A Multi-Resolution Image Alignment Technique Based on Direct Methods for Pose Estimation of Aerial Vehicles

Carol Martínez*, Luis Mejias† and Pascual Campoy*

*Computer Vision Group, Centro de Automática y Robótica CAR
Universidad Politécnica de Madrid, José Gutiérrez Abascal 2, 28006 Madrid, Spain
Email: carolviviana.matinez@upm.es, pascual.campoy@upm.es
†Australian Research Centre for Aerospace Automation, School of engineering Systems
Queensland University of Technology, QLD 4001, Australia
Email: luis.mejias@qut.edu.au

***Abstract*—In this paper, we seek to expand the use of direct methods in real-time applications by proposing a vision-based strategy for pose estimation of aerial vehicles. The vast majority of approaches make use of features to estimate motion. Conversely, the strategy we propose is based on a MR (Multi-Resolution) implementation of an image registration technique (Inverse Compositional Image Alignment ICIA) using direct methods. An on-board camera in a downwards-looking configuration, and the assumption of planar scenes, are the bases of the algorithm. The motion between frames (rotation and translation) is recovered by decomposing the frame-to-frame homography obtained by the ICIA algorithm applied to a patch that covers around the 80% of the image. When the visual estimation is required (e.g. GPS drop-out), this motion is integrated with the previous known estimation of the vehicles' state, obtained from the on-board sensors (GPS/IMU), and the subsequent estimations are based only on the vision-based motion estimations. The proposed strategy is tested with real flight data in representative stages of a flight: cruise, landing, and take-off, being two of those stages considered critical: take-off and landing. The performance of the pose estimation strategy is analyzed by comparing it with the GPS/IMU estimations. Results show correlation between the visual estimation obtained with the MR-ICIA and the GPS/IMU data, that demonstrate that the visual estimation can be used to provide a good approximation of the vehicle's state when it is required (e.g. GPS drop-outs). In terms of performance, the proposed strategy is able to maintain an estimation of the vehicle's state for more than one minute, at real-time frame rates based, only on visual information.**

I. INTRODUCTION

Image information has been used for different purposes in the field of aerial vehicles: collision avoidance [1], surveillance [2][3], autonomous vision-based landing tasks [4][5], or SLAM (Simultaneous Localization and Mapping) [6][7], among others. In all of these applications, the visual information has been used as a main or complementary sensor to improve the vehicle's capabilities.

A common UAV system uses GPS position to correct IMU (Inertial Measurement Unit) data from drift in order to obtain the UAV's state. That is why any loss of GPS signal will cause serious problems in the UAV operation. Nonetheless, visual odometry approaches have been proposed to solve the problems that arise when the GPS information becomes unavailable or is unreliable (e.g. when flying close to obstacles, or during GPS dropouts).

The different approaches presented in the literature that make use of vision as an additional or complementary sensor to estimate the UAV's state can be classified according to the type of information that is recovered to estimate the vehicles' position and orientation. Two categories can be identified: the approaches that use features to obtain the UAV state, and those that use the pixels' information (direct methods).

On the other hand, an additional characterization can arise from the motion estimation method used to extract the vehicle's motion. When the scene is planar or can be assumed planar, and the intrinsic parameters of the camera are known, homography decomposition techniques are commonly used in monocular systems to extract the camera displacement.

In [8], an algorithm for estimating the position and orientation of aerial vehicles assuming planar scenes is presented. Matched corner features are used to calculate a frame-to-frame homography, and the homography decomposition technique is used to estimate the motion. Another system is presented in [9]. This method offers a drift-free estimation when GPS signal is unavailable. The algorithm uses two techniques: a position estimation based on visual odometry (using corner features) that drifts, and an image registration technique that matches the current image with a geo-referenced image in order to compensate the drift. The position estimation is derived from the homography matrix considering that the attitude data is obtained from the IMU, and the distance to the plane is obtained from an altimeter.

In [10], a feature-based pose estimation algorithm for piecewise planar scenes is presented. The algorithm establishes a relationship between images through the homography matrix. GPS data is linked with image data to provide inertial measurements. Simulation results illustrate the performance of the algorithm.

Binocular systems have also been used. In [11], by using a Kalman Filter, the system fuses the vision-based data with inertial data. Again, features are detected, and tracked with the KLT feature-based algorithm, but in this case, their 3D position is found by triangulation.

As can be seen, most of the work presented in the literature makes use of features to determine the homography relationship between images, and also makes use of the homography matrix to obtain the motion of the UAV (rotation and translation).

Motivated by this fact, in this paper we want to address the pose estimation problem using image registration and homography decomposition techniques, but using direct methods [12] instead of feature-based methods [13].

In [14], a direct method was used in an inertially-aided visual odometry system. Its election is based on the fact that the use of direct methods can bring more accurate results in the estimation due to the amount of information that is considered in the evaluation of the motion model [12]. However, as the authors say, their results were presented as a proof of concepts, and additional improvements must be incorporated in order to achieve a real-time operation of the algorithm.

Thus, the contribution of this work is to expand the use of direct methods in a real-time pose estimation application. To achieve this, we propose to use a MR strategy of the ICIA algorithm [15]. This MR strategy helps to improve the estimation when the assumption of small motions of direct methods is not satisfied (e.g. flying at low altitudes -take-off and landing-), and additionally permits to vary the number of pixels used in the estimation of the motion without affecting the quality of the estimation.

