf{a}^T + ||\mathbf{a}||^2\mathbf{n}\mathbf{n}^T$ and $\mathbf{a} = \frac{1}{d}\boldsymbol{R}^T\mathbf{T} \in \mathbb{R}^{3\times 1}$. The vector $\mathbf{a}\times\mathbf{n}$, perpendicular to $\mathbf{a}$ and $\mathbf{n}$, is an eigenvector of $\boldsymbol{H}_\lambda^T\boldsymbol{H}_\lambda$ associated to the eigenvalue λ^2, being that

$$\boldsymbol{H}_\lambda^T\boldsymbol{H}_\lambda(\mathbf{a}\times\mathbf{n}) = \lambda^2(\mathbf{a}\times\mathbf{n}). \tag{2.10}$$

So, if λ^2 is an eigenvalue of $\boldsymbol{H}_\lambda^T\boldsymbol{H}_\lambda$, then $|\lambda|$ is a singular value of $\boldsymbol{H}_\lambda$. It is easy to show that $\boldsymbol{Q}$ in (2.9) has one positive, one zero and one negative eigenvalue, what means that λ^2 is the second ordereigenvalue of $\boldsymbol{H}_\lambda^T\boldsymbol{H}_\lambda$, and $|\lambda|$ will be the second order singular value of $\boldsymbol{H}_\lambda$. That is, if $\sigma_1 > \sigma_2 > \sigma_3$ are the singular values of $\boldsymbol{H}_\lambda$, then

$$\boldsymbol{H} = \pm\frac{\boldsymbol{H}_\lambda}{\sigma_2} \tag{2.11}$$

To get the right sign of $\boldsymbol{H}$, the positive depth condition in (2.6) must be applied. In order to ensure that all the considered points are in front of the camera, all 3D points in plane π projected in the image plane must fulfill

$$(\mathbf{x}_B^j)^T\boldsymbol{H}\mathbf{x}_A^j = \frac{1}{\lambda_j} > 0, \quad \forall j = 1, 2, \ldots, n. \tag{2.12}$$

where $\left(\mathbf{x}_A^j, \mathbf{x}_B^j\right)$ are the projections of all points $\{\mathbf{P}\}_{j=1}^n$ lying on the plane π, at time t_A and t_B respectively.

[1] To avoid the abuse of notation we do not use here the sub and supra indexes A and B that refer to the corresponding coordinate systems.

Estimation of n

The homography $\boldsymbol{H}$ induced by the plane π preserves the norm of any vector in the plane, i.e. given a vector $\mathbf{r}$ such that $\mathbf{n}^T\mathbf{r} = 0$, then

$$\boldsymbol{H}\mathbf{r} = \boldsymbol{R}\mathbf{r} \tag{2.13}$$

and therefore $||\boldsymbol{H}\mathbf{r}|| = ||\mathbf{r}||$. Consequently, knowing the space spanned by the vectors that preserve the norm under $\boldsymbol{H}$, the perpendicular vector $\mathbf{n}$ is also known.

The matrix $\boldsymbol{H}^T\boldsymbol{H}$ is symmetric, and therefore admits eigenvalue decomposition. Being $\sigma_1^2, \sigma_2^2, \sigma_3^2$ the eigenvalues and $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ the eigenvectors of $\boldsymbol{H}^T\boldsymbol{H}$, then

$$\begin{gathered} \boldsymbol{H}^T\boldsymbol{H}\mathbf{v}_1 = \sigma_1^2\mathbf{v}_1, \quad \boldsymbol{H}^T\boldsymbol{H}\mathbf{v}_2 = \mathbf{v}_2, \\ \boldsymbol{H}^T\boldsymbol{H}\mathbf{v}_3 = \sigma_3^2\mathbf{v}_3 \end{gathered} \tag{2.14}$$

since by the normalization $\sigma_2^2 = 1$. That is, $\mathbf{v}_2$ is perpendicular to $\mathbf{n}$ and $\mathbf{T}$, so its norm is preserved under $\boldsymbol{H}$. From (2.14) it can be shown that the norm of the following vectors

$$\begin{aligned} \mathbf{u}_1 &\doteq \frac{\sqrt{1-\sigma_3^2}\mathbf{v}_1 + \sqrt{\sigma_1^2-1}\mathbf{v}_3}{\sqrt{\sigma_1^2-\sigma_3^2}}, \\ \mathbf{u}_2 &\doteq \frac{\sqrt{1-\sigma_3^2}\mathbf{v}_1 - \sqrt{\sigma_1^2-1}\mathbf{v}_3}{\sqrt{\sigma_1^2-\sigma_3^2}} \end{aligned} \tag{2.15}$$

is preserved under $\boldsymbol{H}$ too, as well as all vectors in the sub-spaces spanned by

$$S_1 = \text{span}\{\mathbf{v}_2, \mathbf{u}_1\}, \quad S_2 = \text{span}\{\mathbf{v}_2, \mathbf{u}_2\} \tag{2.16}$$

Therefore, there exist two possible planes that can induce the homography $\boldsymbol{H}$, π_1 and π_2, defined by the normal vectors to S_1 and S_2

$$\mathbf{n}_1 = \mathbf{v}_2 \times \mathbf{u}_1, \quad \mathbf{n}_2 = \mathbf{v}_2 \times \mathbf{u}_2. \tag{2.17}$$

