

HMPMR strategy for real-time tracking in aerial images, using direct methods

Carol Martínez · Pascual Campoy · Iván F. Mondragón ·
José Luis Sánchez-López · Miguel A. Olivares-Méndez

Received: 3 April 2012 / Revised: 27 December 2013 / Accepted: 14 April 2014 / Published online: 1 June 2014
© Springer-Verlag Berlin Heidelberg 2014

Abstract The vast majority of approaches make use of features to track objects. In this paper, we address the tracking problem with a tracking-by-registration strategy based on direct methods. We propose a hierarchical strategy in terms of image resolution and number of parameters estimated in each resolution, that allows direct methods to be applied in demanding real-time visual-tracking applications. We have called this strategy the Hierarchical Multi-Parametric and Multi-Resolution strategy (HMPMR). The Inverse Composition Image Alignment Algorithm (ICIA) is used as an image registration technique and is extended to an HMPMR-ICIA. The proposed strategy is tested with different datasets and also with image data from real flight tests using an Unmanned Aerial Vehicle, where the requirements of direct methods are easily unsatisfied (e.g. vehicle vibrations). Results show that using an HMPMR approach, it is possible to cope with the efficiency problem and with the small motion constraint of direct methods, conducting the tracking task at real-time frame rates and obtaining a performance that is comparable to, or even better than, the one obtained with the other algorithms that were analyzed.

Keywords Visual tracking · Image registration · Hierarchical methods · UAVs · Parametric motion

1 Introduction

The development of visual algorithms for Unmanned Aerial Vehicles (UAVs) can be considered a challenging task. Real-time requirements (e.g. for control tasks), outdoors operations (non-structured environments), vehicle vibrations, and limited computational capacity on-board are some examples of the problems found when working with UAVs. In the literature, different strategies have been presented to solve the tracking problem in aerial images. Most of the strategies are based on feature-based methods [10, 32, 33, 43, 48].

In the application of tracking on-board UAVs (see Fig. 1) we have seen that the adopted feature-based strategies are very sensitive to strong motions (e.g. vehicle vibrations and strong 3D changes), being it difficult to strike a balance between achieving real-time and accurate estimations (defining a specific number of good features to track without increasing the processing time). Although multi-resolution (MR) approaches [5] can help coping with strong and large motion problems, constant vehicle vibrations, low computational capacity available on-board, and delays in the communication (when images are processed on the ground) make the MR strategies insufficient to overcome these problems. Additionally, we have observed that when using feature-based methods under strong motions, the accumulation of errors makes the tracking algorithm fail after a few frames, affecting and making on-line tests difficult.

Conversely, direct methods have the advantage that the motion and the matching of the pixels are found without

Electronic supplementary material The online version of this article (doi:10.1007/s00138-014-0617-2) contains supplementary material, which is available to authorized users.

C. Martínez (✉) · P. Campoy · I. F. Mondragón ·
J. L. Sánchez-López · M. A. Olivares-Méndez
Computer Vision Group, Universidad Politécnica de Madrid,
Centro de Automática y Robótica (CAR) UPM-CSIC, C/José
Gutiérrez Abascal 2, 28006 Madrid, Spain
e-mail: carolviviana.martinez@upm.es

I. F. Mondragón
Department of Industrial Engineering, Faculty of Engineering,
Pontificia Universidad Javeriana, Bogotá D.C., Colombia

M. A. Olivares-Méndez
Automation Research Group, Interdisciplinary Centre for Security,
Reliability and Trust (SnT), University of Luxembourg,
Luxembourg, Luxembourg

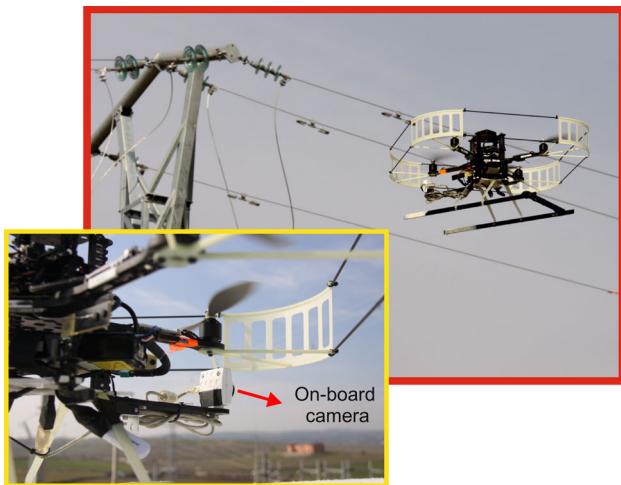


Fig. 1 Tracking on-board UAVs. Robust real-time tracking allows to expand the vehicle's capabilities, such as landing, visual inspection, or coping with vulnerabilities of other on-board sensors (e.g. GPS drop-outs)

intermediate steps using the intensity information of all the pixels in the image template (i.e. the object to track). However, the drawbacks of direct methods are that they rely on some constraints [22] (e.g. small frame-to-frame motions) and that their speed is highly dependent on the number of pixels in the template and the registration method used, being it sometimes difficult to achieve real-time frame rates. That is why in the majority of situations, feature-based methods are preferable to direct methods.

For the application of tracking on-board UAVs, we propose to address the tracking problem with a tracking-by-registration strategy [11], based on direct methods, where the object of interest is tracked by iteratively registering pairs of consecutive images. We propose to use the Inverse Compositional Image Alignment Algorithm (ICIA) [3] as an image registration technique, to deal with the efficiency problem of direct methods, but extending it in a hierarchical strategy in terms of image resolution and number of parameters estimated in each resolution: the Hierarchical Multi-Parametric and Multi-Resolution strategy (HMPMR), to deal with the small motion constraint of direct methods. Using this strategy (HMPMR), the tracking task is improved in situations where MR approaches are insufficient to cope with large frame-to-frame motions without compromising the real-time operation required in on-line applications.

In the literature, to the authors' knowledge, the HMPMR strategy has neither been used for tracking nor for on-line tracking on-board autonomous vehicles. In [36], an HMPMR strategy has been applied for frame-to-frame alignment in image mosaicing. However, an important issue that we address in this paper is the characterization of the HMPMR strategy: how the selection of the multi-parametric structure affects the performance of the algorithm, the definition of cri-

teria to select both MR and MP hierarchies, and an analysis of advantages and limitations of using the HMPMR strategy, in this case for visual tracking; and none of these aspects have been addressed before.

Therefore, the main contributions of this paper are

- Casting [36] into an inverse compositional framework.
- Presenting a detailed analysis of the HMPMR strategy, characterizing the different components of this framework (image pyramid, influence of the combination of parameters in the MR structure, criteria to select the MR and the MP hierarchies).
- Proposing a confidence measurement of reliability of the tracker based on the template image and the number of parameters that can be estimated with this information.
- Extending the ICIA algorithm to an HMPMR-ICIA algorithm.
- Applying this strategy to solve a challenging task: visual tracking for aerial vehicles. Therefore, extending the use of direct methods in real-time applications.
- Presenting a classification of the hierarchical approaches found in the literature.

The paper is organized as follows: Sect. 2 presents a description of related work and a classification of the different hierarchical approaches found in the literature. In Sect. 3, the general idea of the tracking-by-registration strategy, based on direct methods, using the ICIA algorithm is presented. The details of the proposed HMPMR strategy for tracking and its configuration are presented in Sect. 4. In this section, the HMPMR-ICIA algorithm is described. In Sect. 5 the HMPMR strategy is analyzed, and Sects. 6 and 7 present results of the HMPMR-ICIA algorithm tracking different templates. Section 8 deals with the discussion of the results; and finally, in Sect. 9, conclusions and the direction of future work are presented.

2 Related work

There are different ways the tracking task can be conducted. Two common ways are based on image registration (tracking-by-registration) and based on detection algorithms (tracking-by-detection) [25]. In [11], the term tracking-by-registration was introduced, and although tracking and registration (estimating the transformation that aligns two images of the same scene estimating pixel-to-pixel correspondence [50]) are usually seen as two different strategies, they can be used simultaneously for object tracking, in which case the object of interest is tracked by iteratively registering pairs of consecutive images.

Image registration algorithms can be classified according to the kind of information used: based on direct methods or based on features. Feature-based approaches [44] extract

local features (e.g. points) and track them by matching features between frames. These methods rely on the detection of distinguishable image features and the matching method employed (e.g. such as RANSAC). They tend to be robust to scale changes, rotations and translations, and to illumination variations. Conversely, direct methods [22], also called pixel-based methods, estimate the transformation that aligns (register) the images by minimizing an error measurement that is based on the intensity values of the image.

In this paper, a tracking-by-registration algorithm based on direct methods is used for tracking objects. The image registration process consists in aligning two images: the reference image (that contains the object to track) and the current image, by finding the transformation that best aligns them. This transformation is normally found iteratively by minimizing the sum of squared differences (SSD) between the reference image and the current image [41] using different minimization methods. Nonetheless, the gradient descent optimization method (based on a first-order Taylor series approximation of the SSD) is one of the most widely used approaches because of its efficiency [4, 15, 28].

Gradient descent approaches can be classified depending on the update rule of the parameters. In [4], the different approaches were classified as follows: forwards additive [28], forwards compositional [40], inverse additive [15], and inverse compositional [3]. In the inverse compositional approach (where the ICIA algorithm we use in our application belongs to), the roles of the image and the template are inverted, whereupon a more efficient algorithm is obtained because the Jacobian of the warping function and the inversion of the Hessian matrix are calculated once at the beginning of the tracking task and every time the template image changes (not in every iteration, as the forwards additive approaches do).

Under large frame-to-frame motions, these image registration techniques require the use of hierarchical methods to help finding the transformation that aligns the reference and the current images. We have classified the different hierarchical approaches found in the literature in five classes, according to the kinds of hierarchies they have. Some of these approaches use feature-based methods, while others use direct methods. They can be classified as follows:

- The Multi-Resolution approach (MR): this is one of the most widely used approaches. In this strategy, the same motion model is estimated in each level of the image pyramid (the MR pyramid). This scheme has been used for different applications: for tracking planar objects [29], for detecting and tracking moving objects [23], for registration [46], and for recovering scene structure [16], among other applications [26].
- The Multi-Parametric approach (MP): in it, a hierarchy of parameters is applied using the same resolution of the

image. In the different fields where this strategy has been used, the hierarchy of 2D transformations has always been employed. With this strategy, an initial registration is found by first recovering the translation motion model, to reduce the separation of the images; and then the estimation is refined by increasing the complexity of the motion model progressively [9, 23, 35].

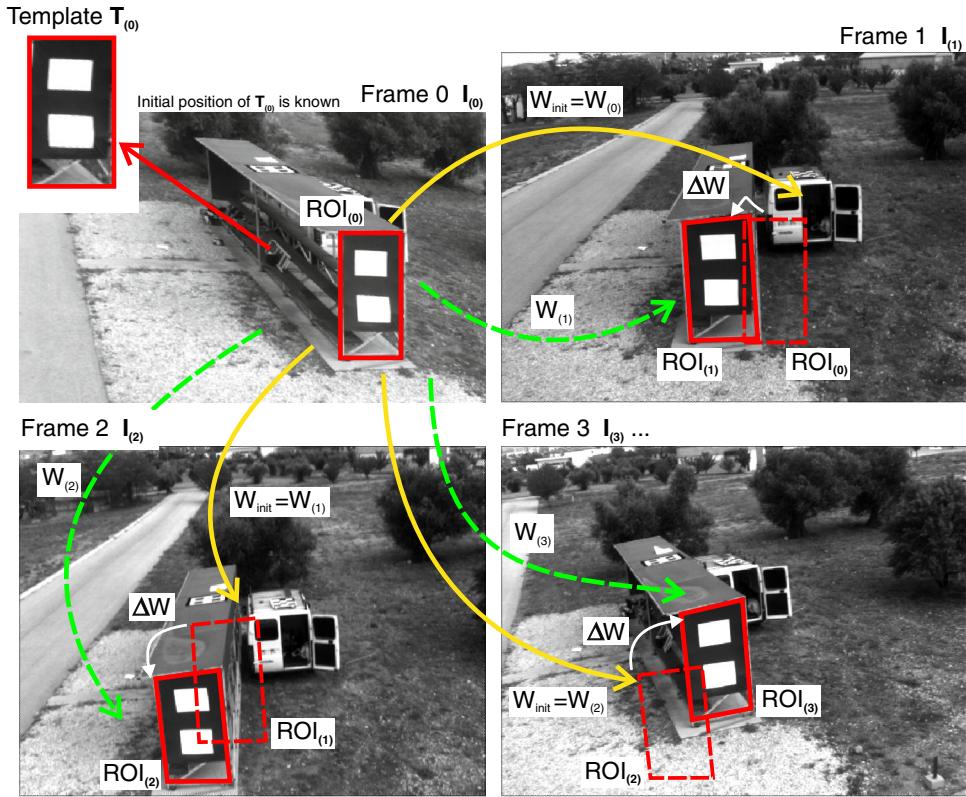
- Multi-Resolution with priors (MRp): in this strategy, before applying the MR approach, an initial estimation of the parameters is obtained applying either an initial algorithm [45] or information from additional sensors [21, 38].
- The Hierarchical Multi-Parametric Multi-Resolution approach (HMPMR): in this approach, different parameters are estimated inside an MR pyramid. In [36], this strategy is used for image mosaicing. They use an MP scheme and an MR pyramid. The number of pixels employed in the optimization process depends on the number of parameters estimated. In the case of translation, a region of interest (ROI) of around $\frac{1}{3}$ of the input image is used. For the affine motion model, $\frac{2}{3}$ of the image are used; and the full image is used for the homography.
- Hierarchical Multi-Parametric Multi-Resolution with priors (HMPMRp): where an initial estimation of the motion is obtained by either an initial algorithm or using additional sensors, and then different parameters are estimated inside an MR pyramid. An example of this strategy can be found in [37] for building image mosaics, where the normalized cross-correlation (NCC) method is used to find an initial 2D translation. This initial estimation is used as an initial estimation of a coarse-to-fine search of more complex motion models in an MR pyramid. On the other hand, in [12] the global motion from image sequences is estimated by first finding the 2D translation using an n-step search matching algorithm, and then using that estimation in an MR pyramid to refine the estimation.