The pose estimation algorithm assumes that an initial estimation of the state of the vehicle is available before the algorithm starts operating (e.g. the state before the GPS dropout). Hence, the proposed algorithm finds the motion between frames by decomposing the homography obtained with the ICIA, and integrates it with the previous estimation in order to obtain the current state of the vehicle, using a method that is similar to the one proposed in [10].

Therefore, our work differs from previous approaches in that in our proposed strategy hierarchical-based direct methods are used to obtain a real-time pose estimation that is based only on visual information. Therefore, our proposal extends the results obtained in [14], where an algorithm based on direct methods was proposed but its complexity did not allow a real-time operation of it.

The paper is organized as follows: in Section II an explanation of the MR implementation of the the ICIA algorithm is presented. Section III then presents the pose estimation algorithm. In Section IV, tests and results that evaluate the performance of the algorithm are shown. Finally, conclusions and the direction of future work are presented.

II. Multi-Resolution Inverse Composition Image Alignment MR-ICIA

The image registration technique will be in charge of identifying the transformation (motion model) that allows to align the current image $\mathbf{I}$ with a reference image $\mathbf{T}$.

In this section, we describe the MR implementation of the ICIA algorithm, by presenting first the motion model used, then the derivation of the ICIA proposed in [15], and finally its hierarchical implementation.

A. Motion Model: Homography

The algorithm we are presenting is based on the assumption of planar scenes. For our application -aerial images-, this assumption is valid considering that the distance to the ground is larger than the height of the objects. On the other hand, by assuming planar scenes, the motion can be recovered by using the homography as the transformation in charge of aligning two consecutive images.

This transformation has eight degrees of freedom (rotation, translation, and surface parameters), and is parameterized as follows:

$$\mathbf{x}' = \begin{bmatrix} 1+p_1 & p_2 & p_3 \\ p_4 & 1+p_5 & p_6 \\ p_7 & p_8 & 1 \end{bmatrix} \mathbf{x} \tag{1}$$

Different parameterizations of the homography matrix can be adopted, depending on the number of degrees of freedom (DOF) the application requires to recover, as shown in [16].

B. Image Registration: MR-ICIA

The starting point of the algorithm is the definition of the position of the template $\mathbf{T}$. In our application, this template image is a fixed patch that encompasses almost 80% of the image. Therefore, the goal of the ICIA algorithm as presented in [15] is to minimize:

$$\sum_x [\mathbf{T}(\mathbf{W}(\mathbf{x};\Delta\mathbf{p})) - \mathbf{I}(\mathbf{W}(\mathbf{x};\mathbf{p}))]^2 \tag{2}$$

Where $\mathbf{T}$ is the template image, $\mathbf{I}$ the current image, $\mathbf{x} = (x,y)^T$ represents the pixel coordinates, and $\mathbf{W}(\mathbf{x};\mathbf{p})$ is the motion model (in our case the homography) where $\mathbf{p} = (p_1, p_2.., p_8)^T$ is the vector of parameters that describes the transformation.

The increment to the parameters is found after a first-order Taylor series expansion as follows:

$$\Delta\mathbf{p} = \mathbf{H^{-1}} \sum_x \left[\nabla\mathbf{T}\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right]^T [\mathbf{I}(\mathbf{W}(\mathbf{x}:\mathbf{p})) - \mathbf{T}(\mathbf{x})] \tag{3}$$

where $\mathbf{H}$ is the Hessian matrix defined by:

$$\mathbf{H} = \sum_x \left[\nabla\mathbf{T}\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right]^T \left[\nabla\mathbf{T}\frac{\partial\mathbf{W}}{\partial\mathbf{p}}\right] \tag{4}$$

where $\nabla\mathbf{T} = \left(\frac{\partial\mathbf{I}}{\partial\mathbf{x}}, \frac{\partial\mathbf{I}}{\partial\mathbf{y}}\right)$ is the gradient of the template that is evaluated at $\mathbf{W}(\mathbf{x},\mathbf{0})$ (when the template is selected), and

$\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ is the Jacobian of the transformation. This Jacobian is also evaluated at $(\mathbf{x}; 0)$.

The advantage of this algorithm is that by changing the roles of images $\mathbf{T}$ and $\mathbf{I}$, $\mathbf{H}$ is constant. As can be seen in (4), there is no term that depends on the parameters $\mathbf{p}$, and as a consequence $\mathbf{H}$ is calculated at the beginning of alignment task.

Finally, the motion model is updated as follows:

$$\mathbf{W}(\mathbf{x}; \mathbf{p}) \leftarrow \mathbf{W}(\mathbf{x}; \mathbf{p}) \circ \mathbf{W}(\mathbf{x}; \Delta\mathbf{p})^{-1} \tag{5}$$

This algorithm iteratively estimates $\Delta\mathbf{p}$ until a stopping criteria is reached ($||\Delta\mathbf{p}|| \leq 10^{-5}$).

The ICIA algorithm allows us to find the motion parameters efficiently. However, this iterative algorithm relies on a linearization stage which is only valid when the range of motion is small. In different applications, this is rarely satisfied, especially when working with aerial image (vehicles' vibrations, low altitude flights). To alleviate this problem, MR methods are used considering, as mentioned in [17], that at low resolution, the vector of motion is smaller, so that long displacements can be better approximated using a propagation of parameters through the MR structure.