Estimation of $\boldsymbol{R}$

The action of $\boldsymbol{H}$ over $\mathbf{v}_2$ and $\mathbf{u}_1$ is equivalent to a pure rotation

$$\boldsymbol{H}\mathbf{v}_2 = \boldsymbol{R}_1\mathbf{v}_2, \quad \boldsymbol{H}\mathbf{u}_1 = \boldsymbol{R}_1\mathbf{u}_1 \tag{2.18}$$

since both vectors are orthogonal to $\mathbf{n}_1$. The rotation of $\mathbf{n}_1$ can be computed as

$$\boldsymbol{R}_1\mathbf{n}_1 = \boldsymbol{H}\mathbf{v}_2 \times \boldsymbol{H}\mathbf{u}_1. \tag{2.19}$$

Defining the matrix $\boldsymbol{U}_1 = [\mathbf{v}_2, \mathbf{u}_1, \mathbf{n}_1]$ and $\boldsymbol{W}_1 = [\boldsymbol{H}\mathbf{v}_2, \boldsymbol{H}\mathbf{u}_1, \boldsymbol{H}\mathbf{v}_2 \times \boldsymbol{H}\mathbf{u}_1]$, from (2.18) and (2.19) we have

$$\boldsymbol{R}_1 \boldsymbol{U}_1 = \boldsymbol{W}_1 \tag{2.20}$$

and given that the set of vectors $\{\mathbf{v}_2, \mathbf{u}_1, \mathbf{n}_1\}$ form an orthogonal base in $\mathbb{R}^3$, the matrix $\boldsymbol{U}_1$ is non-singular, therefore

$$\boldsymbol{R}_1 = \boldsymbol{W}_1 \boldsymbol{U}_1^T, \tag{2.21}$$

that is

$$\boldsymbol{R}_1 = [\boldsymbol{H}\mathbf{v}_2, \boldsymbol{H}\mathbf{u}_1, \boldsymbol{H}\mathbf{v}_2 \times \boldsymbol{H}\mathbf{u}_1][\mathbf{v}_2, \mathbf{u}_1, \mathbf{n}_1]^T. \tag{2.22}$$

Considering now the set $\{\mathbf{v}_2, \mathbf{u}_2, \mathbf{n}_2\}$, in the same way we arrive to

$$\boldsymbol{R}_2 = \boldsymbol{W}_2 \boldsymbol{U}_2^T \tag{2.23}$$

where $\boldsymbol{U}_2 = [\mathbf{v}_2, \mathbf{u}_2, \mathbf{n}_2]$ and $\boldsymbol{W}_2 = [\boldsymbol{H}\mathbf{v}_2, \boldsymbol{H}\mathbf{u}_2, \boldsymbol{H}\mathbf{v}_2 \times \boldsymbol{H}\mathbf{u}_2]$, that is

$$\boldsymbol{R}_2 = [\boldsymbol{H}\mathbf{v}_2, \boldsymbol{H}\mathbf{u}_2, \boldsymbol{H}\mathbf{v}_2 \times \boldsymbol{H}\mathbf{u}_2][\mathbf{v}_2, \mathbf{u}_2, \mathbf{n}_2]^T. \tag{2.24}$$

Estimation of $\frac{\mathbf{T}}{d}$

Once $\boldsymbol{R}$ and $\mathbf{n}$ are known, the estimation of $\frac{\mathbf{T}}{d}$ is direct, as

$$\frac{\mathbf{T}_1}{d_1} = (\boldsymbol{H} - \boldsymbol{R}_1)\,\mathbf{n}_1, \tag{2.25}$$

$$\frac{\mathbf{T}_2}{d_2} = (\boldsymbol{H} - \boldsymbol{R}_2)\,\mathbf{n}_2, \tag{2.26}$$

which completes both solutions of the $\boldsymbol{H}$ decomposition.

Desambiguation

However, it should be noted that the term $\frac{\mathbf{T}}{d}\mathbf{n}^T$ in $\boldsymbol{H}$ introduces a sign ambiguity, since $\frac{\mathbf{T}}{d}\mathbf{n}^T = \frac{-\mathbf{T}}{d}(-\mathbf{n}^T)$, therefore the number of possible solutions rises to four,

$$\begin{matrix} \left\{\boldsymbol{R}_1, \mathbf{n}_1, \frac{\mathbf{T}_1}{d_1}\right\}, & \left\{\boldsymbol{R}_1, -\mathbf{n}_1, \frac{-\mathbf{T}_1}{d_1}\right\}, \\ \left\{\boldsymbol{R}_2, \mathbf{n}_2, \frac{\mathbf{T}_2}{d_2}\right\}, & \left\{\boldsymbol{R}_2, -\mathbf{n}_2, \frac{-\mathbf{T}_2}{d_2}\right\}. \end{matrix} \tag{2.27}$$

In order to ensure that the plane inducing the homography $\boldsymbol{H}$ appears in front of the camera, each normal vector $\mathbf{n}_i$ must fulfill $n_z < 0$, and therefore only two solutions remain. These two solutions are both physically possible, but given that most of the time the camera on the UAV is facing-down, we choose the solution with the normal vector $\mathbf{n}$ closest to $[0, 0, -1]^T$ in terms of the norm L_2.