3 2-D tracking-by-registration based on direct methods

The goal of a 2D tracking-by-registration strategy, based on direct methods, is to find the 2D position of an object in the image plane in each frame of an image sequence, using the intensity values of the object as visual features to estimate the motion. Assuming that an initial position of the object is known, that the motion between frames is small, that the appearance of the object does not change over time (the direct methods' constraints [22]), and that the 3D displacement of the object can be modeled by a 2D transformation (translation, affine, homography [17]); then the tracking task can be formulated as an incremental image registration task, as shown in Fig. 2.

A reference image (the object to track, $\mathbf{T}_{(0)}$) is defined in the first frame (Fig. 2, Frame 0, upper-left image),

Fig. 2 An example of a 2D tracking-by-registration strategy. The object to track is defined in the first frame $\mathbf{T}_{(0)}$ (red box, $\text{ROI}_{(0)}$). When a new frame is analyzed, e.g. $\mathbf{I}_{(1)}$ (upper-right image), the motion $\mathbf{W}_{(1)}$ between $\mathbf{T}_{(0)}$ and $\mathbf{I}_{(1)}$ (green/dashed arrow) is found by an image registration technique, assuming that an initial estimation of the motion \mathbf{W}_{init} is known (yellow/solid arrow). With $\mathbf{W}_{(1)}$, the position of the object in Frame 1 is found (i.e. $\text{ROI}_{(1)}$). Then, $\mathbf{W}_{(1)}$ is propagated to the next frame as an initial estimation of the motion in Frame 2 $\mathbf{W}_{\text{init}} = \mathbf{W}_{(1)}$. The process is repeated in each frame, and therefore the object is tracked (color figure online)



either manually or automatically by detection algorithms. This reference image corresponds to a sub-image or $\text{ROI}_{(0)}$ (Region of Interest), called image template, defined in the first frame $\mathbf{I}_{(0)}$ (the subscript represents the number of the frame).

When a new frame is analyzed $\mathbf{I}_{(1)}$ (Fig. 2, Frame 1, upper-right image), the motion $\mathbf{W}_{(1)}$ between $\mathbf{T}_{(0)}$ and the $\mathbf{I}_{(1)}$ (Fig. 2, Frame 1, green/dashed arrow) is found by an image registration technique, assuming that an initial estimation of the motion \mathbf{W}_{init} is known (Fig. 2, Frame 1, yellow/solid arrow). When an initial estimation is not known, this initial estimation can be assumed as the identity matrix, assuming that the frame-to-frame motion is small.

Therefore, the image registration algorithm is in charge of estimating the incremental motion model $\Delta\mathbf{W}$, in every iteration. Thus, the motion $\mathbf{W}_{(1)}$ is estimated, and as a consequence of this, the $\text{ROI}_{(1)}$ is found, i.e. the position of the object to track in the current frame. Then, the estimated motion $\mathbf{W}_{(1)}$ (Fig. 2, Frame 1, green/dashed arrow) is propagated to the next frame, as an initial estimation of the motion $\mathbf{W}_{\text{init}} = \mathbf{W}_{(1)}$ (yellow/solid arrow, Fig. 2, Frame 2, bottom left image). This process is repeated with each image of the sequence.

The motion model \mathbf{W} is a 3×3 matrix (1) parameterized by the vector of parameters $\mathbf{p} = (p_1, \dots, p_n)^T$ in such a way that \mathbf{W} is the identity matrix when the parameters are equal to zero.

$$\mathbf{x}' = \mathbf{W} \mathbf{x} = \mathbf{W}(\mathbf{x}; \mathbf{p})$$

$$\mathbf{W} = \begin{bmatrix} 1 + p_1 & p_2 & p_3 \\ p_4 & 1 + p_5 & p_6 \\ p_7 & p_8 & 1 \end{bmatrix} \quad (1)$$

\mathbf{W} models different 2D transformations, where each transformation is a specialization of a more complex one [17]:

- **Translation:** has two degrees of freedom (DOF): p_3 and p_6 . They represent the position in the x and y axes in the image plane, and p_1, p_2, p_4, p_5, p_7 and p_8 are equal to zero.
- **Rotation+Translation:** is also known as 2D rigid body transformation and has three DOF: the 2D position in the image plane (p_3, p_6), and rotation ($p_2 = -p_4$, assuming small rotations). p_1, p_5, p_7 and p_8 are equal to zero.
- **Rotation+Translation+Scale:** also known as similarity transformation. This transformation has four DOF: the 2D position in the image plane (p_3, p_6), rotation ($p_2 = -p_4$, assuming small rotations), and scale ($p_1 = p_5$). p_7 and p_8 are equal to zero.
- **Affine:** this model has six DOF: $p_1 \dots p_6$ (p_7 and p_8 are equal to zero).
- **Homography:** This transformation has eight DOF (defined up to scale value) and will be parameterized by eight parameters as shown in (1), although other parameterizations can be used, as presented in [2, 5].

Therefore, \mathbf{W} is the motion model that will transform the 2D pixel coordinates \mathbf{x} (where $\mathbf{x} = (x, y, 1)^T$) of the reference image \mathbf{T} , into the 2D coordinates $\mathbf{x}' = (k x', k y', k)^T$ in the current image \mathbf{I} . If \mathbf{W} represents the homography, then $k = x p_7 + y p_8 + 1$. Otherwise $k = 1$.

For the application of tracking on-board UAVs the assumption of 2D motion models is sufficient: considering that the tracking algorithm will be used for tracking planar surfaces (for applications such as building inspection, helipad for landing) or non-planar surfaces that can be assumed planar when flying at high altitudes, as shown in [30].

The goal of the image registration technique consists therefore in finding the vector of parameters \mathbf{p} . The algorithm used to find these parameters is the ICIA, proposed in [3]. It is classified as *Inverse Compositional* in [4]. This algorithm minimizes the Sum of Squared Differences (SSD), as shown in (2), using a gradient descent approach.

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

The increment of the parameters $\Delta \mathbf{p}$ (see $\Delta \mathbf{W}$ in Fig. 2) is found after a first-order Taylor series expansion of (2):

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

where \mathbf{H} is the approximation of the Hessian matrix, defined as:

$$\mathbf{H} = \sum_{\mathbf{x}} \left[\nabla \mathbf{T} \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right]^T \left[\nabla \mathbf{T} \frac{\partial \mathbf{W}}{\partial \mathbf{p}} \right] \quad (4)$$

where $\nabla \mathbf{T} = (\frac{\partial \mathbf{T}}{\partial x}, \frac{\partial \mathbf{T}}{\partial y})$, is the gradient of the template that is evaluated at $\mathbf{W}(\mathbf{x}, \mathbf{0})$ (i.e. when the template is defined); and $\frac{\partial \mathbf{W}}{\partial \mathbf{p}}$ is the Jacobian of the transformation also evaluated when the template image is defined (i.e. at $(\mathbf{x}, \mathbf{0})$).

Then, the motion model is updated as follows:

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

This algorithm iteratively estimates the increment in the parameters of the motion model ($\Delta \mathbf{p}$), until stopping criteria are reached, denoting the best local alignment solution. In our implementation, we have defined three criteria: the minimum is reached if the increment of the parameters is below the following threshold $||\Delta \mathbf{p}|| \leq 10^{-5}$, if the MAE (mean absolute error) between \mathbf{T} and \mathbf{I} does not decrease after a defined number of iterations (10 iterations), or if the maximum number of iterations has been reached (100 iterations).

The key points of the ICIA algorithm, when compared with the first proposed gradient descent-based registration:

the Lucas and Kanade (LK) approach [28], are the changes of roles of images \mathbf{I} and \mathbf{T} in (2), and the way the motion model is updated in (5). This change of roles makes the Hessian matrix to be constant, calculated at the beginning of the tracking task, thus making the ICIA algorithm a more efficient approach than the LK one. This is the main reason why this image registration approach is used in this paper.

4 The hierarchical multi-parametric and multi-resolution strategy (HMPMR)

As mentioned in the previous section, an incremental image registration algorithm can be used to track objects. If the iterative ICIA algorithm is used, an efficient tracking can be achieved using direct methods.

This iterative algorithm (the ICIA) relies on the assumption that a previous estimation of the parameters of the motion model is known and that after a linearization of the cost function (2) the algorithm iteratively estimates the increments of the parameters of the motion model until stopping criteria are reached. Nevertheless, this linearization is valid only when the range of motion between frames is small. Concerning tracking with cameras on-board aerial vehicles, as well as regarding other applications, this constraint cannot always be ensured (because of fast 3D motions, limited capacity on-board, etc.).

In the literature, multi-resolution (MR) approaches were proposed to help dealing with the small frame-to-frame motion constraint of direct methods [5]. Images are down-sampled creating an MR pyramid, and in each resolution level the same motion model is estimated. The MR strategy suggests that at low resolutions, the vector of motion is smaller and long displacements can be better approximated, as was pointed out in [5]. The higher the frame-to-frame motion is, the bigger the number of levels the MR strategy requires to be able to cope with the large displacement. Nevertheless, there must be a compromise between the number of levels required to overcome the large inter-frame motion and the amount of information required to estimate the motion, because in practice, as pointed out in [41], “it is hard to use more than two or three levels of a pyramid before important details start to be blurred away”.

In Fig. 3, it can be seen that the maximum motion an MR strategy can estimate (i.e. the maximum number of levels the MR structure should have) is closely related with the image information in each level. In the example shown in Fig. 3, an MR pyramid with five levels is used to estimate the homography motion model (8 parameters).

An intuitive analysis of Fig. 3 suggests that the information at low resolutions, e.g. at levels 4 and 5, may not be good enough in terms of quality and quantity for estimating the eight parameters of the homography in those levels (in an

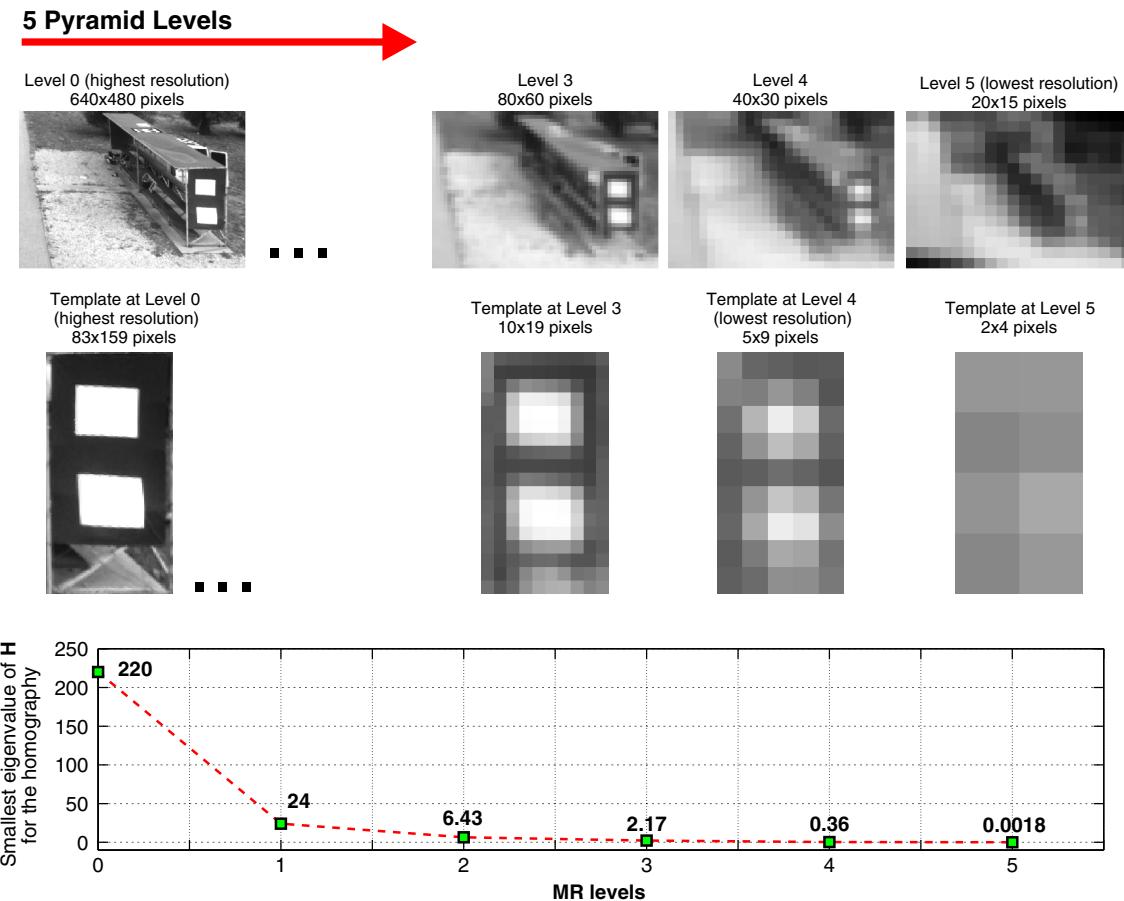


Fig. 3 General idea of an MR strategy. Images are downsampled and in each resolution level the same motion model is estimated. At low resolutions (e.g. levels 4 and 5), the information could be not good enough (important details blurred away) to estimate a motion model with large numbers of parameters (e.g. a homography, 8 parameters). In the plot, it can be seen that there is a relation between the image information and

the number of parameters that can be estimated with that information, which can be analyzed with the smallest eigenvalue criterion. As can be seen in levels 4 and 5 \mathbf{H} is close to be singular (its smallest eigenvalue is close to 0), when the eight parameters of the homography are estimated in those levels

MR strategy, the same motion model is estimated in each level of the pyramid). As was said in [42]: sometimes “the homography contains more free parameters than necessary, in those cases, the algorithm suffers from slow convergence and sometimes gets stuck in local minima”. This is why in this paper we propose to estimate simple motion models at low resolutions.

On the other hand in [39], a threshold in the smallest eigenvalue as a measurement of feature quality was proposed. In this paper, we propose to use the smallest eigenvalue of the Hessian (\mathbf{H}) as a confidence measurement of the tracking algorithm for estimating specific numbers of parameters with the information available in a given resolution. In the plot shown in the lower part of Fig. 3, it can be seen that if the MR-ICIA algorithm is used, the smallest eigenvalues of \mathbf{H} in the fourth and the fifth levels of the pyramid are close to zero, i.e. \mathbf{H} is ill-conditioned in those levels. The quality of the visual information available in those levels is not

going to be sufficient to correctly estimate the eight parameters of the homography (the tracking algorithm requires a well-conditioned \mathbf{H}).