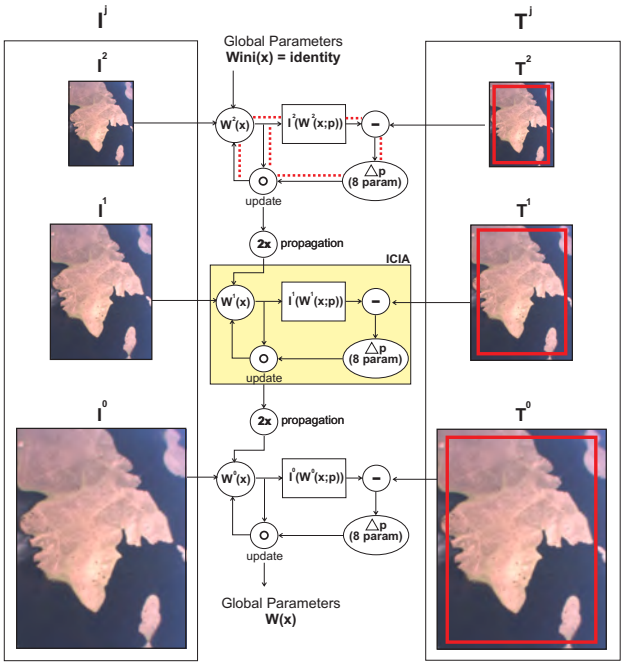

Fig. 1. Multi-Resolution ICIA. Images $\mathbf{I}$ and $\mathbf{T}$ are downsampled to create the MR structure. In each level, the ICIA is applied iteratively. In each iteration, $\mathbf{I}$ is warped, the error between the warped image and $\mathbf{T}$ is calculated, and the parameters are updated. When the stopping conditions are reached, the parameters are propagated to the next level. The process is repeated until the lowest level of the pyramid is reached (highest resolution image).

The MR-ICIA algorithm is described in Fig. 1. The current $\mathbf{I}$ and the reference $\mathbf{T}$ images are downsampled by a factor of 2, according to the number of levels defined in the pyramid, in order to create the MR structure.

The process starts at the lowest resolution level ($\mathrm{j} = \mathrm{j}_{\max}$). In this level, the ICIA algorithm iteratively finds the motion model $\mathbf{W}^{\mathrm{j}}$. The parameters of this motion model are propagated to the next level of the pyramid, as follows:

$$\begin{aligned} p_{\mathrm{i}}^{\mathrm{j}} &= p_{\mathrm{i}}^{\mathrm{j}-1} \quad for \quad \mathrm{i} = \{1, 2, 4, 5\} \\ p_{\mathrm{i}}^{\mathrm{j}} &= 2.0 * p_{\mathrm{i}}^{\mathrm{j}-1} \quad for \quad \mathrm{i} = \{3, 6\} \\ p_{\mathrm{i}}^{\mathrm{j}} &= \frac{p_{\mathrm{i}}^{\mathrm{j}-1}}{2.0} \quad for \quad \mathrm{i} = \{7, 8\} \end{aligned} \tag{6}$$

where the subscript i represents the parameters, and j represents the level of the pyramid.

The process is repeated until the lowest level of the pyramid is reached (highest resolution image). In this level, the parameters that best minimize the differences between images $\mathbf{I}$ and $\mathbf{T}$ are obtained.

III. HOMOGRAPHY-BASED POSE ESTIMATION

The method for estimating the absolute position and orientation is based on the homography decomposition method [18]. Considering that the vehicle is equipped with a downwards-looking camera, the homography induced by the ground plane is used to obtain the motion of the vehicle.

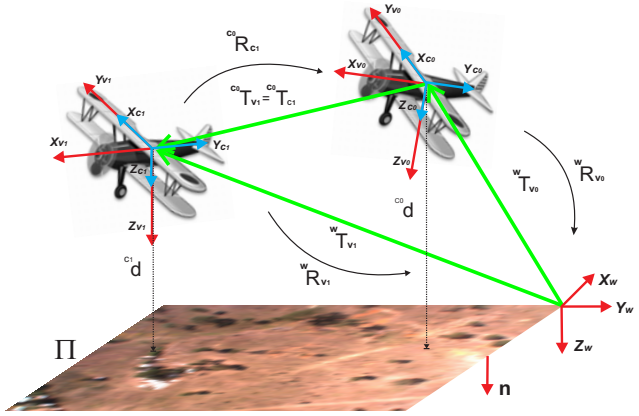

Fig. 2. Pose Estimation based on homographies.

- **Planar Homography**

If we consider two images, $\mathbf{I_0}$ and $\mathbf{I_1}$, of the same planar object at two different instants of time (the camera is moving), their corresponding camera coordinate systems are related by a rigid transformation, as follows:

$$^{\mathbf{c_1}}\mathbf{x} = {}^{\mathbf{c_1}}\mathbf{R}_{\mathbf{c_0}}\, {}^{\mathbf{c_0}}\mathbf{x} + {}^{\mathbf{c_1}}\mathbf{t}_{\mathbf{c_0}} \tag{7}$$

Where $^{\mathbf{c_0}}\mathbf{x}$, and $^{\mathbf{c_1}}\mathbf{x}$ are the coordinates of a 3D point $\mathbf{P}$ relative to each camera frame.