2.3 Spectral Features Correspondence

The estimation of the homography given by two consecutive images from a moving camera requires a set of corresponding points. Classically, this set of points is obtained by detecting features, such as lines and corners in both images, and determining correspondences. The feature detectors are typically based on image gradient methods. An alternative to this approach is to use frequency-based features, or spectral features, and to determine correspondences in the frequency domain.

The so-called spectral feature refers to the Fourier domain representation of an image patch of $2^n \times 2^n$, where $n \in \mathbb{N}^+$ is set accordingly to the allowed image displacement [3]. The power of 2 of this patch size is selected based on the efficiency of the Fast Fourier Transform (FFT) algorithm. The number and position of spectral features in the image are set beforehand. Even though a minimum of four points are needed to estimate the homography, a higher number of features are used to increase the accuracy, and the RANSAC algorithm [8] is used for outliers elimination.

Consider two consecutive frames, where spectral features on each image were computed. To determine the correspondence between features is equivalent to determine the displacement between them. This displacement can be obtained using the spectral information by means of the Phase Correlation Method (PCM) [11]. This method is based on the Fourier shift theorem, which states that the Fourier transforms of two identical but displaced images differ only in a phase shift.

Given two images i_A and i_B of size $N \times M$ differing only in a displacement (u, v), such as

$$i_A(x, y) = i_B(x - u, y - v) \tag{2.28}$$

where

$$u \leq x < N - u, v \leq y < M - v, \tag{2.29}$$

their Fourier transforms are related by

$$I_A(\omega_x, \omega_y) = e^{-j(u\omega_x + v\omega_y)} I_B(\omega_x, \omega_y), \tag{2.30}$$

where I_A and I_B are the Fourier transforms of images i_A and i_B, respectively; u and v are the displacements for each axis. From (2.30), the amplitudes of both transformations are the same and only differ in phase which is directly related to the image displacement (u, v), and therefore this displacement can be obtained using the cross-power spectrum (CPS) of the given transformations I_A and I_B. The CPS of two complex functions is defined as

$$\mathcal{C}(F, G) = \frac{F(\omega_x, \omega_y) G^*(\omega_x, \omega_y)}{|F(\omega_x, \omega_y)||G^*(\omega_x, \omega_y)|} \tag{2.31}$$

where G^* is the complex conjugate of G.

Using (2.30) in (2.31) over the transformed images I_A and I_B, gives

$$\frac{I_A I_B^*}{|I_A||I_B^*|} = e^{-j(u\omega_x + v\omega_y)}. \tag{2.32}$$

The inverse Fourier transform of (2.32) is an impulse located exactly in (u, v), which represents the displacement between the two images

$$\mathcal{F}^{-1}[e^{-j(u\omega_x + v\omega_y)}] = \delta(x - u, y - v). \tag{2.33}$$

Using the discrete Fast Fourier Transform (FFT) algorithm instead of the continuous version, the result will be a pulse signal centered in (u, v) [17].

2.3.1 Corresponding Points

The previous subsection describes how to calculate the displacement between two images using PCM. Applying this method to each image patch pair, the displacement between spectral features is determined. The set of corresponding points required to estimate the homography can be constructed with the patch centers of the first image and the displaced patch centers of the second one, that is

$$\{\mathbf{x}_{A_i} \leftrightarrow \mathbf{x}_{A_i} + \Delta\mathbf{d}_i = \mathbf{x}_{B_i}\} \tag{2.34}$$

where $\Delta\mathbf{d}_i$ represents the displacement between the i-th spectral feature, and $\mathbf{x}_{A_i}$ the center of the i-th spectral feature in the CS_A. This is schematically shown in the zoomed area of Fig. 2.2. As shown in Sect. 2.2.1, this set of corresponding points is

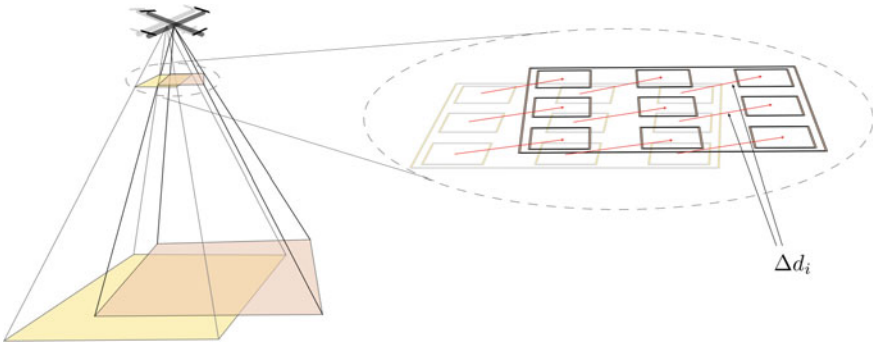

Fig. 2.2 Estimation of the rotation and translation between two consecutive images based on spectral features