Therefore, from Fig. 3 it can be inferred that when only an MR strategy is used, the low-resolution information could be not sufficient to find a robust estimation of a motion model, especially if it has large numbers of parameters (e.g. a homography). Nonetheless, if less pyramid levels are considered to avoid the loss of information produced by the low resolution, the reduction of levels in the pyramid will cause a reduction in the range of motion that the algorithm can tolerate. For this reason, for many applications, MR approaches are sometimes not good enough to solve the tracking problem in the presence of large frame-to-frame motions, presenting an unstable behavior when estimating complex motion models with large numbers of parameters, especially when they are estimated in the low resolution levels.

The aforementioned problem motivated the idea of including the multi-parametric (MP) estimation of the motion model inside the MR pyramid. We have called this strategy the Multi-Parametric and Multi-Resolution strategy: the HMPMR. The HMPMR approach allows to continue taking advantage of the low-resolution information to find large ranges of motion, even when motion models with large numbers of parameters are required to be estimated in the highest level of resolution.

4.1 The HMPMR strategy for tracking

The main objective of the HMPMR structure is to cope with the deficiencies of the MR structure. It is focused on robustly estimating complex motion models under large frame-to-frame motions.

The HMPMR strategy makes use of two hierarchical structures: the multi-resolution (MR) structure, created by down-sampling the images (*Gaussian* [1] or *Laplacian* [8]); and the multi-parametric (MP) one, where for each MR level, a specific motion model is recovered (different motion models are estimated in each level).

The main idea of using simultaneously the MP and MR structure is to use the low resolution levels to estimate simple motion models (with a few parameters) and to increase the complexity of the motion model with the resolution of the image (i.e. the number of estimated parameters increases). By doing this, we ensure to have a well-conditioned Hessian in the lowest resolution levels. The values of the smallest eigenvalue of \mathbf{H} increase, in comparison with the ones obtained when a motion model with large numbers of parameters is estimated in each level (i.e. when the MR approach is used).

The previous considerations can be confirmed using the example shown in Fig. 3. For this example, Fig. 4 shows the smallest eigenvalues of \mathbf{H} when different configuration of parameters is used in different resolution levels. In each bar of the figure the upper number represents the number of parameters estimated in that resolution, and the lower number corresponds to the value of the smallest eigenvalue of \mathbf{H} . In the figure, it can be seen that when the eight parameters of the homography (blue/dark solid bars) are estimated

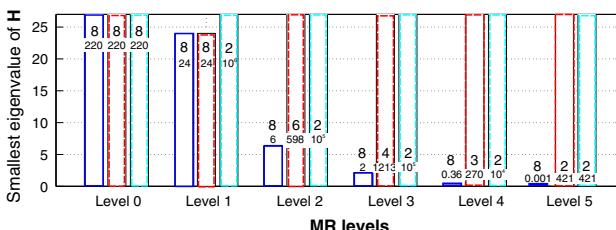


Fig. 4 Comparison of the smallest eigenvalues of \mathbf{H} (upper number) found for the images of Fig. 3, when different parameters are estimated in each level (lower number)

in all the levels (using an MR-ICIA), then only the visual information of levels 0–3 should be used to estimate the homography motion model, because for levels 4 and 5 \mathbf{H} will be ill-conditioned (smallest eigenvalues are close to 0). Conversely, it can be seen that a well-conditioned Hessian is obtained when a few parameters (e.g. 2 parameters) are estimated in the low resolution levels (e.g. levels 4 and 5) using an HMPMR-ICIA strategy.

Therefore, as shown with the example of Fig. 3, using the HMPMR strategy the range of motion that the tracking algorithm can handle is increased compared with the one handled when only an MR approach is used. This is due to the fact that with the HMPMR structure more multi-resolution levels can be used. In the example, the HMPMR structure could use the five levels to estimate the homography at the highest level of resolution.

The advantages of the HMPMR strategy are discussed and tested later in Sect. 5.1. In that section, an analysis of how the configuration of the MP structure affects the performance of the tracking algorithm is presented.

– Definition of pyramid levels (pL)

The different multi-resolution levels (pL) are defined as a function of the size of the ROI that defines the template image \mathbf{T} (i.e. the one that contains the object to track), in such a way that in the lowest resolution level (i.e. the j_{\max} level), an image with no less than a defined number of pixels ($minPixels$) will be used. Therefore, pL is defined as follows, taking into account that the images are downsampled by a factor of 2:

$$2^{pL} = \frac{lowS}{minPixels} \quad (6)$$

where $lowS$ represents the lowest size between the width and height values of image \mathbf{T} (the ROI that contains the object of interest). On the other hand, $minPixels$ is defined as the minimum size the template must have in the lowest resolution image, to have enough information to find the motion model of that level (e.g. 5 pixels could be enough to estimate the translation). Thus, with (6), the different levels of the MR structure can be defined automatically and could be modified on-line according to the size of the template in the current image.

– Definition of motion models in the MP structure

The MP structure suggests an increase in the complexity of the motion model as the resolution of the image increases. It is defined according to the motion model selected at the lowest level of the pyramid (the highest resolution level) \mathbf{W}^0 (where the superscript represents the level of the pyramid).

In this level, \mathbf{W}^0 must be chosen as the best transformation that represents the motion of the object in the image plane. On the other hand, to ensure the detection of large frame-to-frame motions, the translation motion model is chosen for the highest level of the pyramid (the level that has the lowest resolution image) $\mathbf{W}^{j_{\max}}$ (where $j_{\max} = pL - 1$). This ensures a well-conditioned Hessian in the lowest resolution level. Finally, the criterion to define the intermediate levels is to select them in such a way that a smooth transition of the parameters from the highest to the lowest level of the pyramid is obtained (all these criteria are later discussed in Sect. 5.1).

For example, if a camera is moving in the 3D space with different rotations and translations, then the homography can be chosen in \mathbf{W}^0 to represent the motion of an object in the image plane; then assuming a hierarchical structure of four levels, the following combination of parameters could be used in the lower resolution levels:

$$\begin{aligned}\mathbf{W}^{j_{\max}} &= \mathbf{W}^3 = 2 \text{ parameters} \rightarrow \text{Lowest resolution} \\ \mathbf{W}^{(j_{\max}-1)} &= \mathbf{W}^2 = 3 \text{ parameters} \\ \mathbf{W}^{(j_{\max}-2)} &= \mathbf{W}^1 = 4 \text{ parameters} \\ \mathbf{W}^{(j_{\max}-3)} &= \mathbf{W}^0 = 8 \text{ parameters} \rightarrow \text{Highest resolution.}\end{aligned}$$

The MP structure will be represented by the number of parameters that define the motion model that will be estimated in each multi-resolution level. Therefore, in the previous example, the MP configuration is in the form: 8-4-3-2. The first number corresponds to the motion model \mathbf{W}^0 that will be estimated in the lowest pyramidal level (highest resolution image), e.g. the homography. The last number corresponds to the motion model $\mathbf{W}^{j_{\max}}$ that will be estimated in the highest pyramidal level (lowest resolution image), e.g. the translation; and the other numbers correspond to the motion models estimated in the intermediate levels (the similarity and the translation+rotation motion models).

– Propagation of parameters

An important part of the HMPMR structure is the propagation of parameters inside the MR structure and among frames. The parameters that are estimated in each level by the image registration algorithm are used as an initial estimation of the motion for the following levels, taking into account that the images have been scaled by a factor of two, as follows:

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

Where the subscript i represents the parameter shown in (1) and j represents the level of the pyramid, and $j = \{j_{\max}, j_{\max} - 1, \dots, 0\} = \{pL - 1, pL - 2, \dots, 0\}$.

On the other hand, in level 0, another kind of propagation occurs from the lowest level of the pyramid (level 0) of the previous image $\mathbf{I}_{(F-1)}$, to the highest level of the pyramid ($j = j_{\max}$) of the new image $\mathbf{I}_{(F)}$, where F represents the number of the frame. This propagation is the basis of the tracking-by-registration strategy, explained in Sect. 3.

The propagation is conducted as follows:

$$\begin{aligned}p_{i(F)}^{j_{\max}} &= p_{i(F-1)}^0 \quad \text{for } i = \{1, 2, 4, 5\} \\ p_{i(F)}^{j_{\max}} &= \frac{p_{i(F-1)}^0}{s} \quad \text{for } i = \{3, 6\} \\ p_{i(F)}^{j_{\max}} &= s p_{i(F-1)}^0 \quad \text{for } i = \{7, 8\}\end{aligned}\tag{8}$$

where $s = 2^{j_{\max}}$. This propagation of the parameters allows to validate the linearization of (2) done by the image registration algorithm. Therefore, when a new image is analyzed, using the motion model $\mathbf{W}(\mathbf{x}; \mathbf{p})$ estimated in the previous frame, the images \mathbf{T} and \mathbf{I} are close enough to find a minimum in the alignment of the images.

4.2 The HMPMR-ICIA algorithm

Figure 5 shows the diagram of the HMPMR structure for tracking using the ICIA algorithm, and Algorithm 1 describes in more detail the different steps of the algorithm.

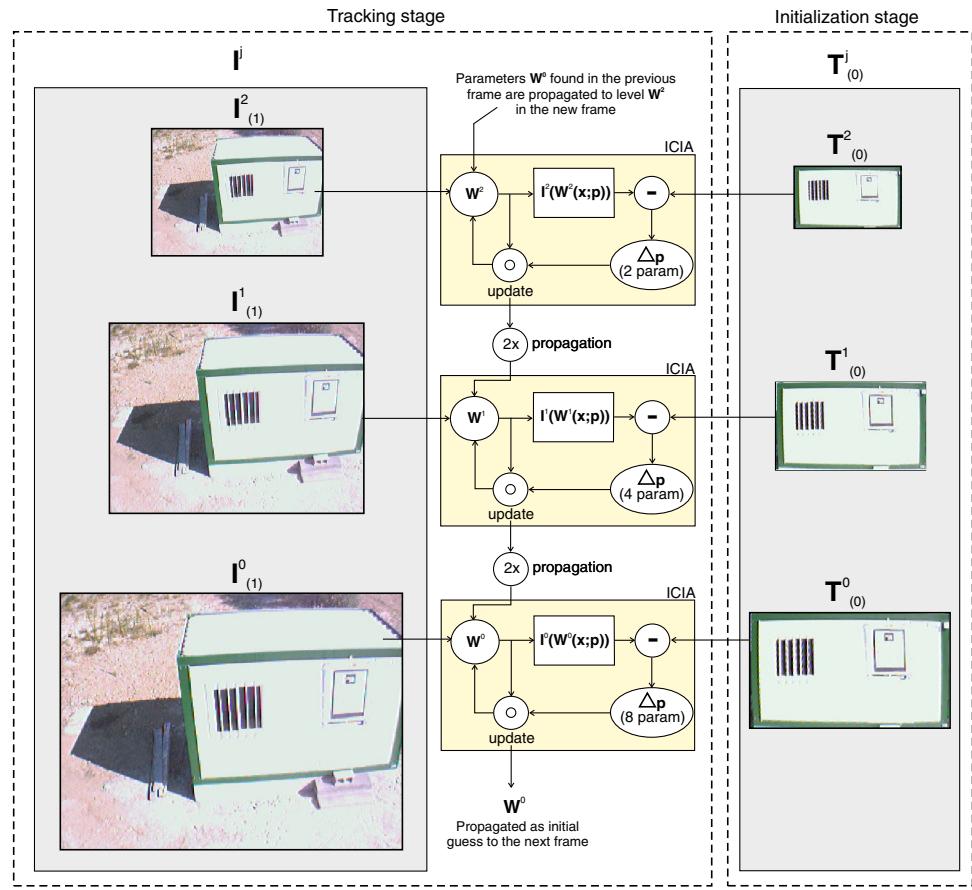
As input, the algorithm requires the information of $\mathbf{I}_{(0)}$ (where the subscript represents the number of frames), and the \mathbf{x} coordinates in $\mathbf{I}_{(0)}$ (i.e. in the initial frame) of the object to track. These coordinates can be found manually or automatically, e.g. using template matching approaches [31]. Additionally, the algorithm requires the definition of the levels of the MR structure (pL), which are defined using (6); and the definition of the different motion models in the MP structure \mathbf{W}^j .

In general terms, the algorithm is based on two stages: the initialization and the tracking stages.

In the initialization stage (steps 1–6, Algorithm 1), the initial frame $\mathbf{I}_{(0)}$ is downsampled according to pL . Thus creates the template image $\mathbf{T}_{(0)}^j$ for each level (where j represents the level of the MR pyramid), as shown in Fig. 5. Additionally, in this initialization stage, for each level of the pyramid the Hessian matrix $\mathbf{H}_{(0)}^j$ and its inverse are calculated. These steps, 1–6 of Algorithm 1, are carried out at the beginning of the tracking task and every time the template image is updated.

One of the advantages of the ICIA is that \mathbf{H} is calculated in the initialization stage and not in every frame. Hence, before starting, it is possible to know if \mathbf{H} is well-conditioned for the kind of image to track and for the parameters chosen

Fig. 5 Example of the HMPMR-ICIA framework. Images $\mathbf{I}_{(F)}$ and $\mathbf{T}_{(0)}$ are downsampled to create the MR structure. In each level, the ICIA is applied to estimate the parameters defined for that level (the MP structure). When the stopping conditions have been reached, the parameters are propagated to the next level. At level $j = 0$, the most complex defined motion model is estimated ($\mathbf{W}_{(F)}^0$), and with it, the object to track $\mathbf{T}_{(0)}$ in image $\mathbf{I}_{(F)}$ is found. Finally, $\mathbf{W}_{(F)}^0$ is propagated to the highest level of the pyramid of the next frame, as initial guess of the motion



for each level. This advantage could be especially useful in an automatic update of the template, so that both the MP and the MR structures of the tracker can change dynamically during the tracking task based on the new information of \mathbf{H} . However, the discussion of this dynamic update strategy of the template is out of the scope of this paper.

During the tracking stage, when a new image is analyzed $\mathbf{I}_{(F)}$ (e.g. $\mathbf{I}_{(1)}$), it is first downsampled to create the MR structure, as shown in Fig. 5 (tracking stage, left-dashed box). The motion model at the highest level of the pyramid, e.g. $\mathbf{W}_{(1)}^{j_{\max}}$ (lowest resolution) is initialized using (8), as shown in step 8 of Algorithm 1. Because this is the first frame analyzed in the tracking stage, $\mathbf{W}_{(1)}^{j_{\max}}$ is the identity matrix.