If the ground plane is characterized by its normal vector $^{\mathbf{c_0}}\mathbf{n}$ and its distance $^{\mathbf{c_0}}d$, and knowing that:

$$\begin{aligned} {}^{\mathbf{c_0}}\mathbf{n}^{\mathrm{T}}\, {}^{\mathbf{c_0}}\mathbf{x} &= n_1 x + n_2 y + n_3 z = {}^{\mathbf{c_0}}d \\ \frac{1}{{}^{\mathbf{c_0}}d}\, {}^{\mathbf{c_0}}\mathbf{n}^{\mathrm{T}}\, {}^{\mathbf{c_0}}\mathbf{x} &= 1 \end{aligned} \tag{8}$$

then, equation (7) can be expressed as:

$$^{c_1}\mathbf{x} = \left(^{c_1}\mathbf{R}_{c_0} + \frac{1}{^{c_0}d} ^{c_1}\mathbf{t}_{c_0} ^{c_0}\mathbf{n}^{\mathrm{T}} \right) ^{c_0}\mathbf{x} \quad (9)$$

Where

$$\mathbf{H_e} = ^{c_1}\mathbf{R}_{c_0} + \frac{1}{^{c_0}d} ^{c_1}\mathbf{t}_{c_0} ^{c_0}\mathbf{n}^{\mathrm{T}} \quad (10)$$

is the euclidean homography matrix that denotes a linear transformation from $^{c_0}\mathbf{x} \in R^3$ to $^{c_1}\mathbf{x} \in R^3$ [18]. This matrix depends on the motion parameters ($^{c_1}\mathbf{R}_{c_0}$, $^{c_1}\mathbf{t}_{c_0}$) as well as on the structure parameters ($^{c_0}\mathbf{n}^{\mathrm{T}}$, ^{c_0}d) of the plane Π. Considering the scale ambiguity in the term $\frac{1}{^{c_0}d} ^{c_1}\mathbf{t}_{c_0}$, it is expected to recover from $\mathbf{H_e}$ the ratio of the translation scaled by the distance ^{c_0}d.

If we consider the pinhole camera model, equation (9) can be expressed in terms of the image coordinates as:

$$^{I_1}\mathbf{x} = \gamma \mathbf{K} \ (^{c_1}\mathbf{R}_{c_0} + \frac{1}{^{c_0}d} ^{c_1}\mathbf{t}_{c_0} ^{c_0}\mathbf{n}^{\mathrm{T}}) \ \mathbf{K^{-1}} \ ^{I_0}\mathbf{x} \quad (11)$$

Where $\gamma = \frac{^{c_1}z}{^{c_0}z}$, $\mathbf{K}$ the calibration matrix, and

$$\mathbf{H_p} = \gamma \mathbf{K} (^{c_1}\mathbf{R}_{c_0} + \frac{1}{^{c_0}d} ^{c_1}\mathbf{t}_{c_0} ^{c_0}\mathbf{n}^{\mathrm{T}}) \mathbf{K^{-1}} \quad (12)$$

Equation (12) is also a homography (projective homography), that allows to map points from image $\mathbf{I_0}$ to points in image $\mathbf{I_1}$. From (12), the homography in the euclidean space can be calculated as shown in (13) using the intrinsic camera parameters $\mathbf{K}$, and recovering the scale factor as $\gamma = med(svd(\mathbf{H_L}))$ [19].

$$\begin{aligned} \mathbf{H_L} &= \mathbf{K}^{-1}\mathbf{H_p}\mathbf{K} \\ \mathbf{H_e} &= \frac{\mathbf{H_L}}{\gamma} = (^{c_1}\mathbf{R}_{c_0} + \frac{1}{^{c_0}d} ^{c_1}\mathbf{t}_{c_0} ^{c_0}\mathbf{n}^{\mathrm{T}}) \end{aligned} \quad (13)$$

- **Pose Estimation from Homography Decomposition**

The pose estimation algorithm assumes that a camera, in a downwards-looking configuration, is located on-board an aerial vehicle, and the position of its coordinate system $\mathbf{X_c}$ coincides with the vehicle's body frame $\mathbf{X_v}$, as shown in Fig. 2. Thus, the transformation from $\mathbf{X_c}$ to $\mathbf{X_v}$ is defined by a fixed rotation $^{v}\mathbf{R_c}$ of 90° around the $\mathbf{Z}$ axis.

The algorithm also assumes that in $\mathrm{t}(0)$, when the algorithm starts, an initial estimation of the position $^{w}\mathbf{t}_{v_0}$ and orientation of the vehicle $^{w}\mathbf{R}_{v_0}$ with respect to a world coordinate system are known (e.g. GPS/IMU estimation).

Hence, the inter-frame motion of the aerial vehicle can be estimated by decomposing the euclidean homography (10) obtained with the MR-ICIA algorithm, using the method described in [18]. This decomposition gives four solutions. The correct solution is chosen, assuming the positive depth constraint, and considering that the ground plane is not slopped $\mathbf{n} = [0, 0, 1]^{\mathrm{T}}$.

Therefore, the method recovers the rotation matrix $^{c_0}\mathbf{R}_{c_1}$ and the scaled translation vector $\frac{^{c_0}\mathbf{t}_{c_1}}{^{c_0}d}$. The absolute translation $^{c_0}\mathbf{t}_{c_1}$ can be recovered using a measured or calculated height, as was shown in [10].