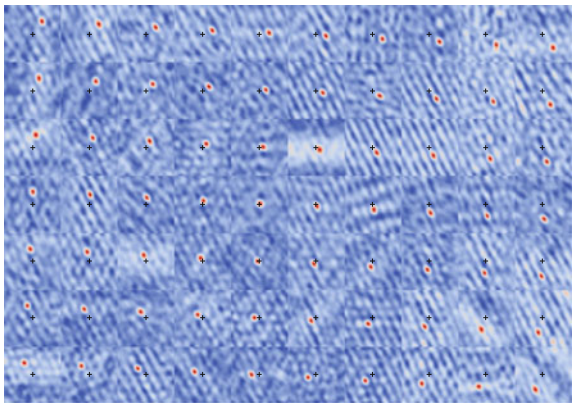

Fig. 2.3 Displacements between patches

related by a homography from which, using linear methods plus nonlinear optimization, the associated homography matrix can be computed [9].

In Fig. 2.3 a real set of spectral features is shown, where the black crosses represent each patch center and the yellow circles represent the output of PCM. It is important to note that the number, size, and position of spectral features are set beforehand: therefore, neither a search nor a correspondence process needs to be performed.

2.4 Implementation and Results

Summarizing, Algorithm 1 shows the proposed procedure to estimate the position and orientation, Algorithm 2 shows the procedure to determine the displacement between patches, and in Algorithm 3 the homography decomposition process is detailed.

Algorithm 1 Position and orientation estimation: function POSEESTIMATION(i_t, i_{t-1})

Extract patches p_{it} and p_{it-1} from i_t y i_{t-1}
for $\forall\{p_{it}, p_{it-1}\}$ **do**
  $\Delta\mathbf{d}_i \leftarrow$ FINDDISPLACEMENT(p_{it}, p_{it-1})
  $\mathbf{x}_{it} \leftarrow \mathbf{x}_{it-1} + \Delta\mathbf{d}_i$
end for
$H_\lambda \leftarrow$ FINDHOMOGRAPHY($\mathbf{x}_{it}, \mathbf{x}_{it-1}$)
R, $\mathbf{n}$, $\mathbf{T/d} \leftarrow$ GETRTN(H_λ)
return R, $\mathbf{n}$, $\mathbf{T/d}$

Algorithm 2 Patches displacement determination: function FINDDISPLACEMENT (p_{it}, p_{it-1})

$P_{it} \leftarrow$ FASTFOURIERTRANSFORM(p_{it})
$P_{it-1} \leftarrow$ FASTFOURIERTRANSFORM(p_{it-1})
$C \leftarrow$ CROSSPOWERSPECTRUM(P_{it}, P_{it-1})
$r \leftarrow$ INVERSEFASTFOURIERTRANSFORM(c)
$\Delta\mathbf{d}_i \leftarrow \text{argmax}\, r$
return $\Delta\mathbf{d}_i$

Algorithm 3 Homography matrix decomposition: function GETRTN(H_λ)

$U_\lambda, \Sigma_\lambda, V_\lambda^T \leftarrow \text{SVDecomp}(H_\lambda)$
$H \leftarrow H\lambda/\sigma_2$
$U, \Sigma, V^T \leftarrow \text{SVDecomp}(H)$
$[\mathbf{v}_1\ \mathbf{v}_2\ \mathbf{v}_3] \leftarrow V$

$$\mathbf{u}_1 \leftarrow \frac{\mathbf{v}_1\sqrt{1-\sigma_3^2} + \mathbf{v}_3\sqrt{\sigma_1^2-1}}{\sqrt{\sigma_1^2-\sigma_3^2}}\ ; \qquad \mathbf{u}_2 \leftarrow \frac{\mathbf{v}_1\sqrt{1-\sigma_3^2} - \mathbf{v}_3\sqrt{\sigma_1^2-1}}{\sqrt{\sigma_1^2-\sigma_3^2}}$$

$\mathbf{n}_1 \leftarrow \mathbf{v}_2 \times \mathbf{u}_1$; $\mathbf{n}_2 \leftarrow \mathbf{v}_2 \times \mathbf{u}_2$

Choose only the two physically possible solutions (this ensures that $\mathbf{n}_1$ and $\mathbf{n}_2$ have n_z positive component)

$U_1 \leftarrow [\mathbf{v}_2\ \mathbf{u}_1\ \mathbf{n}_1]$; $U_2 \leftarrow [\mathbf{v}_2\ \mathbf{u}_2\ \mathbf{n}_2]$
$W_1 \leftarrow [H\mathbf{v}_2\ H\mathbf{u}_1\ H\mathbf{v}_2 \times H\mathbf{u}_1]$; $W_2 \leftarrow [H\mathbf{v}_2\ H\mathbf{u}_2\ H\mathbf{v}_2 \times H\mathbf{u}_2]$
$R_1 \leftarrow W_1 U_1^T$; $R_2 \leftarrow W_2 U_2^T$
$T_1/d \leftarrow (H - R_1)\mathbf{n}_1$; $T_2/d \leftarrow (H - R_2)\mathbf{n}_2$

Choose the solution with n_z of each normal plane vector closest to zero
return $R, \mathbf{n}, \mathbf{T/d}$