Then, for each level of the pyramid, as illustrated in Fig. 5, the HMPMR-ICIA algorithm is applied. The process is repeated in the different levels of the pyramid. The complexity of the motion model increases as the resolution of the image increases. Therefore, at the lowest level of the pyramid (i.e. the one that has the image with the highest resolution), the most complex motion model is estimated. With this information, it is possible to determine the position of $\mathbf{T}_{(0)}$ (i.e. the object to track) in the current image, e.g. $\mathbf{I}_{(1)}$ (steps 15–16 Algorithm 1).

Finally, the motion model found in this frame (e.g. $\mathbf{W}_{(1)}$) is propagated as initial guess to the highest level of the pyramid

\mathbf{W}^0 of the next frame (e.g. $\mathbf{I}_{(2)}$), using (8). This propagation of the parameters permits to validate the linearization of (2) and permits that when a new frame is analyzed using the estimation of $\mathbf{W}_{(F-1)}$ in the previous frame as an initial estimation of the motion in the current frame, images $\mathbf{T}_{(0)}$ and $\mathbf{I}_{(F)}$ are close enough to each other to find a minimum.

5 Evaluation of the HMPMR strategy

Different tests have been conducted to evaluate the performance of the HMPMR-ICIA algorithm. In these tests the advantages of extending the ICIA algorithm with a HMPMR framework are shown, especially under the presence of large frame-to-frame motions. Qualitative and quantitative results are presented.

In the tests different algorithms are used to compare the performance of the HMPMR strategy. Some of those algorithms are based on direct methods (test 1 and 2), and others are based on features (test 2). The different configurations of the ICIA algorithm (HMPMR-ICIA, MR-ICIA, ICIA) have been implemented in C++, and the OpenCV libraries [7] have been used for managing image data. On the other hand, the KLT [6] feature-based tracking algorithm that is used is based on the implementation of the tracker included in the OpenCV

```

input :  $\mathbf{I}_{(0)}$ ,  $\mathbf{W}^j$ ,  $pL$ ,  $\mathbf{I}_{(0)}$ ,  $\mathbf{x}$ 
output: Transformation that tells where  $\mathbf{T}_{(0)}$  is located in  $\mathbf{I}_{(F)}$ 
Initialization stage
1. Downsample  $\mathbf{I}_{(0)}$  according to  $pL$  and create  $\mathbf{T}_{(0)}$ 
2. Initialize  $j_{\max} = pL - 1$ 
for  $j \leftarrow j_{\max}$  to 0 do
    3. Evaluate the gradient  $\nabla \mathbf{T}_{(0)}^j = \left( \frac{\partial \mathbf{T}_{(0)}^j}{\partial x}, \frac{\partial \mathbf{T}_{(0)}^j}{\partial y}, \mathbf{1} \right)$ 
    4. Evaluate the Jacobian  $\mathbf{J}^j = \frac{\partial \mathbf{W}^j}{\partial \mathbf{p}}$  at  $(\mathbf{x}, \mathbf{0})$ 
    5. Compute the steepest descent images
         $\mathbf{SDI}^j(\mathbf{x}) = \nabla \mathbf{T}(\mathbf{x})_{(0)}^j \frac{\partial \mathbf{W}^j}{\partial \mathbf{p}}$ 
    6. Compute the Hessian matrix and its inverse
         $\mathbf{H}^j = \sum_{\mathbf{x}} [\mathbf{SDI}(\mathbf{x})^{j^T} \mathbf{SDI}(\mathbf{x})^j],$  and  $\mathbf{H}^{j-1}$ 
end
Tracking stage
foreach new image  $\mathbf{I}_{(F)}$  do
    7. Downsample  $\mathbf{I}_{(F)}$  according to  $pL$  to create  $\mathbf{I}_{(F)}^j$ 
    8. Initialize  $\mathbf{W}_{(F)}^{j_{\max}}$  according to Equation (8). If it is the
        first image,  $\mathbf{W}_{(F)}^{j_{\max}}$  is the identity matrix
    for  $j \leftarrow j_{\max}$  to 0 do
        repeat
            foreach  $\mathbf{x}$  in  $\mathbf{T}_{(0)}$  do
                9. Warp  $\mathbf{W}_{(F)}^j(\mathbf{x}; \mathbf{p})$  to find
                     $I^j(\mathbf{W}_{(F)}^j(\mathbf{x}; \mathbf{p}))$ 
                10. Compute
                     $E^j = [I_{(F)}^j(\mathbf{W}_{(F)}^j(\mathbf{x}; \mathbf{p})) - T_{(F)}^j(\mathbf{x})]$ 
                11. Compute  $\mathbf{b}^j = \mathbf{b}^j + \mathbf{SDI}(\mathbf{x})^{j^T} E^j$ 
            end
            12. Compute  $\Delta \mathbf{p}^j = \mathbf{H}^{j-1} \mathbf{b}^j$ 
            13. Update the warp
                 $\mathbf{W}_{(F)}^j(\mathbf{x}; \mathbf{p}) \leftarrow \mathbf{W}_{(F)}^j(\mathbf{x}; \mathbf{p}) \circ \mathbf{W}_{(F)}^j(\mathbf{x}; \Delta \mathbf{p}^j)^{-1}$ 
        until  $\|\Delta \mathbf{p}^j\| \leq \varepsilon;$ 
        14. Propagate the parameters in  $\mathbf{W}^j(\mathbf{x}; \mathbf{p})$  to the next level
            using Equation (7)
    end
    15. Use  $\mathbf{W}_{(F)}^0$  to find the position of  $\mathbf{T}_{(0)}$  in  $\mathbf{I}_{(F)}$ 
    16. Draw results
end

```

Algorithm 1: HMPMR-ICIA tracking algorithm

libraries [7]. Additional functions were included to allow the estimation of different motion models. The maximum number of features was defined as 200, a window size of 5 was used, and different pyramid levels were used in the multi-resolution structure of the algorithm. On the other hand, the SIFT algorithm is based on the implementation developed by Rob Hess [18, 19]. The values of the different parameters the algorithm requires correspond to the standard values that come with the implementation of the algorithm.

5.1 Test 1: analysis of the MP hierarchy

In this test the selection of the MP structure of the HMPMR-ICIA algorithm and its influence in the tracking results are

analyzed. Additionally, the performance of the HMPMR strategy during a tracking task recovering simple and complex motion models is evaluated. Different configurations of the ICIA algorithm are compared: the proposed HMPMR-ICIA, the ICIA algorithm without hierarchies, and the MR-ICIA (all of them are based on direct methods).

In the image sequence used in the test, the object to track is a flat symbol located on the ground. The size of the images is 640×480 pixels; and the size of the template is 115×125 pixels, so that according to (6) $pL = 4$, considering $\minPixels = 5$. It is important to mention that the selected image sequence contains large frame-to-frame motions (5, 10, and sometimes >20 pixels), as can be seen in Fig. 6b, where the frame-to-frame change in position of the generated ground truth data is plotted. This characteristic makes this sequence challenging from the visual-tracking point of view, especially if it is considered that the linearization of (2) is a good approximation when the frame-to-frame motion is <1 pixel [22].

In this test, the algorithms are evaluated based on manually generated ground truth data (GT) in all the frames of the sequence, as shown in Fig. 6a (the red cross in the upper-left corner).

The GT data in each image ($\mathbf{x}_{gt(F)}$) is transformed into the coordinate system of the first frame (i.e. it is back-warped), using the transformation found by the tracking algorithm $\mathbf{x}_{bw(F)} = \mathbf{W}_{(F)}^{-1} \mathbf{x}_{gt(F)}$. The accuracy of the motion model found with the different tracking algorithms is evaluated by measuring the Mean Absolute Error (MAE) of the upper-left corner, as follows:

$$\text{MAE} = \frac{1}{n} \sum_{F=1}^n \text{ME}_{(F)}$$

$$\text{ME}_{(F)} = \frac{|x_{bw(F)} - x_{gt(0)}| + |y_{bw(F)} - y_{gt(0)}|}{2} \quad (9)$$

where ME is the mean error, n is the total number of frames of the image sequence, $x_{bw(F)}$ and $y_{bw(F)}$ are the back-warped coordinates of the GT point of each frame (F), and $x_{gt(0)}$ and $y_{gt(0)}$ are the coordinates of the GT point in the first image (in the frame where the tracking algorithm was initialized).

Based on the analysis of the ME of the corner's coordinates, the percentage of frames tracked by the different algorithms is examined (TF). If $\text{ME}(F) > 2$ pixels, the tracker is considered lost in that frame. This threshold was defined assuming that the error by “clicking” the GT data is ± 2 pixels.

In Figs. 7, 8, and 9, a collection of images that show the performance of the tested algorithms is presented. Figure 7 presents the results of the tracking task using the ICIA algorithm without any hierarchy, recovering different motion models with different numbers of parameters: eight parameters (the homography, first row in the figure), six parameters (the affine transformation, second row), and four

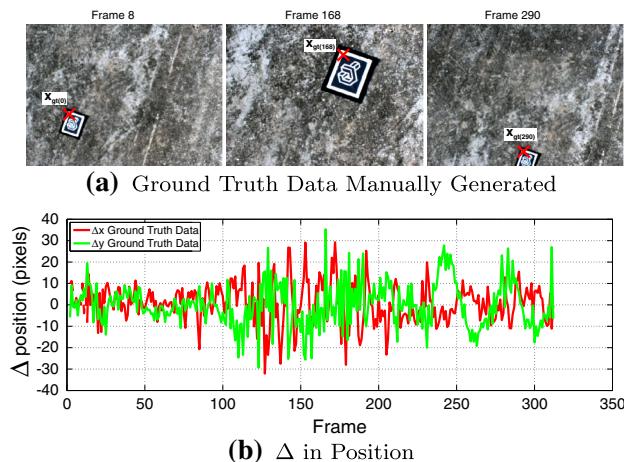


Fig. 6 Ground Truth Data (GT). The *upper-left* corner of the template is selected manually in all the frames of the sequence **(a)**. In **(b)**, the frame-to-frame changes in position of the generated GT data are plotted. This sequence contains large frame-to-frame motion, which sometimes is higher than 20 pixels, as can be seen in **(b)**

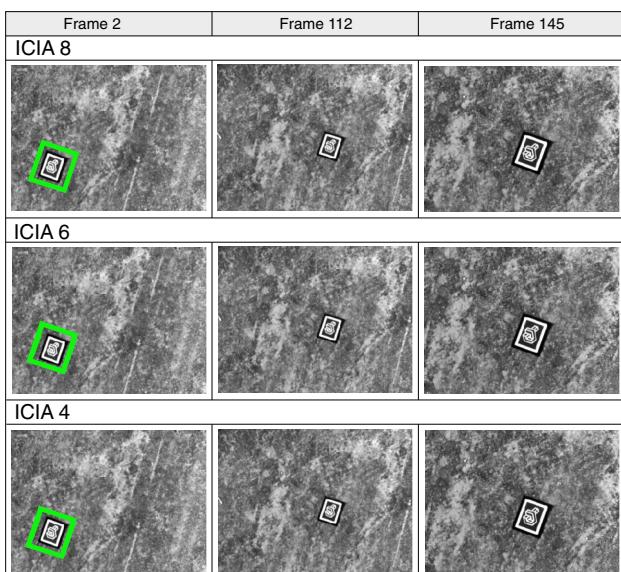


Fig. 7 ICIA results. The *green box* indicates the result of the tracking task when recovering motion models with eight, six, and four parameters. Without using any hierarchy, none of the tested configurations of the ICIA was able to track the template because of the large motion of the sequence (>5 pixels) (color figure online)

parameters (the similarity transformation, third row). The green/light box indicates the result of the tracking task. As can be seen in Fig. 7, the large frame-to-frame motion of this sequence violates one of the main constraints of direct methods (small motion), and so the ICIA (without hierarchies) was not able to track the template in more than three frames in any of the tested configurations.

Figure 8 presents the results of the MR-ICIA. In this configuration, the same number of parameters is found in the four levels of the hierarchical structure. The first row

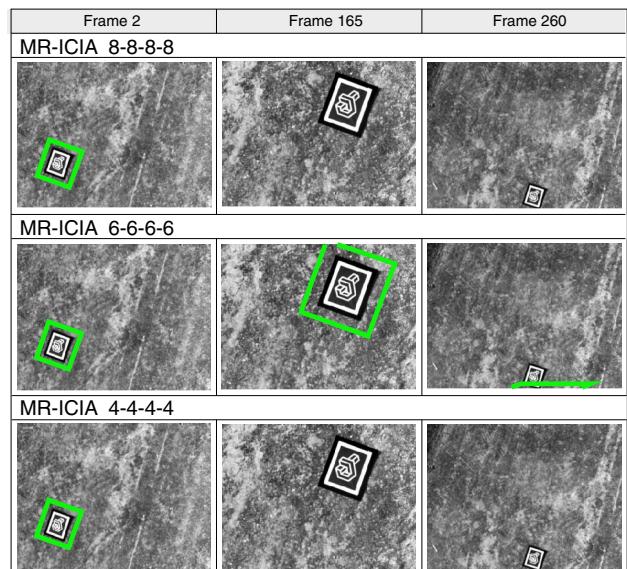


Fig. 8 MR-ICIA results. The *green/light box* indicates the result of the tracking task when recovering eight, six, and four parameters. As can be seen, none of the tested configurations of the MR-ICIA was able to track the template in all the sequence (color figure online)

presents the results recovering the homography in the four levels of the pyramid (8 parameters), and the second and third rows present the results recovering the affine (6 parameters) and the similarity transformation (4 parameters), respectively. None of the tested configurations tracked the template throughout the sequence.

In Fig. 9 the results of the tracking task using the HMPMR-ICIA algorithm are presented. Different configurations of parameters were tested with this algorithm to analyze the behavior of the MP structure and to define a criterion to select the motion models, especially under the presence of large frame-to-frame motions. The configurations that were tested allow to obtain different motion models in the highest resolution level, such as the homography, the affine, and the similarity transformations.

The first row of Fig. 9 shows the different configurations of parameters that were able to track the template throughout the sequence. The images correspond to the configuration shown in bold letters (HMPMR-ICIA 8-6-4-2), although the other configurations also obtained similar results when tracking the template (i.e. the template was tracked in all the frames of the sequence). The second row shows the configurations that were not able to track the template in the sequence.