With all this information known, the pose with respect to the world coordinate system is found as follows. The absolute rotation $^{w}\mathbf{R}_{v_1}$ is found as:

$$^{w}\mathbf{R}_{v_1} = ^{w}\mathbf{R}_{v_0} ^{v_0}\mathbf{R}_{c_0} ^{c_0}\mathbf{R}_{c_1} ^{c_1}\mathbf{R}_{v_1} \quad (14)$$

where, $^{v_0}\mathbf{R}_{c_0} = ^{v}\mathbf{R_c}$ and $^{c_1}\mathbf{R}_{v_1} = ^{v}\mathbf{R_c^{\mathrm{T}}}$

From the rotation matrix $^{w}\mathbf{R}_{v_1}$, the Euler angles roll (ϕ), pitch (θ), and yaw (ψ), can be obtained as shown in (15). Assuming that $\mathbf{R} = ^{w}\mathbf{R}_{v_1}$, then:

$$\begin{aligned} \theta &= \mathrm{atan2}(-\mathrm{R}_{31}, \sqrt{\mathrm{R}_{32}^2 + \mathrm{R}_{33}^2}) \\ \psi &= \mathrm{atan2}(\mathrm{R}_{32}, \mathrm{R}_{33}) \\ \phi &= \mathrm{atan2}(\mathrm{R}_{21}, \mathrm{R}_{11}) \end{aligned} \quad (15)$$

The translation vector $^{w}\mathbf{t}_{v_1} = ^{w}\mathbf{t}_{c_1}$ is recovered as:

$$^{w}\mathbf{t}_{v_1} = ^{w}\mathbf{R}_{c_0} ^{c_0}\mathbf{t}_{c_1} + ^{w}\mathbf{t}_{c_0} \quad (16)$$

where $^{w}\mathbf{R}_{c_0} = ^{w}\mathbf{R}_{v_0} ^{v_0}\mathbf{R}_{c_0}$.

When a new image is analyzed. If the distance to the plane is not available from an on-board sensor (e.g. altimeter), this distance can be calculated from the previous data as:

$$^{c_1}d = ^{c_0}d + ^{c_0}\mathbf{t}_{c_1} \cdot ^{c_0}\mathbf{n} \quad (17)$$

This distance is then used to recover the absolute translation from frame 1 to frame 2. The process is repeated after this, and the estimation of the pose is propagated.

IV. EXPERIMENTS AND RESULTS

Different tests have been conducted to analyze the performance of the algorithm. The analysis is done in three different stages of the flight: take-off, cruise, and landing. The evaluation of the results is presented in terms of the RMSE (Root Mean Square Error), comparing the obtained vision data with the data obtained by other on-board sensors (GPS/IMU). On the other hand, an analysis of the final frame rate of the algorithm is presented.

A. Experimental setup

- Data collection

The data used in the experiments corresponds to a flight of more than 90 minutes in duration. The image data was collected by the Airbone System Laboratory (ASL) [20]. This laboratory consists of a Cessna 172, as shown in Fig 3, owned and operated by the Australian Research Centre of Aerospace Automation (ARCAA).

The ASL is capable of capturing images from the on-board cameras at a rate of 30 HZ with a resolution of 1024×768 using the open-source Videography software package [21].

Fig. 3. Airbone System Laboratory (ASL). This laboratory consists of a Cessna 172 equipped with an x86 computer running Linux; a NovAtel SPAN, which computes a tightly-coupled GPS/INS solution for position and attitude data; two cameras: one pointing forwards and the other one pointing downwards; and custom electronics for data synchronization. The ASL is capable of capturing images at a rate of 30 HZ with a resolution of 1024×768.

Data is recorded from an on-board camera in a downwards-looking configuration. This camera is externally triggered, allowing the captured images to be precisely timestamped. The position and attitude data used as ground truth is obtained by an on-board GPS/INS system.

The flight path was chosen in order to maximize the terrain variability (mountains, flat terrains, sea). From the available data, a collection of images that correspond to three stages of the flight was selected, these stages being: take-off, cruise, and landing. Fig. 4 shows some of the images selected for the tests. As can be seen there, the strong variability of the terrain, and the different conditions of the flight (different heights) represent a big challenge to the visual algorithm.

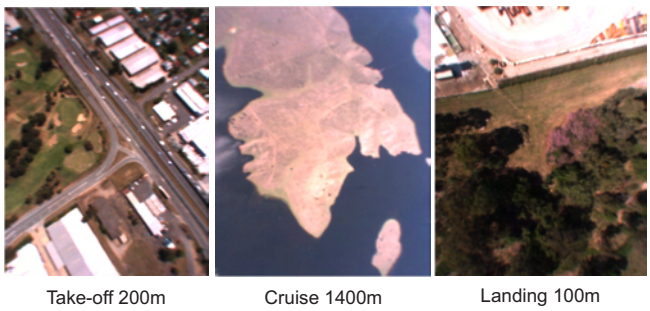

Fig. 4. Image data. A collection of images that correspond to three stages of the flight were selected for the tests: take-off, cruise, and landing. As can be seen, the variability of the data (mountains, flat terrains, sea) and the different heights play an important role for testing the proposed strategy, and represent a big challenge for the visual algorithm.