2.4.1 Spectral Features Evaluation

In order to evaluate the performance of the spectral features in comparison with the intensity features, we use Shi-Tomasi algorithm [15] to detect intensity features in the first frame and Lucas–Kanade algorithm [4] to track these features in the second frame. OpenCV implementations of these algorithms are called `goodFeaturesToTrack()` and `calcOpticalFlowPyrLK()`. The evaluation was done using a camera mounted on a conveyor belt, shown in Fig. 2.4a, simulating a camera movement along Y axis at a constant height. In this way two frames differ only on a pure translation, without changes in scale or angles that affect the test. The displacement of the conveyor belt is measured with a laser telemeter, and the running distance in all the tests is of 0.3 m. The parameters estimated using spectral features are plotted in red, and those estimated using optical flow are plotted in blue. The texture of the floor seen by the camera is shown on Fig. 2.4b.

The performance of the algorithm with both types of features is tested using a zone in the conveyor belt plenty of corner-like features. In this case both approaches perform with low error and high stability. The results are shown in Fig. 2.5: the

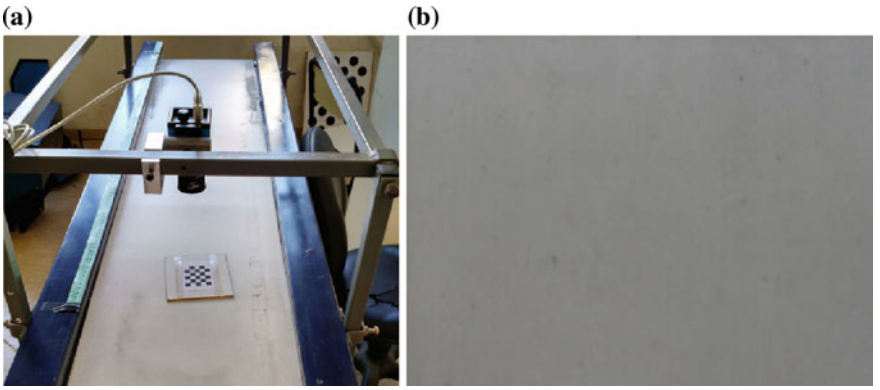

Fig. 2.4 **a** Camera mounting over a conveyor belt used to compare the performance of spectral feature against Shi-Tomasi algorithm. **b** Floor texture

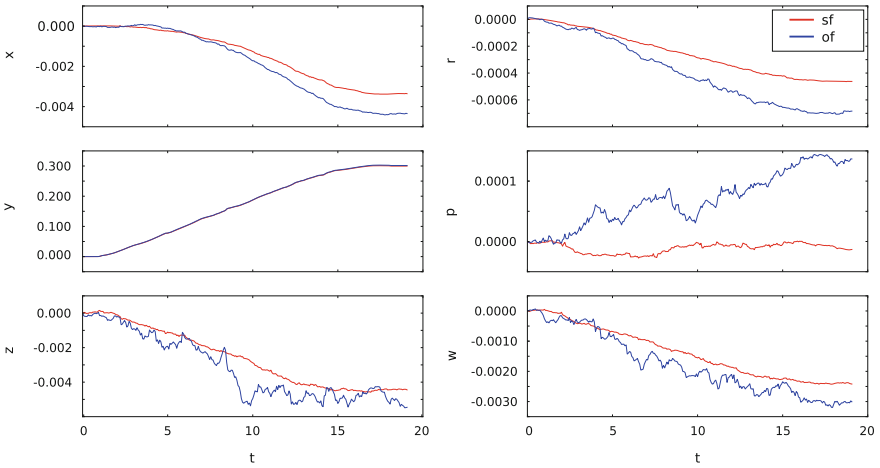

Fig. 2.5 Pose estimation using a textured floor. Plots are x, y, z in m and roll, pitch, yaw in *rad* versus time in s. *Red* spectral features, *Blue* corner-like features

distance measurements along the Y, X and Z axes, and the calculated yaw, pitch and roll angles using both types of features.

In Fig. 2.6 the estimated odometry using the conveyor belt texture with less corner-like features is shown, where the estimation with spectral features are plotted in red and the remaining in blue. As can be seen, the measurements calculated using spectral features are more accurate and stable.

Figure 2.7 shows a situation (pretty common when the floor contains low quality of corner-like features) where the intensity features failed, making the computation of the odometry totally incorrect. This failure is a consequence of a mismatch in the correlation of features, and occurs even more when the image goes out of focus, which