On the other hand, the analyses of the smallest eigenvalue criteria for the selected template in this image sequence show that although the Hessian \mathbf{H}^j was well-conditioned in all the levels of all the tested configurations (smallest eigenvalues >1) for estimating large and small numbers of parameters, other factors such as the large frame-to-frame motion of this sequence constrained the performance of the tracking algo-

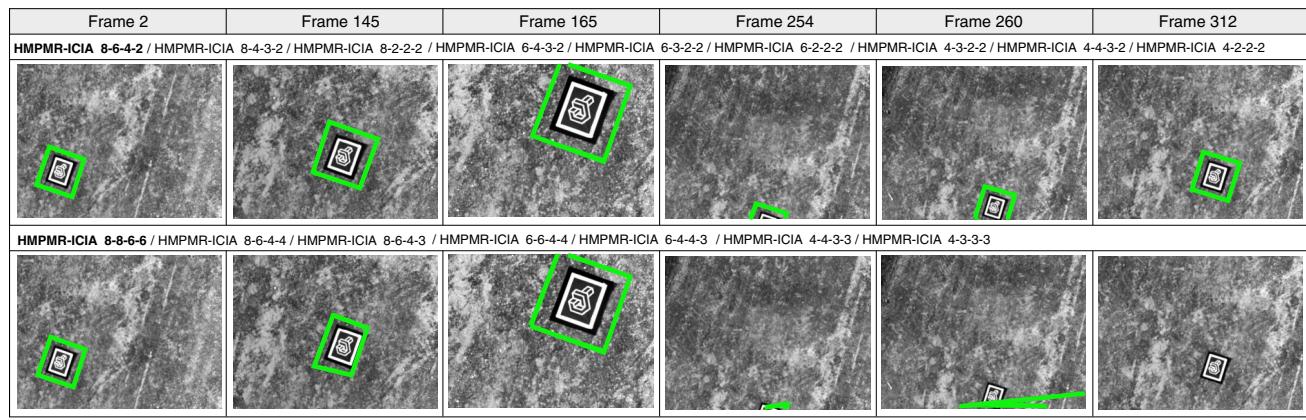


Fig. 9 HMPMR-ICIA results. The green/light box indicates the results of the tracking task when recovering different configuration of parameters in the pyramidal structure. As can be seen, the definition of the

MP structure affects the behavior of the tracking algorithm. Some of the tested configurations are not able to track the template in all the sequence (color figure online)

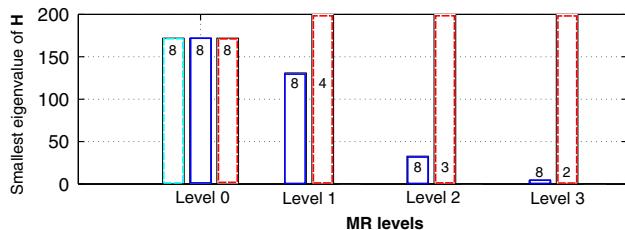


Fig. 10 Comparison of the smallest eigenvalues of \mathbf{H} with different configurations of the ICIA: ICIA 8 without hierarchies (cyan/light dashed), MR-ICIA 8-8-8-8 (blue/dark solid), and HMPMR-ICIA 8-4-3-2 (red/dark solid) (color figure online)

rithms and this is why, for example, the MR-ICIA 4-4-4-4, that estimated few parameters at low resolutions, was not able to track the template either (as was shown in Fig. 7).

Figure 10 compares graphically the values of the smallest eigenvalues of \mathbf{H}^j in some of the configurations of the ICIA algorithm that were tested: the ICIA, cyan/dashed-light bars; the MR-ICIA 8-8-8-8, blue/solid bars; and the HMPMR-ICIA 8-4-3-2, red/dashed-dark bars.

In the figure, it can be seen that although the ICIA (without hierarchies) had a well-conditioned hessian for estimating the eight parameters of the homography (cyan/dashed-light bars), the large frame-to-frame motion made the tracker algorithm fail in the first two frames of the sequence (as was shown in Fig. 7). Additionally, it can be seen that for the selected template, estimating a large number of parameters at low resolutions (e.g. Level 3, 8 parameters blue/solid-dark bars) leads to an \mathbf{H}^j that is not as well conditioned as it is when estimating only a few parameters (e.g. Level 3, 2 parameters, red/dashed-dark bars).

Therefore, using configurations that estimate the translation at the lowest resolution level, the probabilities of obtaining a well-conditioned \mathbf{H}^j increase and also the probabilities of overcoming the large frame-to-frame motion (reducing

the probabilities of being trapped in local minima), e.g. with the HMPMR-ICIA 8-6-4-2 the template was tracked in the image sequence (see Fig. 9, first row).

Figure 11 compares the optimization function of the algorithms in the frame where the ICIA 8 (i.e. without hierarchies) failed (from Frame 2 to 3). It can be seen that because of the large motion of the sequence the ICIA 8 (left plot) did not converge (the frame-to-frame motion was ≈ 5 pixels). Conversely, the hierarchical structures of the MR-ICIA 8-8-8-8 (center plot) and the HMPMR-ICIA 8-4-3-2 (right plot) make it possible to overcome this large motion and find a minimum. Therefore, these algorithms tracked the template in that frame.

Figure 11 also permits to analyze the hierarchical structures of the MR-ICIA (center plot) and the HMPMR-ICIA (right plot). Taking into account that the same maximum number of iterations is defined in all the algorithms, in the plot of the center it can be seen that for the lowest resolution level (Level 3, blue/dark solid-squares line) the MR-ICIA obtained an ME that was smaller than the one obtained with the HMPMR-ICIA (right plot, Level 3, blue/dark solid-squares line). This is because the MR-ICIA estimated eight parameters in that level, modeling the motion of the object in a better way than when only two parameters were estimated with the HMPMR-ICIA. Nonetheless, as can be seen in the plot on the right, as soon as the HMPMR-ICIA increases the complexity of the motion model with the resolution of the image, the obtained MEs are similar to the ones obtained with the MR-ICIA (see right plot, Level 2, black/dark solid line), with the advantage that less parameters have been estimated and therefore reducing the possibilities of being trapped in local minimum.

On the other hand, analyzing the optimization function of the algorithms in the frame where the MR-ICIA 8-8-8-8 failed (Frame 110), in Fig. 12 it can be seen that the MR-ICIA 8-8-8-8 (right plot) was not able to overcome the large frame-

Fig. 11 Comparison of the optimization function from Frames 2 to 3. Because of the large motion of the sequence the ICIA 8 (left plot) fell to a local minimum. Conversely, the hierarchical structures of the MR-ICIA (center plot) and the HMPMR-ICIA (right plot) make it possible to overcome this large motion and find a minimum in that frame

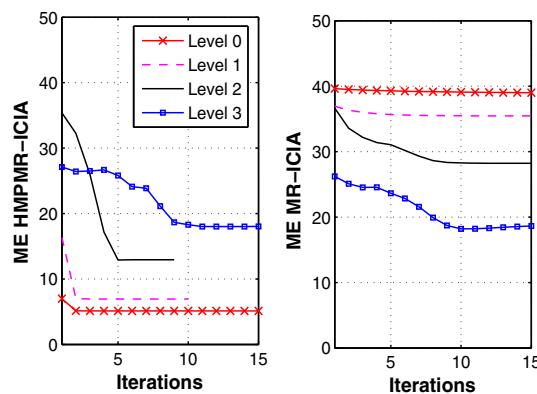
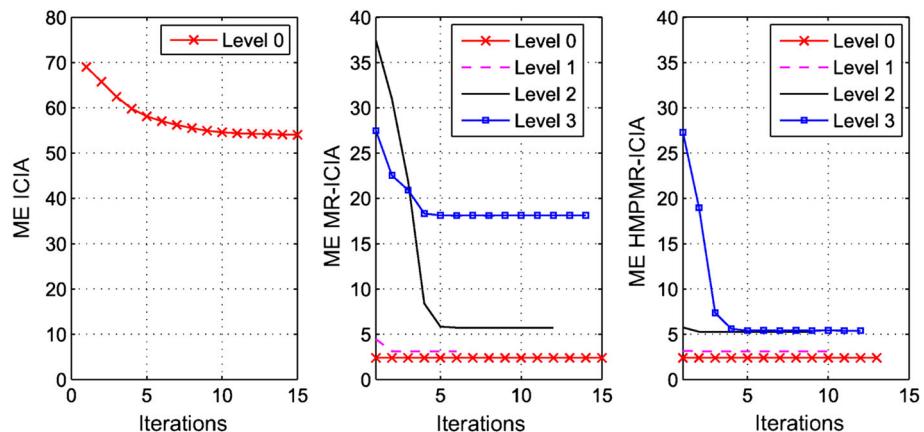


Fig. 12 Comparison of the optimization function from Frames 109 to 110. Because of the large inter-frame motion ≈ 20 pixels, the MR-ICIA (right plot) fell to a local minimum. Conversely, HMPMR-ICIA (left plot) found a minimum in those frames

to-frame motion of the sequence (in Frame 110 the frame-to-frame motion was ≈ 20 pixels). Therefore, the minimum found in the lowest resolution level (Level 3), estimating eight parameters, was similar to the one obtained when only two parameters were estimated with the HMPMR-ICIA 8-4-3-2 (see Level 3 in the plot).

However, when the motion model found in Level 3 is propagated to the next level (i.e. Level 2), the MR-ICIA 8-8-8-8 was not able to improve the estimation of the parameters, falling in a local minimum, as can be seen in the plot on the right side of Fig. 12. On the contrary, as can be seen in the plot on the left side of the figure, the HMPMR-ICIA 8-4-3-2 found a minimum thanks to the propagation of the two parameters from Level 3 to Level 2, that were well-estimated, and also thanks to the fact that in Levels 3 and 2 only a few parameters were estimated. This reduces the possibilities of being trapped in local minima.

Table 1 shows the results of the analysis of accuracy conducted to the three tested algorithms (HMPMR-ICIA, MR-ICIA, and ICIA without hierarchies), using the ground truth (GT) data shown in Fig. 6. The GT data are used to analyze

Table 1 Analysis of the results of accuracy conducted to the HMPMR-ICIA, the MR-ICIA, and the ICIA without hierarchies, using the ground truth (GT) data of Fig. 6

Algorithm	MAE GT data	Tracked frames TF (%)
ICIA 8	1.03×10^5	0.96 (3 frames)
ICIA 6	6.1×10^8	0.32 (1 frame)
ICIA 4	1.9×8^8	0.32 (1 frame)
MR 8-8-8-8	4.11	34.7
MR 6-6-6-6	39.79	89.7
MR 4-4-4-4	7.6×10^4	44.6
HMPMR 8-4-3-2	0.7454	100
HMPMR 8-6-4-2	0.7455	100
HMPMR 8-2-2-2	0.7501	100
HMPMR 8-6-4-3	62.034	80.385
HMPMR 8-8-6-6	35.046	80.385
HMPMR 8-6-4-4	2.08×10^3	44.373
HMPMR 6-2-2-2	0.7576	100
HMPMR 6-4-3-2	0.7577	100
HMPMR 6-3-2-2	0.7572	100
HMPMR 6-6-4-4	96.80	44.69
HMPMR 6-4-4-3	126.62	80.38
HMPMR 4-3-2-2	0.8314	100
HMPMR 4-4-3-2	0.8317	100
HMPMR 4-2-2-2	0.8317	100
HMPMR 4-4-3-3	101.50	44.69
HMPMR 4-3-3-3	61.87	80.38

the recovered transformation (motion model). In Table 1 the mean absolute error MAE of the upper-left corner of the template is shown. Additionally, the percentage of tracked frames (%TF) is presented.

In Table 1, it can be seen that none of the configurations of the ICIA without hierarchies tracked the template in more than three frames. Additionally, these configurations obtained the highest MAE. On the other hand, with the MR-

ICIA algorithm, it can be seen that the MR-ICIA 8-8-8-8 configuration was the one that tracked less frames (34 %), but the one with the smallest error in the estimation of the motion model (more degrees of freedom in the motion model allow to have a better representation of the motion, but the algorithm is more sensible to be trapped in a local minimum). However, with the different evaluation criteria, it can be seen that none of the configurations of the MR-ICIA tracked the template in the sequence. This shows that a multi-resolution hierarchy is not always sufficient to solve the tracking problem when large frame-to-frame motions are presented.

The last section of Table 1 shows the results obtained with the different configurations of the HMPMR-ICIA. The shadowed configurations correspond to the ones that tracked the template throughout the sequence, overcoming the large frame-to-frame motion of the sequence (TF = 100 %). It can be seen that a common characteristic of these configurations is that the translation motion model was estimated in the lowest resolution level.

As was pointed out in [47], the ICIA gives good results when the initial estimates is close to the global minimum. In the first part of the sequence, an MR approach was sufficient to satisfy this constraint (see Fig. 7). However, when the frame-to-frame motion of the sequence increased, only the HMPMR hierarchy was able to deal with the large motion with configurations that estimated the translation (2 parameters) in the lowest resolution level, as can be seen in Fig. 9, first row.

Therefore, the results of this test indicate that by including the MP structure, it was possible to track the template throughout the sequence and to recover complex motion models in the highest resolution level, which was not possible to achieve using only an MR structure.

5.2 Test 2: comparison with feature-based algorithms

In this test, the behavior of the HMPMR-ICIA algorithm under perspective changes is analyzed. This test compares different configurations of the HMPMR-ICIA recovering the homography motion model (the most complex one being considered) in the highest resolution level: 8-6-4-2, 8-4-3-2, and 8-2-2-2. Additionally, the HMPMR-ICIA algorithm is compared with three other algorithms: the MR-ICIA 8-8-8-8, based on direct methods; and two feature-based algorithms, the SIFT [27] and the pyramidal Lucas Kanade [6] (KLT) algorithms.

In the image sequence used in the test, the front of a “house” is used as a template image T (i.e. the object to track). The size of the images is 320×240 pixels, and the size of the template is 213×123 pixels, so that according to (6) four pyramid levels are used. In this sequence, the images contain constant changes in positions because of the UAV vibrations, changes in the appearance of the object to

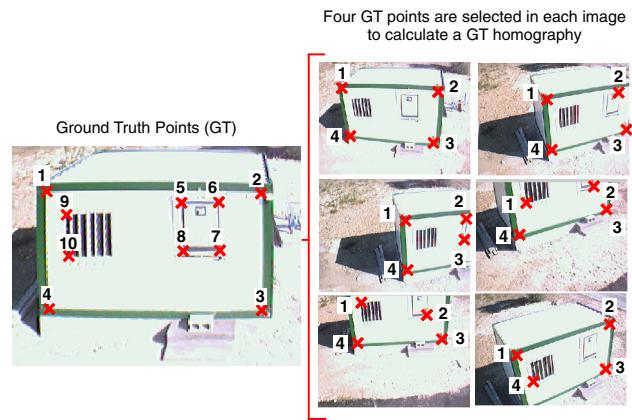


Fig. 13 Ground Truth Data. Four points (right images) of the possible ten points (left image) are selected to calculate a ground truth homography that relates points in the first frame to points in each frame of the sequence

track (due to 3D movements), and loss of information when the object goes out of the field of view (FOV) of the camera.