- Vision algorithm setup

Four pyramid levels are used in the tests. In each level, the number of iterations and the number of parameters are fixed: 100 iterations per level, and 8 parameters are estimated in each level. Two termination criteria are used. Criteria $\mathbf{T_1}$: the minimum is reached when the increment of the parameters is below a threshold (10^{-5}). Criteria $\mathbf{T_2}$: the minimum is reached if the mean error does not decrease after a defined number of iterations (10 iterations). On the other hand, a template that covers approximately 80% percent of the image is used, although in each level the number of pixels that are considered in the minimization process vary according to the level: in the highest level (lowest resolution) all the pixels are used; however, in the lowest level (highest resolution) 1 of every 5 pixels is employed. With this criteria, real-time frame rates are achieved without compromising the accuracy of the estimation.

The algorithm was developed in $C++$ and the OpenCV libraries [22] were used for managing image data.

B. Take-off Mode

The MR-ICIA algorithm is used to register pairs of images, and the homography found in the registration is decomposed in order to obtain the relative position and orientation of the vehicle. This process is integrated in time to obtain the vehicles' state. We compare our estimation (red/dark line) with the truth data (green/light line) in Fig 5. From this figure, we can observe that the errors that were obtained are relatively low, considering that the tests correspond to approximately 1 minute of flight and that the estimation is based only on visual information: the RMSEs in position are $[152.69, 273.93, 50.6591]$ m for X, Y, and Z respectively, and the RMSEs in orientation $[3.72, 1.17, 1.58]$ degrees. for roll, pitch, and yaw angles. This will allow to maintain a good approximation of the vehicle's state while the system recovers from a GPS drop-out.

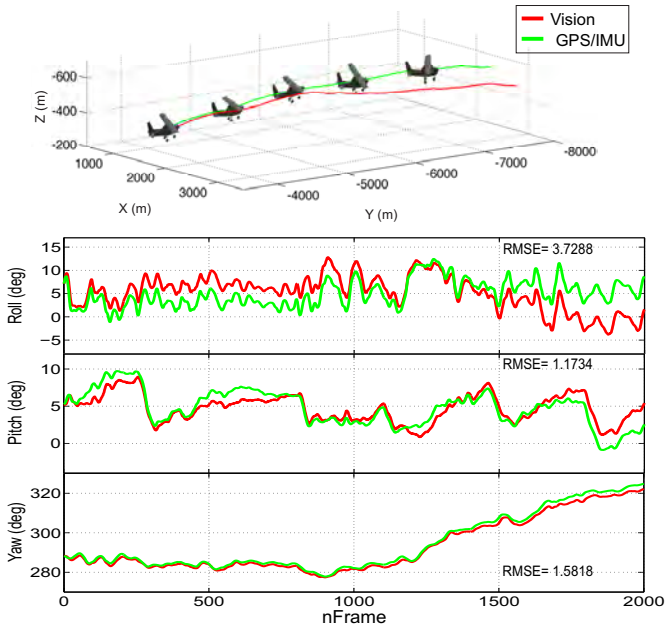

Fig. 5. Results of the take-off stage. The visual estimation (red/dark line) is compared with the GPS/IMU (green/light line) data. As can be seen, the signals have a similar behavior. The normal drift due to the integration is also present. However, its rate is low, allowing to maintain a good approximation of the vehicle's state while the system recovers from a GPS drop-out.

C. Cruise Mode

Fig. 6 shows the results of the algorithm in the cruise stage. The visual estimation (red/dark line) shows a behavior that is

similar to that of the GPS/IMU data (green/light line).

The RMSEs obtained in the angles' estimation are in the range of 4 degrees, and the RMSEs in position are $[376.27, 467.15, 60.42]$ m for X, Y, and Z respectively. The previously mentioned errors in position can be considered low if we take into account the total traversed distance and that the mean speed was $\approx$ 211 km/hr. Therefore, those errors represent 7% and 4% of the total traversed distance for the X and Y axes, respectively (6470 m and 7700 m).

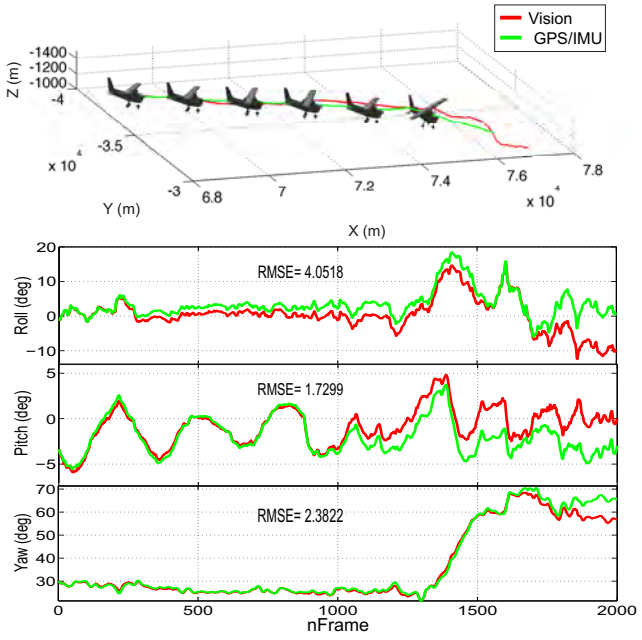

Fig. 6. Results of the cruise stage. The visual estimation (red/dark line) is compared with the GPS/IMU (ligth/dashed line) data. The upper figure corresponds to the 3D position of the vehicle, and the other figures show the estimation of the rotation angles. The errors that were obtained are relatively low considering that the tests correspond to approximately 1 minute of flight.