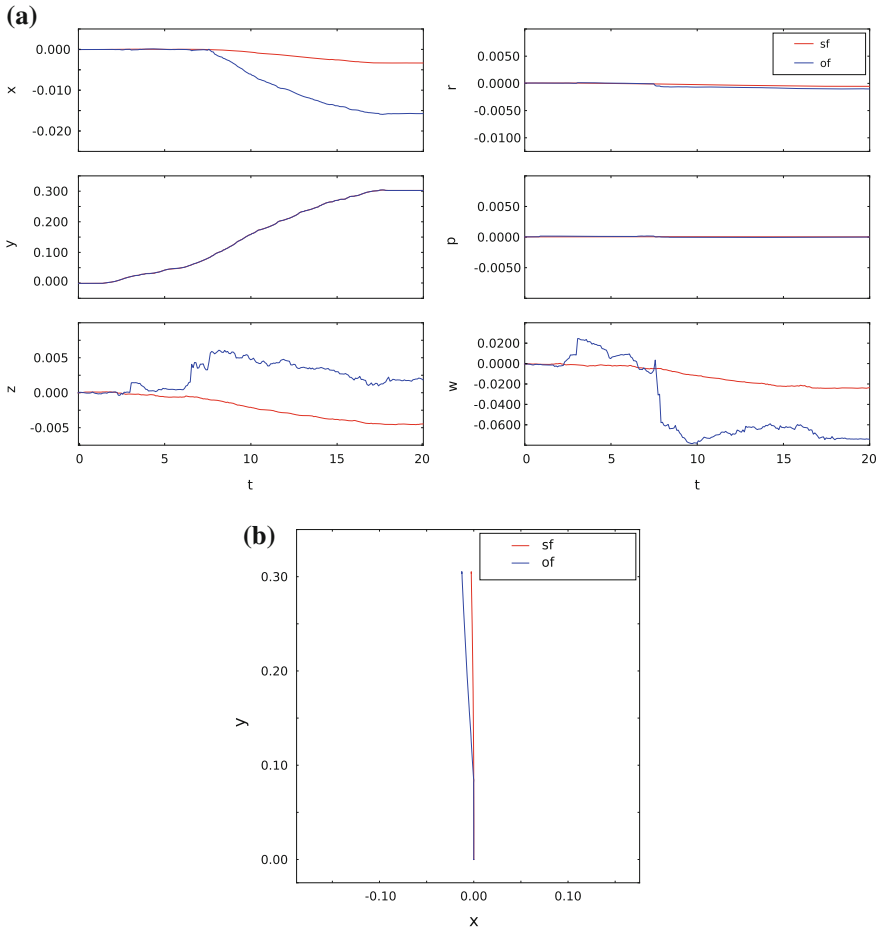

Fig. 2.6 Pose estimation using a floor with low number of corner-like features, similar to that shown in Fig. 2.4a. *Red* spectral features, *Blue* corner-like features

is a very usual situation during a quadrotor flight. In the third image of the sequence shown in Fig. 2.8 it is possible to appreciate this mismatch on the correlation of the features used by the optical flow algorithm, which are drawn in blue. This sequence corresponds to the pose estimation shown in Fig. 2.7d.

2.4.2 Pose Estimation in Simulated Quadcopter

The evaluation of the proposed visual pose estimation approach is performed with synthetic images obtained from a simulated quadrotor. In order to generate a six

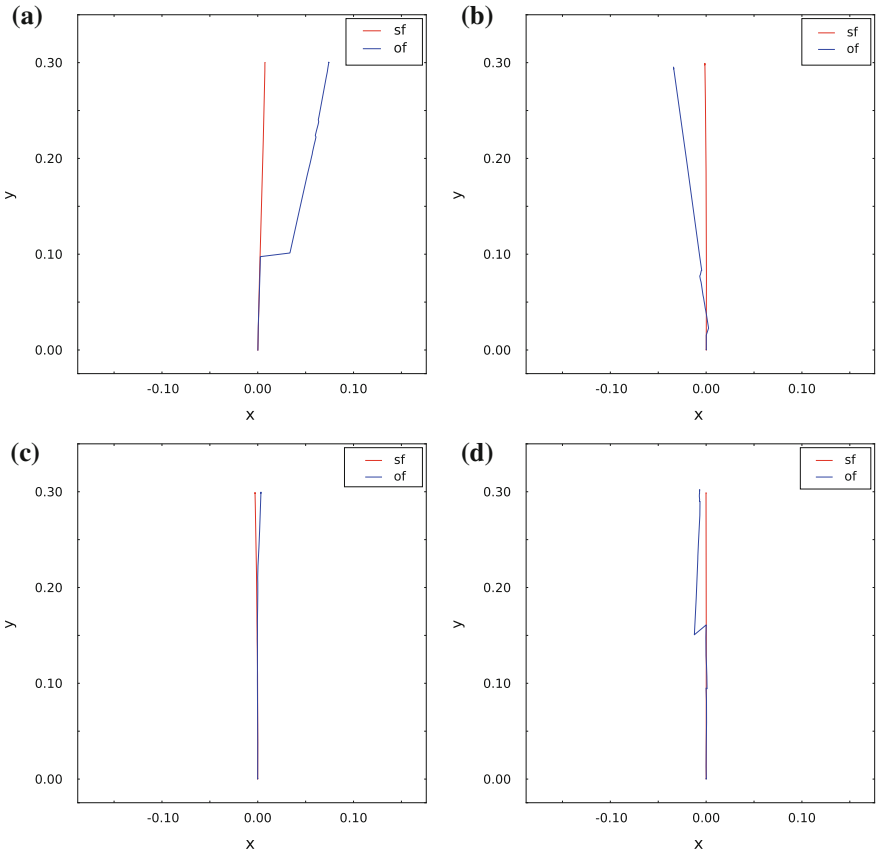

Fig. 2.7 *XY* plot of the pose estimation in a floor with low corner-like features. *Red* spectral features, *Blue* corner-like features

degrees of freedom motion similar to the motion of a real quadrotor, a simulated dynamic model was used. The truth robot position and orientation obtained in this way are then used to crop a sequence of images from a big one representing the observed flat surface. The ground truth pose is also used for evaluation purposes. The simulation of the quadrotor is based on Simulink, and the dynamic model is presented in [6]. Figure 2.9 shows an example of the path followed by the quadrotor used to generate the synthetic dataset.