Ground truth (GT) data are used to evaluate the performance of the algorithms which is based on an analysis of the parameters of the homography estimated by the tracking algorithms. Figure 13 shows the GT data. As can be seen in the image located on the left, ten different manually selected GT points can be used. Nonetheless, taking into account that due to the movements of the UAV the front of the “house” goes out of the FOV of the camera, only four GT points (Fig. 13, right images) well distributed over the template are manually selected in each frame to calculate a ground truth homography ($\mathbf{x}_{(F)}^i = \mathbf{H}_{\text{GT}} \mathbf{x}_{(0)}^i$, for $i = \{1, 2, \dots, 10\}$). This GT homography relates points in the first frame (Frame 0 where the template image was initialized) to points in each frame of the sequence.

In Fig. 14, parameters p_2 and p_8 of the GT homography (green/solid line) are compared with the ones obtained with three different tested configurations of the HMPMR strategy: the HMPMR-ICIA 8-6-4-2 (cyan/light solid line), the HMPMR-ICIA 8-2-2-2 (red/dashed line), and the HMPMR-ICIA 8-4-3-2 (blue/solid line).

In these figures, it can be seen that around Frame 660 and Frame 887, the homographies recovered by the HMPMR-ICIA 8-6-4-2 (cyan/light solid line) and the HMPMR-ICIA 8-2-2-2 (red/dashed line) configurations present a small deviation from the ones estimated by the HMPMR-ICIA 8-4-3-2 (blue/solid line) and the GT data (green/solid line).

When a visual examination of the results was conducted, this small deviation was very difficult to perceive. Nonetheless, the images shown in the right side of Fig. 14 clearly show the small variation of the results. In these four images, the template image that was selected in the first frame (green/light lines) is overlapped with the back-warped template in Frames 660 and 887 (black/dashed lines). The back-

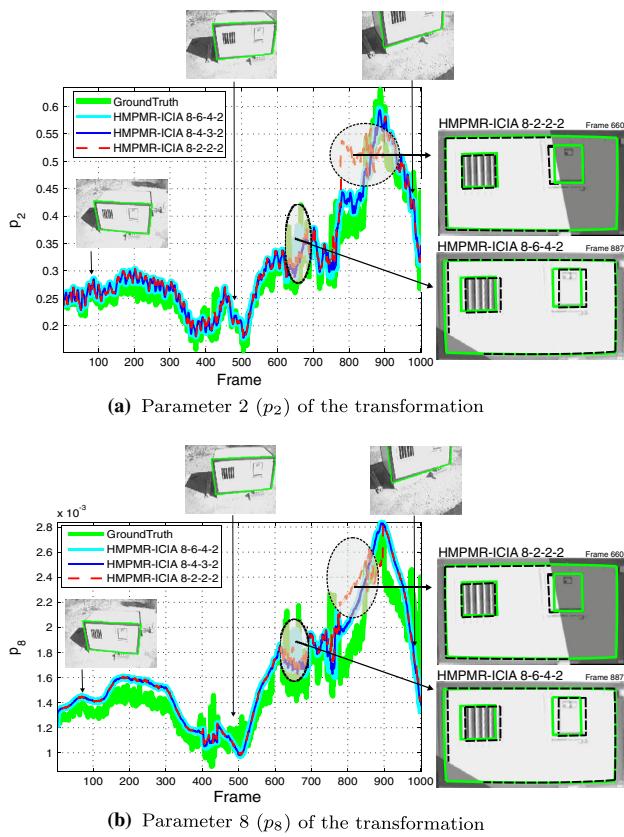


Fig. 14 Comparison of the estimated homography with GT data. Parameters p_2 and p_8 are compared with the ones obtained with a GT homography (green/light-solid line). Small errors, that are not perceived when analyzing the tracking results, are present in the HMPMR-ICIA 8-6-4-2 and HMPMR-ICIA 8-2-2-2 configurations (color figure online)

warped template is found by transforming Frames 660 and 887 into the coordinate system of the first frame using the homography calculated by the different configurations of the HMPMR-ICIA algorithm. If the estimated homography is good, the black/dark dashed lines and the green/light solid lines should coincide. However, analyzing these overlapped images, it can be seen that there is a small deviation in the homography that causes a slight difference between the templates.

This can also be seen analyzing the optimization function in those frames. Figure 15 shows the mean error from Frame 886 to Frame 887, of the highest resolution level of the configurations of the HMPMR structure that were tested. In the plot it can be seen that because of the small number of parameters estimated in the intermediate levels (there was not a smooth transition of the parameters), the HMPMR-ICIA 8-2-2-2 configuration reached a minimum higher than the one obtained with the other two configurations. That is why in this frame the parameters of the HMPMR-ICIA 8-2-2-2 slightly differ from the GT ones. This causes the slight difference

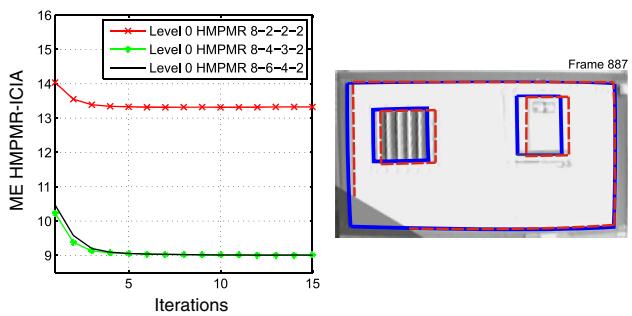


Fig. 15 Comparison optimization function from Frame 886 to Frame 887. Because of the small number of parameters estimated in the intermediate levels (there was not a smooth transition of the parameters), the HMPMR-ICIA 8-2-2-2 configuration reached a minimum higher than the one obtained with the other two configurations

between the template and the back-warped image shown in the right side of Fig. 15.

These results permit to affirm that, although visually all the tested configurations of the HMPMR that estimated the translation motion model at the lowest resolution level were able to track the template, the best results were achieved with the configuration HMPMR-ICIA 8-4-3-2 (blue/dark solid line in Fig. 14), which progressively increases the complexity of the motion model in the different levels using less numbers of parameters, e.g. than the HMPMR-ICIA 8-6-4-2 configuration. Therefore, it converges faster (as can be seen in Fig. 15, green/light solid-dot line), avoiding the risk of being trapped in local minima.

On the other hand, in following figures the performance of the HMPMR-ICIA 8-4-3-2 algorithm is compared with the performance of the MR-ICIA 8-8-8-8 algorithm, which is also based on direct methods; and with the KLT and SIFT algorithms, based on features. Figure 16 shows a collection of images illustrating the performance of the algorithms.

In Fig. 16, it can be seen that the feature-based methods: SIFT (first row) and KLT (second row), failed to track the template, e.g. as seen in Frames 426, 669, and 1,000. The direct method MR-ICIA (third row) failed in some frames (e.g. in Frames 426 and 669). Whereas the HMPMR-ICIA tracked the template in all the frames of the sequence (fourth row).

Additionally, when comparing some of the parameters estimated by each algorithm with the ones of the GT homography, the same results are found. Figure 17 shows the comparison among some of the parameters found by the HMPMR-ICIA (blue/dark solid line), the MR-ICIA (black/dark dashed-dot line), the KLT (red/dark dashed line), the SIFT (cyan/light dashed line), and the H_{GT} (green/light solid line).

In Fig. 17, it can be seen that most of the time, the KLT (red/dark dashed line) and the SIFT (cyan/light dashed line) failed to detect a correct transformation, in spite of the differ-

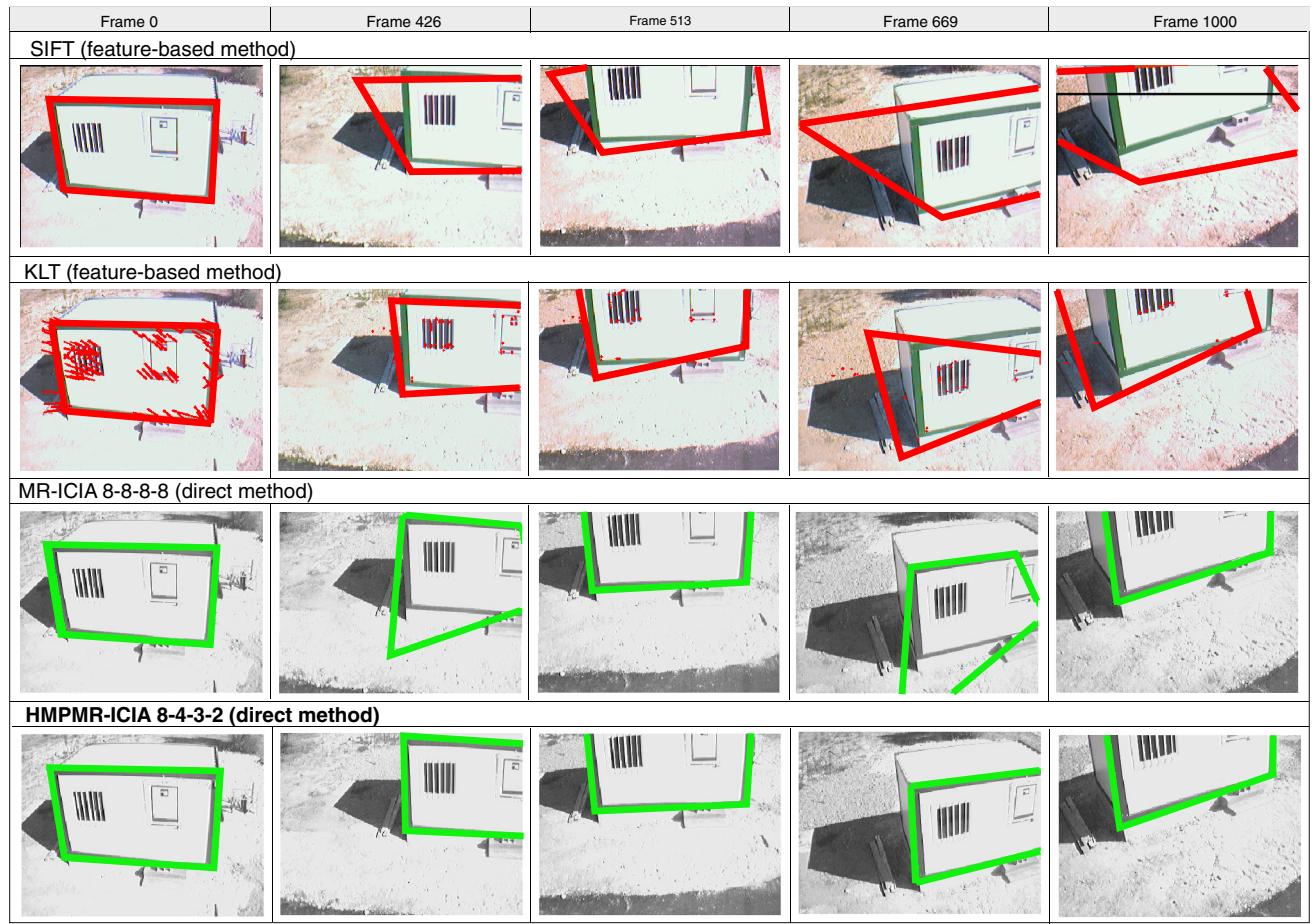


Fig. 16 Comparison of tracking results. The *polygons* indicate the template found by the algorithms

ent features found (the KLT found an average of 85 features and the SIFT found an average of 100 features). The parameters recovered by the tested feature-based algorithms differ from the ones of the \mathbf{H}_{GT} (green/light solid line).

With respect to the algorithms based on direct methods, in Fig. 17 it can be seen that the MR-ICIA (black/dark dashed-dot line) fails in some parts of the sequence (e.g. Frames 426 and 669), but because the position of the template in the following frames coincided with the wrong position estimated by the MR-ICIA, the tracker recovered the template after Frame 775. On the other hand, it can be seen that the values of the parameters estimated by the HMPMR-ICIA algorithm (blue/dark solid line) do have behavior and values that are similar to the ones of the ground truth data (green/light solid line).

Table 2 shows the average speed of the four algorithms. From this table it can be seen that the KLT obtained the fastest speed. This is why the KLT algorithm is one of the most widely used approaches for tracking objects in aerial images (due to its low computational cost; concerning this, see [10, 14]). The SIFT, on the other hand, was the slowest

one. This is due to the high computational overheads in the different steps of the algorithm: e.g. the calculation of the descriptor for each point, matching of points, etc. In the case of the tested configurations of direct methods, it can be seen that by adding the MP structure, the HMPMR strategy tracks the template faster (16 FPS) than using only an MR strategy (8 FPS), making it possible to reach real-time frame rate.