D. Landing Mode

During the landing stage (see Fig. 7), the vision algorithm was able to estimate the vehicle's state until the vehicle reached a height of 60 m. From that point on (taking into account the characteristics of the on-board camera), the visual estimation was not robust. This happened because after the vehicle reached that height, the conditions of the terrain were not the appropriate ones for the registration algorithm (there was not enough texture information), and because when the vehicle was close to the ground there was not a common frame-to-frame visual information that allowed the estimation of the vehicle's relative state. Therefore, the performance was degraded during this stage.

For this test, the obtained RMSEs in orientation are $[2.4, 3.9, 5]$ degrees for roll, pitch, and yaw angles; and in position the RMSEs are $[468.13, 150.68, 10.91]$ m for X, Y, and Z, respectively.

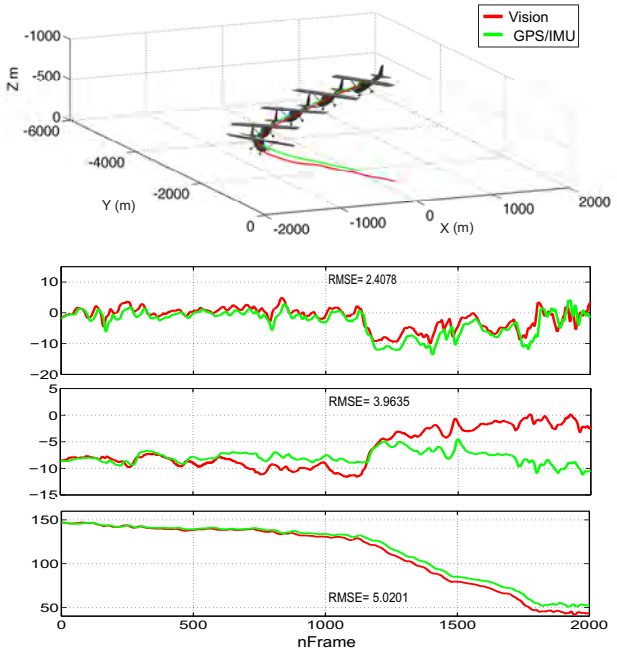

Fig. 7. Results of the landing stage. The visual estimation (red/dark line) is compared with the GPS/IMU (green/light line) data during the landing stage. As can be seen, the signals have a similar behavior, the vision algorithm was able to estimate the vehicle's state until reaching a height of 60 m, and the RMSEs obtained in the attitude estimations are low.

E. Discussion

In general, the tests show that the vision-based proposed strategy obtains an adequate estimation of the state of the vehicle. In terms of errors, it can be seen that depending on the stage of the flight, the errors in position are stable, and also low if the total traversed distance and the speed $(150 - 200$ km/hr) are considered in the analysis of the results. Therefore, the visual system can offer a good approximation of the vehicles' state when it is so required. The same situation is reflected in the estimation of the vehicles' attitude, where the obtained errors were $< 5^\circ$ during all the stages (see Fig. 5, Fig. 6, and Fig. 7), being all the results obtained for a flight of more than 1 minute.

The limitations of the system are related to the kind of terrain analyzed and the configuration of the on-board camera (camera field of view). The algorithm requires texture information to register pairs of images, and from the tests it was found that during take-off and most of the cruise sequences high texture information was available. However, in the last stage of the landing (when the altitude was lower than 60 m), it is seen that due to the low texture information in the sequence, and also due to the current configuration of the system (there is not common information in consecutive images), the image registration technique is not able to find the appropriate homography between images.

In terms of performance, it has been shown that the adopted image registration strategy (MR-ICIA + variation of the number of pixels used in each level of the pyramid) allowed to achieve a pose estimation at real-time frame rates (12 fps),

without compromising the accuracy of the estimation. This speed can be improved depending on the application and the degree of precision the system requires: a faster speed can be obtained by using lower image sizes (the results were obtained with an image resolution of 1024×768) and by estimating different parameters in the different levels.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we have presented an algorithm for pose estimation of aerial vehicles based on direct methods and homography decomposition techniques. We have shown that by using the proposed strategy, direct methods can be employed to obtain real-time frame-to-frame motion estimations making an optimal use of all the available information in the image.

Real flight data was used to analyze the performance of the algorithm during representative stages of a flight: cruise, landing, and take-off, where two of those stages can be considered critical (take-off and landing).

The results show a good correlation of the visual estimation with the GPS/IMU data that validates the proposed strategy and makes it useful to provide valid data of the aircraft's state (e.g. during GPS dropouts).

In terms of performance, the adopted image registration strategy (MR-ICIA + variation of the number of pixels used in each level of the pyramid) allowed a pose estimation at real-time frame rates: 12 fps, without compromising the accuracy of the estimation. Nonetheless, future work will focus on improving this speed by adopting a dynamic strategy in terms of the number of parameters that are estimated in the hierarchical structure.

By using direct methods, the vision-based data has the advantage of drifting slowly in time, providing a good approximation of the vehicle's state during long periods of time (minutes) based only on visual information. Taking this into account, future work will focus on filtering the visual estimation to improve its robustness, especially when flying in places with low texture information, integrating the obtained visual data with other sensors, and will also focus on the implementation of the system on-board a UAV.