The path consists on a change of altitude followed by two loops maintaining constant radius. During the loops, the heading angle, also called yaw angle, was set to grow up to 2π radians.

The images were obtained from a *virtual* downward-looking camera following the path described above, cutting portions of 640×480 from a bigger image of uniformly distributed noise in order to simulate a carpet. The virtual camera was

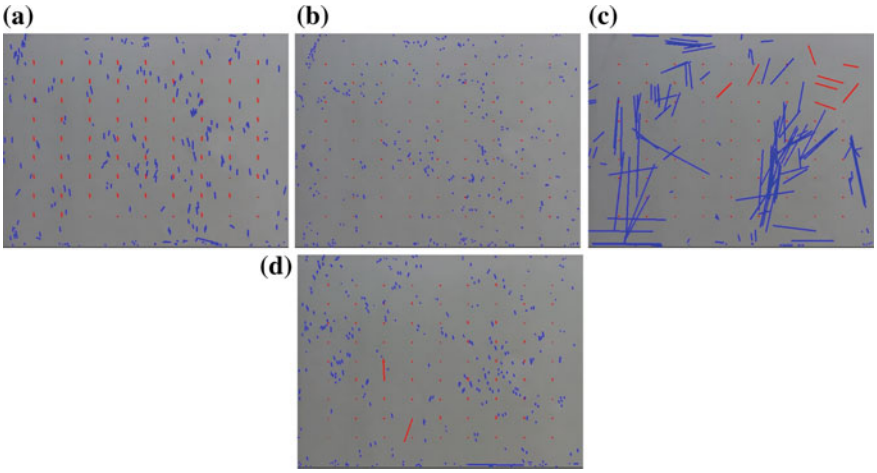

Fig. 2.8 Image sequence corresponding to a wrong pose estimation using intensity features. The floor texture is a low quality corner-like features type, similar to that shown in Fig. 2.4a. *Red* spectral features, *Blue* corner-like features

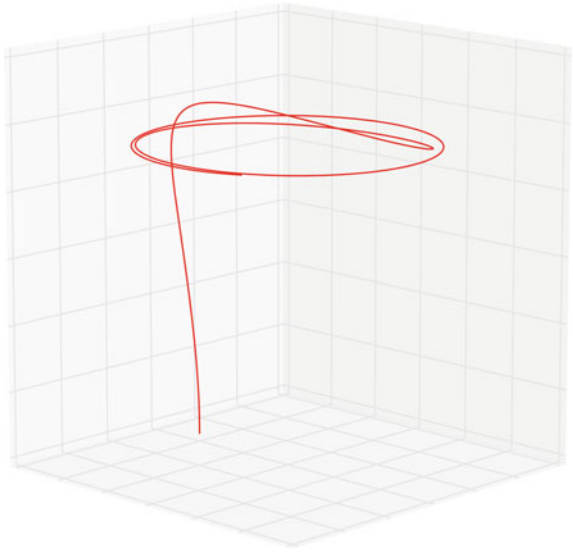

Fig. 2.9 Simulated position of a quadrotor with a six-degrees-of-freedom motion

configured with a pixel size of 5.6 μm and a focal length of approximately 1 mm. The algorithm was set with 42 patches of 128 × 128 pixels, equally distributed in the image.

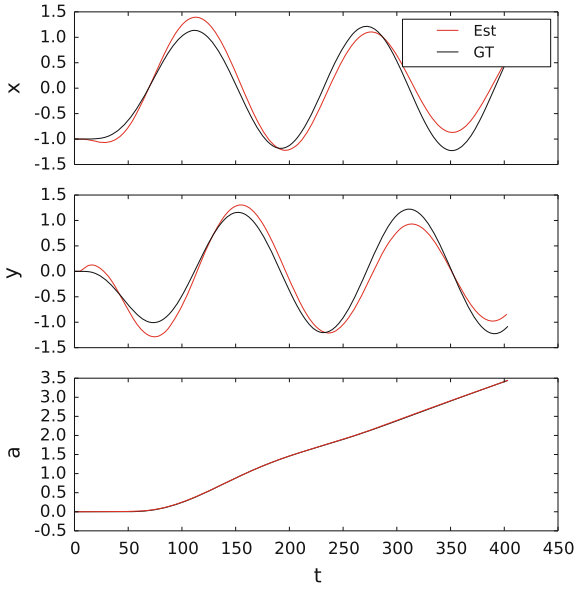

Fig. 2.10 Estimation of the XY-position and yaw angle of the UAV during a 20 s flight

In Fig. 2.10 the estimated parameters together with the ground truth are shown. The graphic at the top shows the X position estimation of the UAV, which performs a total of 2.5 m of change in the complete trajectory. The Y position estimation is plotted in the middle, and it has a similar behavior to the X one. As can be seen, the estimation error remains bounded in both axes all the time. The last graphic shows the yaw angle estimation, which follows the ground truth with a very small error.