6 Public dataset tests

In this section, the proposed strategy is tested with the Zimmermann's database [49], which has available ground truth data for each frame ($\approx 12,000$ frames). The sequences contain images of three planar objects: a MOUSEPAD (M), a TOWEL (T), and a PHONE (P). Although there are several publicly available datasets with ground truth information, we have found that this data set is the one that is best suited to test our tracking algorithm: e.g. the kind of template used is planar. Additionally, the kind of motions presented in the sequences is appropriate for the objective of the experiments:

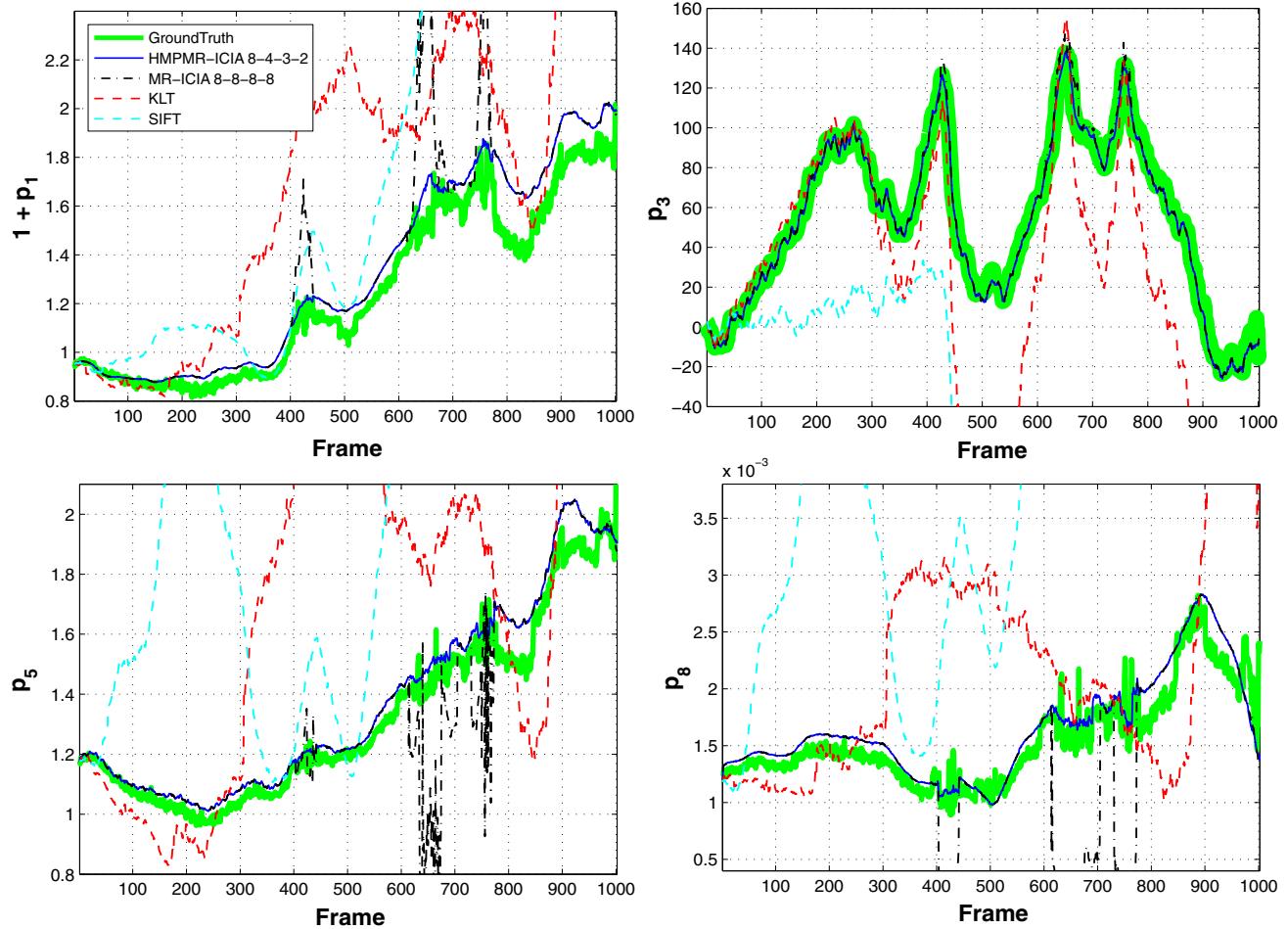


Fig. 17 Comparison with \mathbf{H}_{GT} data. The parameters: p_1 , p_3 , p_5 , and p_8 of the homography estimated with the tested algorithms are compared with the ones obtained with the GT homography (green/light solid line) (color figure online)

to analyze the performance of the HMPMR-ICIA algorithm under fast scale, perspective, and rotation changes; and also under partial occlusions of the object; which are examples of the motions encountered in our application (aerial images).

The same evaluation criteria shown in [49] are used to analyze the performance of the algorithm: the average error in object corners, expressed in percentage and normalized by the actual size of the object upper edge; and the loss-of-locks defined in [49], as the cases where the error was higher than 25 % in at least one of the corners. In those frames, the tracker was reinitialized using the ground truth positions.

The HMPMR configurations used in these sequences were created according to (6): HMPMR-ICIA 8-8-4-3-2-2, for the M sequence; HMPMR-ICIA 8-8-4-2-2, for T and P sequences. Taking into account that the template images are big in the three sequences ($\approx 500 \times 400$ pixels) and that the number of pyramid levels is also high, to achieve real-time frame rates, not all the pixels of the template in levels 0 and 1 were considered in the estimation of the motion model.

Figure 18 shows results of the tracking task conducted by the HMPMR-ICIA algorithm. The first row shows the results

Table 2 Speed comparison feature-based methods and direct methods

Algorithm	MP structure	Frame rate (FPS)
KLT		27
SIFT		3
MR-ICIA	8-8-8-8	8
HMPMR-ICIA	8-4-3-2	16

of the MOUSEPAD (M) sequence, and the second and third rows show the results of the TOWEL (T) and the PHONE (P) sequences, respectively. As can be seen in Fig. 18, these sequences contain strong motion blur (e.g. images in the first and third rows); partial occlusions of the template (e.g. first row images); template going partially out of sight (e.g. images in the second and third rows); scale, rotation, and perspective changes; and repetitive structure of the object (e.g. phone buttons), among other features.

Table 3 compares the results obtained with the proposed HMPMR-ICIA algorithm with the results of state of the art algorithms that were applied to these sequences, which were



Fig. 18 Results of the HMPMR-ICIA applied to the Zimmerman's database. The *first row* shows the results of the MOUSEPAD sequence; the *second and third rows* show the results of the TOWEL and the

PHONE sequences, respectively. The sequences contain strong motion blur, partial occlusions, and fast 3D changes (e.g. scale, rotation, perspective)

reported in [13, 49] and [20]: NoSLLiP [49] (tracker formed by a Number of Sequences of Learned Linear Predictors), SIFT [27], Lucas-Kanade tracker [28], the ICIA [4] (called IC in [49]), LLiP LS [24] (Learned Linear Predictors learned by the Least Squares method), ML-ALPs [20] (Multilayer Adaptive Linear Predictors), and ZSPs [13] (Ultra-fast tracking based on zero-shift points). Most of these algorithms have been tested with the MOUSEPAD sequence which is the longest one.

Comparing the results, in Table 3 it can be seen that the performance of the HMPMR-ICIA in the different sequences is comparable with that of state of the art algorithms. The obtained number of loss-of-locks and the errors are low in the three sequences. From this table, it can be seen that when the SIFT is applied to the MP sequence, the frame rate is slower and the number of loss-of-locks is higher than the HMPMR-ICIA. On the other hand, it can also be seen that using the HMPMR strategy, the results of the ICIA are improved (this was also found when the SIFT and the ICIA algorithms were tested with aerial images in Sect. 5.2).

On the other hand, Fig. 19 shows some of the images where the HMPMR-ICIA failed, i.e. where loss-of-locks were detected. In the M sequence the four loss-of-locks detected correspond to situations with acute perspectives after fast chaotic motions (see Fig. 19, first row). The num-

ber of loss-of-locks obtained by the HMPMR-ICIA in this sequence (M) are smaller than the ones obtained by most of the other algorithms. Additionally, Fig. 19, second row, shows examples of the frames where the loss-of-locks count increased for the P (on the right) and T (on the left) sequences.

With the P sequence, the loss-of-locks were found when the object was inclined, in front of the camera producing acute perspective effects in the image plane. In these cases, the HMPMR-ICIA did not lose the template completely, but the motion model was not the correct one. On the other hand, the motions found in the T sequence are very similar to the ones experienced in the application of tracking using cameras on-board aerial vehicles. This sequence contains strong motion blur (handheld camera), as can be seen in Fig. 19, second row. Concerning this, the HMPMR-ICIA has shown to be robust to this kind of perturbation, obtaining the lowest loss-of-locks count of the algorithms that have been tested with this sequence (see Table 3).

7 The HMPMR strategy applied on-board UAVs

The different tests described in the previous sections have shown that the proposed HMPMR strategy is able to track objects under different conditions, recovering low and large

Table 3 Results Zimmerman's database

Method	Object	Frame rate (fps)	Loss-of-locks (–/–)	Error (%)
HMPMR-ICIA	M	15.25	4/6,849	1.7
NoSLLiPa ^a	M	18.9	13/6,935	1.5
SIFT ^a	M	0.5	281/6,935	1.4
LK (ICIA) ^a	M	2.5 (25)	398/6,935	2.4
LLiP LS ^a	M	24.4	1,083/6,935	6.3
LLiPLS 1/2 ^a	M	24.2	93/6,935	3.0
ZSPs ^b	M	–	92	–
ML-ALPs ^c	M	17.2	1/6,945	2.1
HMPMR-ICIA	T	16.8	1/3,203	1.1
NoSLLiPa ^a	T	21.8	5/3,229	2.1
ZSPs ^b	T	–	8	–
HMPMR-ICIA	P	17.3	4/2,259	1.1
NoSLLiPa ^a	P	16.8	20/1,799	1.8
ALPs ^c	P	96.7	10/2,299	1.2
ZSPs ^b	P	–	17	–

Comparison of the results of the HMPMR-ICIA applied to Zimmerman's database [49] with the results reported in [13, 49], and [20]

^a Results reported in [49]

^b Results reported in [13]

^c Results reported in [20]

numbers of parameters with a performance that is comparable or even better than the one obtained with other algorithms. In this section, the HMPMR-ICIA algorithm is used for tracking planar templates on-board UAVs.

7.1 Test 1: performance during a landing task

In this test, the performance of the HMPMR-ICIA algorithm under the visual conditions present in a vision-based landing task (large frame-to-frame motions and rapid changes in scale) is analyzed. The HMPMR-ICIA 8-4-3-2 algorithm is used to track a planar template located on the ground, which simulates the landing area. A UAV (a quadrotor) flies over the landing area, and an on-board camera (USB camera) placed in a downwards-looking configuration is used to capture the images (of 640×480 pixels size). This image sequence contains large frame-to-frame motions >10 pixels, and rapid changes in scale (e.g. Fig. 20, Frame 696) commonly found during a landing task. In this test, the performance of the HMPMR-ICIA algorithm is compared with the KLT feature-based algorithm.

Figure 20 shows a collection of images that illustrate the performance of the tracking algorithms. As can be seen in the images, the template that is used allows the detection of a large number of features, which helps the KLT to track the template in almost all the sequence. However, in the final

part of the task (see Fig. 20, Frame 696), large frame-to-frame motions (>20 pixels) make the KLT algorithm fail.

On the other hand, as can be seen in Fig. 20, the HMPMR-ICIA algorithm was able to track the template in all the sequence, in spite of the large frame-to-frame motion, vibrations (due to the UAV's movements), perspective changes (e.g. Fig. 20, Frame 682), and scale changes (e.g. Fig. 20, Frames 488 and 696), among other challenging factors.

In general terms, both algorithms performed well tracking the helipad during the landing task, except for the final part of the task where the KLT lost the template. The average frame rates reached during the task were 22 FPS (KLT) and 17 FPS (HMPMR-ICIA). In light of this test, it is possible to see that the performance and the speed reached by the direct method (HMPMR-ICIA) are appropriate for vision-in-the-loop applications.

7.2 Test 2: vision-based position estimation

In this test, vision-based position and orientation data of a UAV estimated using a direct method (the HMPMR-ICIA 8-4-3-2) and a feature-based method (the KLT) are analyzed. The performance of the algorithms is examined comparing the vision-based position estimations with the UAV's state estimation obtained from other on-board sensors (the GPS/IMU). Both algorithms recover the homography at the highest level of the pyramid.

Figure 21 shows the state of the UAV during the test (green/light line) and some of the captured images. An on-board camera (FireWire camera) placed in a downward-looking configuration is used to capture the images (640×480 pixels). The UAV (a helicopter) is flying over a helipad following the trajectory described in Fig. 21: the UAV moves to the left (the object in the image moves to the right, see Frame 114), and then the helicopter moves forward (the object seen in the image plane moves backward, see Frame 386).

A collection of images illustrating the results of the tracking task obtained by the HMPMR-ICIA (first row) and the KLT algorithms (second row) is shown in Fig. 22. As can be seen in Frames 224 and 384, the adverse conditions of the task (constant vibrations) make the KLT algorithm unable to track the template in all the frames of the sequence, whereas the HMPMR-ICIA algorithm tracked the template properly (see Fig. 22, first column).

The known dimensions of the helipad are used to define the 3D coordinates of the helipad with respect to the world reference frame located in its center. With these 3D positions and the 2D positions of the corners of the helipad in the image plane recovered by the tracking algorithms (the HMPMR-ICIA and the KLT), the homography is estimated and decomposed [34], to recover the 3D motion of the UAV. The vision-based positions are obtained with respect



Fig. 19 Loss-of-locks of the HMPMR-ICIA. The *first row* shows the results of the MOUSEPAD sequence; and the *second and third rows* show the results of the TOWEL and the PHONE sequences, respectively

to the camera coordinate system shown in Fig. 21, and then are transformed to the UAV coordinate system shown in Fig. 21.

The different plots of Fig. 23 show the comparison of the UAV's position and orientation estimated by the GPS/IMU sensors (green/light solid line) with the one calculated with the data obtained with the tracking algorithms: the KLT (red/dark dashed line) and the HMPMR-ICIA (blue/dark solid line). In the plots, it can be seen that the homography recovered by the KLT algorithm does not allow a good reconstruction of the position and orientation of the UAV (the KLT algorithm was unable to track the template in all the frames). Conversely, the HMPMR-ICIA (blue/dark solid line) algorithm not only tracked the template, but the position and orientation estimations (blue/dark solid line) also show a behavior that is similar to the one obtained by the GPS/IMU

sensors (green/light solid line). The RMSE (root mean square error) for the estimations based on the HMPMR-ICIA are [0.31, 0.25, 0.16 m] for the X_w , Y_w , Z_w axes, respectively; and 1.7° for the yaw angle.

8 Discussion

Tests were conducted to analyze the performance of the HMPMR-ICIA for tracking planar structures (or structures that can be assumed to be planar) under the presence of large frame-to-frame motion, scale and perspective changes. Different feature-based algorithms and different configurations of the ICIA algorithm (a direct method) to estimate different motion models were tested.

With the ICIA without hierarchies, the MR-ICIA, and the feature-based algorithms, the template was not correctly tracked in all the tested sequences. In this regard, we could see that the well-known MR approach is sometimes not enough to overcome frame-to-frame motions that are >5 pixels, whereas a well-configured HMPMR strategy can deal with larger motions >5 pixels, that can sometimes reach 30 pixels, as was shown in Fig. 6b.