VI. ACKNOWLEDGMENT

This paper is the result of a collaborative research program between the Computer Vision Group and the Australian Research Centre for Aerospace Automation (ARCAA). This work has been supported by the European Commission under the FP7-PEOPLE-IRSES-2008 grant (ICPUAS International Cooperation Program for Unmanned Aerial Systems Research and Development). The authors would like to thank the Universidad Politécnica de Madrid for one of the Authors' Ph.D. Scholarship. Additionally, the authors would like to thank Damien Dusha from the ARCAA for the collection of the data used in the tests.

REFERENCES

[1] Luis Mejias, Scott McNamara, John Lai, and Jason J. Ford. Vision-based detection and tracking of aerial targets for UAV collision avoidance. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2010.

[2] Pascual Campoy, Juan Correa, Iván Mondragón, Carol Martínez, Miguel Olivares, Luis Mejias, and Jorge Artieda. Computer Vision Onboard UAVs for Civilian Tasks. *Journal of Intelligent and Robotic Systems*, 54:105–135, 2009. 10.1007/s10846-008-9256-z.

[3] Luis Mejias, Srikanth Saripalli, Pascual Campoy, and Gaurav S. Sukhatme. Visual servoing of an autonomous helicopter in urban areas using feature tracking. *Journal of Field Robotics*, 23, 2006.

[4] Srikanth Saripalli, James F. Montgomery, and Gaurav S. Sukhatme. Vision-based Autonomous Landing of an Unmanned Aerial Vehicle. In *IEEE International Conference on Robotics and Automation (ICRA*, pages 2799–2804, 2002.

[5] Carol Martinez, Ivan Mondragon, Miguel Olivares-Mendez, and Pascual Campoy. On-board and Ground Visual Pose Estimation Techniques for uav Control. *Journal of Intelligent & Robotic Systems*, 61:301–320, 2011. 10.1007/s10846-010-9505-9.

[6] Jorge Artieda, José Sebastian, Pascual Campoy, Juan Correa, Iván Mondragón, Carol Martínez, and Miguel Olivares. Visual 3-D SLAM from UAVs. *Journal of Intelligent & Robotic Systems*, 55:299–321, 2009. 10.1007/s10846-008-9304-8.

[7] F. Caballero, L. Merino, J. Ferruz, and A. Ollero. Vision-Based Odometry and SLAM for Medium and High Altitude Flying UAVs. *Journal of Intelligent and Robotic Systems*, 54:137–161, 2009. 10.1007/s10846-008-9257-y.

[8] F. Caballero, L. Merino, J. Ferruz, and A. Ollero. A visual odometer without 3D reconstruction for aerial vehicles. Applications to building inspection. In *IEEE International Conference on Robotics and Automation (ICRA 2005)*., 2005.

[9] Gianpaolo Conte and Patrick Doherty. Vision-based unmanned aerial vehicle navigation using geo-referenced information. *EURASIP J. Adv. Signal Process*, 2009:10:1–10:18, January 2009.

[10] K. Kaiser, N. Gans, and W. Dixon. Position and Orientation of an Aerial Vehicle through Chained, Vision-Based Pose Reconstruction. *Proc. AIAA Conf. on Guidance, Navigation and Control*, 2005.

[11] Jonathan Kelly, Srikanth Saripalli, and Gaurav S. Sukhatme. Combined Visual and Inertial Navigation for an Unmanned Aerial Vehicle. In *Proc. 6th Int'l Conf. Field and Service Robotics (FSR'07)*, Chamonix, France, July 2007.

[12] Michal Irani and P. Anandan. About Direct Methods. In *ICCV '99: Proceedings of the International Workshop on Vision Algorithms*, pages 267–277, London, UK, 2000. Springer-Verlag.

[13] Philip H. S. Torr and Andrew Zisserman. Feature based methods for structure and motion estimation. In *ICCV '99: Proceedings of the International Workshop on Vision Algorithms*, pages 278–294, London, UK, 2000. Springer-Verlag.

[14] Bryce Ready and Clark Taylor. Inertially Aided Visual Odometry for Miniature Air Vehicles in GPS-denied Environments. *Journal of Intelligent and; Robotic Systems*, 55:203–221, 2009. 10.1007/s10846-008-9294-6.

[15] Simon Baker and Iain Matthews. Equivalence and Efficiency of Image Alignment Algorithms. In *Proceedings of the 2001 IEEE Conference on Computer Vision and Pattern Recognition*, volume 1, pages 1090 – 1097, December 2001.

[16] Simon Baker, Ankur Datta, and Takeo Kanade. Parameterizing Homographies. Technical Report "CMU-RI-TR-06-11", "Robotics Institute", "Pittsburgh, PA", "March" "2006".

[17] James R. Bergen, P. Anandan, Th J. Hanna, and Rajesh Hingorani. Hierarchical model-based motion estimation. pages 237–252. Springer-Verlag, 1992.

[18] Yi Ma, Stefano Soatto, Jana Kosecka, and S. Shankar Sastry. *An Invitation to 3-D Vision: From Images to Geometric Models*. SpringerVerlag, 2003.

[19] Ezio Malis and Manuel Vargas. Deeper understanding of the homography decomposition for vision-based control. Research Report RR-6303, INRIA, 2007.

[20] Duncan G. Greer, Rhys Mudford, Damien Dusha, and Rodney Walker. Airbone systems laboratory for automation research. In 27^{TH} *International Congress of the Aeronautical Sciences, ICAS 2010*, Nice, France.

[21] Videography. http://videography.sourceforge.net/, 2010.

[22] Gary Bradski and Adrian Kaehler. *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly, 2008.