2.5 Conclusions

In this work a new approach for visual estimation of the pose change of a quadrotor with a down-looking camera was presented. The proposed algorithm is based on the plane-induced homography that relates two views of the floor, and uses what we call "spectral features" to establish point-correspondences between images.

The main advantage of using spectral features as in this implementation is its robustness in low quality corner-like features floors. Evaluation of this was done using a conveyor belt to simulate a displacement of the camera, and comparing the performance of the spectral features with the Shi-Tomasi intensity features. The spectral features have shown to be more accurate and stable than the intensity features, especially in those scenarios with low quality corner-like features which appears frequently when the camera goes out of focus.

The evaluation of the visual algorithm using a synthetic dataset has shown that the XY-position is estimated without significant absolute error, despite the typical

accumulated error of the integration process. It is important to note that the view changes introduced by the orientation change (roll and pitch) over the flight did not induce any considerable error in the XY-position estimation. Likewise, the estimation of the heading (yaw) angle has shown to be accurate enough to be used in an IMU-camera fusion schema.

Acknowledgments This work was partially funded by the Argentinean institutions Universidad Tecnológica Nacional through the project 'Fusión Sensorial para Estimación de Posición y Orientación 3D", UTN-PID-2155, and the National Agency for Science and Technology Promotion through the project "Autonomous Vehicle Guidance Fusing Low-cost GPS and other Sensors", PICT-PRH-2009-0136, both currently under development at CIII, UTN, Córdoba, Argentina.

References

1. Angermann M, Frassl M, Doniec M, Julian B, Robertson P (2012) Characterization of the indoor magnetic field for applications in localization and mapping. In: 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN), pp. 1–9
2. Araguás G, Sánchez J, Canali L (2010) Monocular visual odometry using features in the fourier domain. In: VI Jornadas Argentinas de Robótica. Instituto Tecnológico de Buenos Aires, Buenos Aires, Argentina
3. Araguás G, Paz C, Gaydou D, Paina GP (2014) Quaternion-based orientation estimation fusing a camera and inertial sensors for a hovering UAV. J Intell RobotSyst 77(1):37–53
4. Baker S, Matthews I (2004) Lucas-kanade 20 years on: a unifying framework: Part 1: the quantity approximated, the warp update rule, and the gradient descent approximation. Int J Comput Vis 56(3):221–255
5. Bonin-Font F, Ortiz A, Oliver G (2008) Visual navigation for mobile robots: a survey. J Intell Robot Syst 53(3):263–296
6. Corke P (2011) Robotics, vision and control, springer tracts in advanced robotics, vol 73. Springer, Berlin
7. Faugeras O, Luong QT (2004) The geometry of multiple images: the laws that govern the formation of multiple images of a scene and some of their applications. MIT press, Cambridge
8. Fischler MA, Bolles RC (1981) Random sample consensus: a paradigm for model fitting with applications to image analysis and automated cartography. Commun ACM 24(6):381–395
9. Hartley R, Zisserman A (2003) Multiple view geometry in computer vision. Cambridge University Press, Cambridge
10. Kaminski JY, Shashua A (2004) Multiple view geometry of general algebraic curves. Int J Comput Vis 56(3):195–219
11. Kuglin CD, Hines DC (1975) The phase correlation image alignment method. Proc Int Conf Cybern Soc 4:163–165
12. Li B, Gallagher T, Dempster A, Rizos C (2012) How feasible is the use of magnetic field alone for indoor positioning? In: 2012 International Conference on Indoor Positioning and Indoor Navigation (IPIN), pp. 1–9
13. Ma Y, Soatto S, Kosecká J, Sastry SS (2010) An invitation to 3-d vision: from images to geometric models. Springer, New York
14. Scaramuzza D, Achtelik M, Doitsidis L, Friedrich F, Kosmatopoulos E, Martinelli A, Achtelik M, Chli M, Chatzichristofis S, Kneip L, Gurdan D, Heng L, Lee GH, Lynen S, Pollefeys M, Renzaglia A, Siegwart R, Stumpf J, Tanskanen P, Troiani C, Weiss S, Meier L (2014) Vision-controlled micro flying robots: from system design to autonomous navigation and mapping in GPS-Denied environments. IEEE Robot Autom Mag 21(3):26–40

15. Shi J, Tomasi C (1994) Good features to track. In: 1994 IEEE computer society conference on computer vision and pattern recognition, 1994. Proceedings CVPR'94, pp. 593–600
16. Weiss S, Achtelik MW, Lynen S, Achtelik MC, Kneip L, Chli M, Siegwart R (2013) Monocular vision for long-term micro aerial vehicle state estimation: a compendium. J Field Robot 30(5):803–831
17. Zitová B, Flusser J (2003) Image registration methods: a survey. Image Vis Comput 21(11):977–1000