An important aspect of the proposed HMPMR strategy is the selection of the MP hierarchy. In the tests, we have presented an analysis of the distribution of parameters in the MR structure and its influence in the tracking results. It has been found that the most critical level is the one with the lowest resolution image (the highest level). We have proved that to have a well-conditioned Hessian, and to overcome large motions taking advantage of low-resolution information, the translation motion model should be estimated in this level. On the other hand, in the highest resolution level the selected motion model should be the one that describes the motion of the object with the smallest numbers of parameters (in order

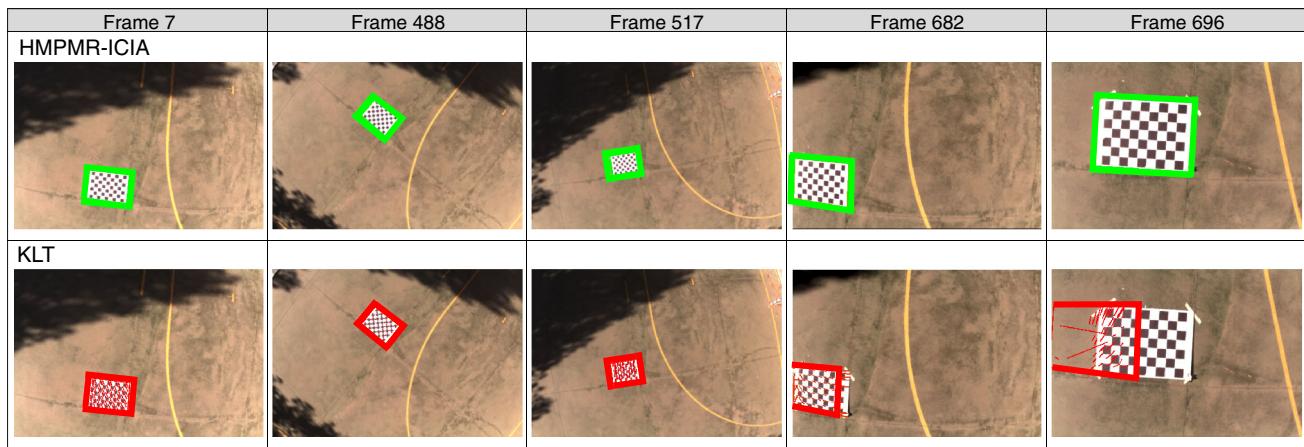


Fig. 20 Comparison of tracking algorithms: the HMPMR-ICIA and the KLT. The performance of the algorithms is analyzed under the visual conditions present in a vision-based landing task (e.g. large frame-to-frame motions and rapid changes in scale)

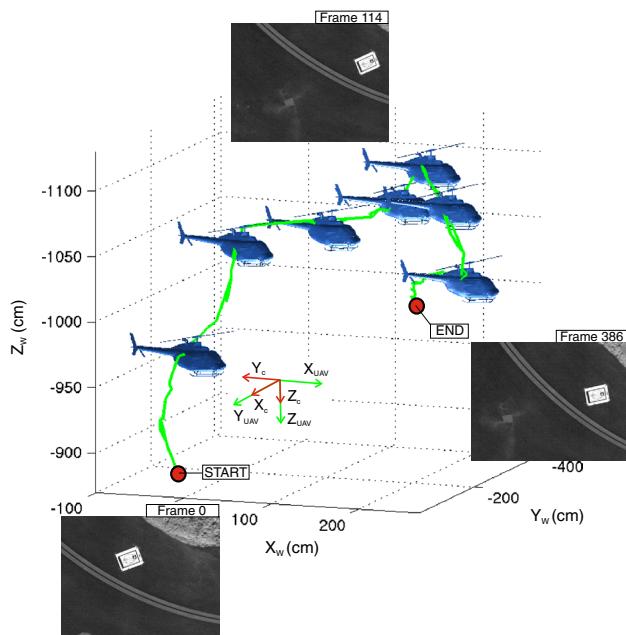


Fig. 21 Flight test. A helicopter is flying over a helipad, and the on-board camera is used to capture the images. On-board sensors (GPS/IMU) are in charge of estimating the UAV's state (green line) and will be used to compare the vision-based results (color figure online)

to obtain a fast response and to avoid local minima), and in the intermediate levels the parameters should be selected to have a smooth transition of parameters between the lowest and the highest levels of the pyramid.

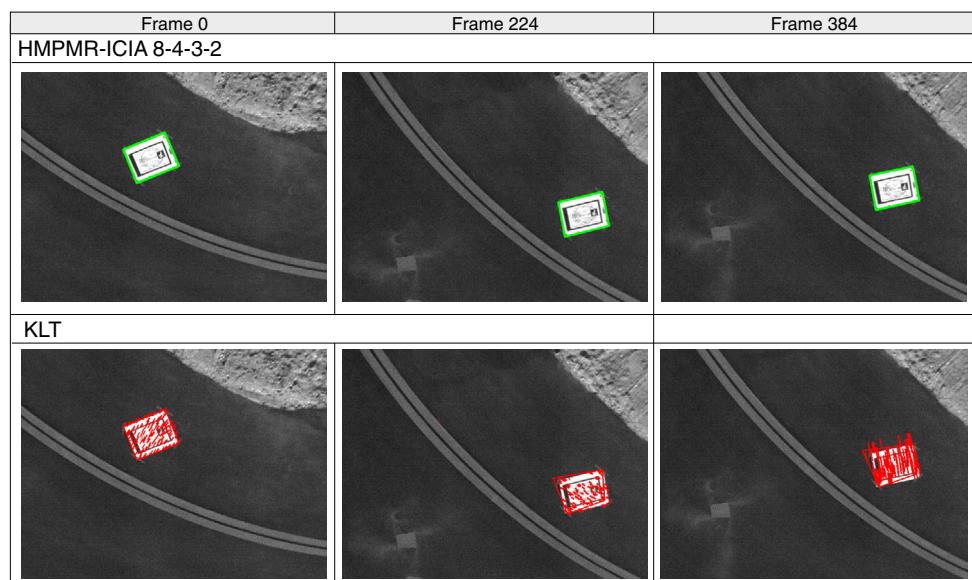
One of the advantages of the ICIA algorithm, in terms of efficiency, is that an important part of the algorithm (e.g. the Hessian \mathbf{H}^j) is calculated at the beginning of the tracking task (when the template is selected). Taking advantage of this, the eigenvalue analysis of \mathbf{H}^j can give an idea (i.e. a confidence

measurement), in advance, on whether the selected motion model can be estimated with the available image information. However, it cannot predict the behavior of the tracking algorithm, because with the ICIA the hessian is pre-computed (it depends on the selected template and not on the current image). In the tests, it has been shown that the use of the HMPMR structure improves the conditioning of \mathbf{H} in the lowest resolution levels (the smallest eigenvalues are bigger than when only an MR structure was used). Additionally, it was shown that when only an MR structure is used, the extra flexibility in the motion model (having more parameters) in the lower resolution levels sometimes leads to bad local image warps, especially under larger frame-to-frame motions.

One limitation of the HMPMR strategy has to do with the range of motion the algorithm can tolerate, which depends on different factors. One of those is the size of the template. When it is small, the MR structure cannot have many levels, and therefore, the amount of inter-frame motion that the algorithm can tolerate is reduced. Additionally, when the frame-to-frame motion is so excessively large that it does not permit the estimation of the two parameters in the lowest resolution level, then the HMPMR algorithm will be trapped in a local minimum. In these cases, the incorporation of additional information (e.g. from other sensors) could help the algorithm to converge to a minimum.

In this paper it has been shown that the inter-frame motion plays an important role in the performance of the tracking task, especially in the application of tracking on-board UAVs. This information is not available in advance and is a critical datum to determine, for example, the appropriate number of levels of the MR structure. The selection of the number of levels depends strongly on the image quality (as was previously mentioned) and therefore the maximum motion that can be estimated is constrained by this relation. Nonetheless,

Fig. 22 Tracking results: flight test. The *first row* shows the results of the HMPMR-ICIA, and the *second row* the ones of the KLT. The KLT is unable to track the template in all the frames of the sequence, whereas the HMPMR-ICIA algorithm manages to perform the tracking task



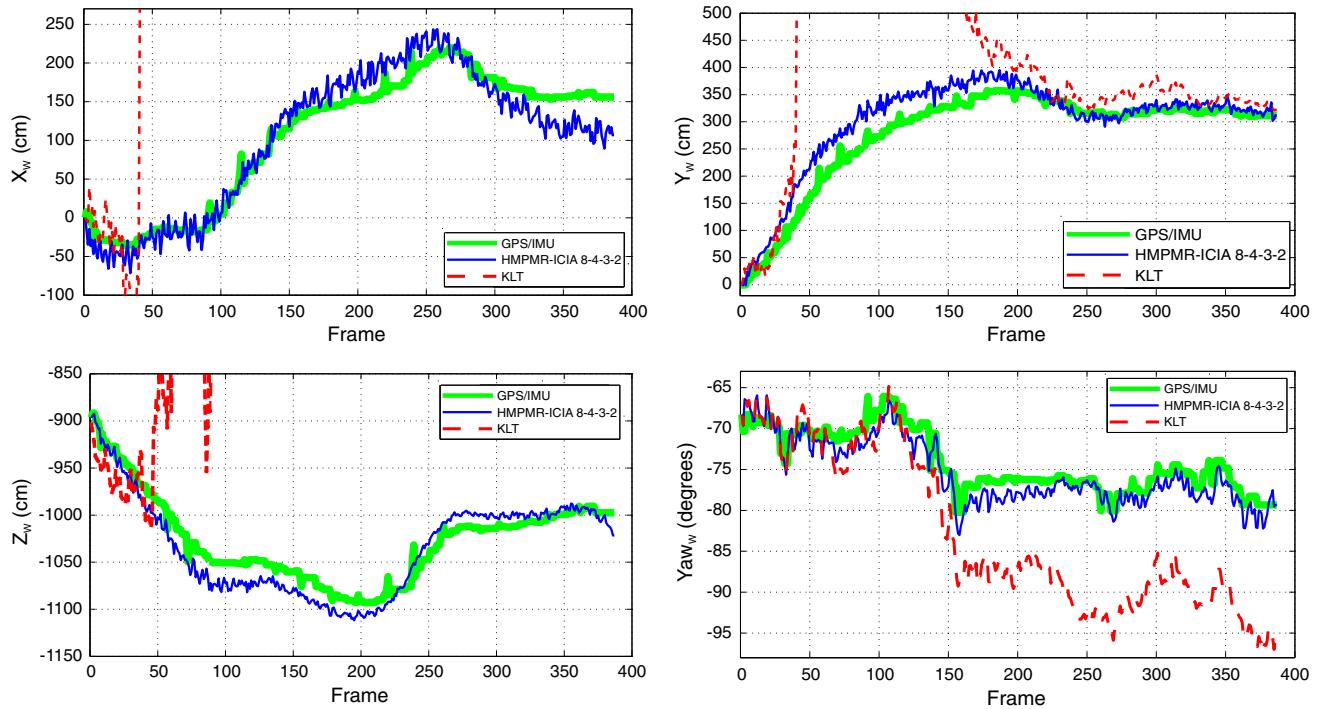


Fig. 23 Comparison of vision-based estimations with IMU/GPS data. The UAV's position and orientation estimated by the HMPMR-ICIA algorithm (blue/dark solid line) present behaviors and values that are similar to the ones obtained by the GPS/IMU sensors (green/light solid

line). The state recovered by the KLT algorithm (red/dark dashed line) does not allow a good reconstruction of the position and orientation of the UAV (color figure online)

we have shown that the adopted criteria to select the number of multi-resolution levels, explained in Sect. 4.1, and the use of the HMPMR strategy, permit to obtain a tracking algorithm that tolerates larger frame-to-frame motions than the ones tolerated by other algorithms, such as the KLT, the SIFT, and the MR-ICA.

Another limitation of the tracking algorithm can be related to the characteristics of the object to track, taking into account that direct methods require texture information to align the images. Additionally, the speed of the algorithm can be another critical factor, if the size of the template is exceedingly large it can make the algorithm run slowly, since direct methods make a pixel by pixel search of the parameters, and this search will be repeated in each level of the HMPMR structure. Nonetheless, despite the amount of information the algorithm analyzes, we have seen that in most of the templates, using the MP and MR strategies at the same time, a robust real-time tracking algorithm can be obtained. Additionally, in Sect. 6 it has been shown that when the template images are big and when the required number of pyramid levels is large, using some of the pixel of the template in the higher resolution levels (e.g. every 3 pixels), robust estimations at real-time frame rates can still be obtained. It is important to notice that the speed is also dependent on the number of parameters estimated; this is why when estimating motion models with a lower number of parameters (e.g.

with the HMPMR), the speed of the algorithm is faster than when using only an MR strategy.

Finally, results have shown that the HMPMR-ICIA can be used in the application of tracking with cameras on-board aerial vehicles, where the requirements of direct methods, especially the small frame-to-frame motion constraint, are easily unsatisfied, due to factors as vehicle vibrations, problems caused by outdoors operations, among others. However, it has been shown that with the HMPMR-ICIA algorithm, it is possible to obtain robust estimations under the presence of large frame-to-frame motion, estimating both simple and complex motion models, and that the information obtained by the tracking algorithm (e.g. homography) can be used for different vision-based applications on-board UAVs (e.g. pose estimation for landing tasks).

Additional tests with different sequences and different configuration of the HMPMR-ICIA algorithm are found in [31], where the algorithm was applied to solve the tracking problem in an aerial refueling task. The videos of the different conducted tests can be seen in Online Resource 1.

9 Conclusions and future work

Previous works in visual tracking have often been based on feature-based methods. In this paper, we have proposed to

address the tracking task with a tracking-by-registration strategy based on direct methods, taking into account that with such methods robust motion estimations can be obtained (the information of every pixel of the object is used to estimate the motion). Nonetheless, the drawbacks of direct methods are that they have good performances when the frame-to-frame motion is small, and due to the amount of information that direct methods have to evaluate, these kinds of methods are mostly used in off-line applications.

Therefore, in this paper, we proposed to use the ICIA algorithm to deal with the efficiency problem of direct methods and also proposed to extend it with a hierarchical strategy that combines a multi-parametric scheme inside a multi-resolution pyramid (the HMPMR), which is in charge of dealing with the small motion constraint of these methods (direct methods have almost always used MR schemes to overcome the small motion constraint). In the paper, a detailed analysis of the HMPMR strategy is presented, characterizing the different components of this framework such as the selection of the MP and MR hierarchies.

The performance of the HMPMR-ICIA algorithm has been compared with state of the art algorithms based both on direct methods and feature-based methods. In the tests, it has been shown that typical MR tracking approaches can easily fail in the presence of partial occlusions and of strong changes in position. It has also been shown that the performance of a tracker based on direct methods is improved when the MP and MR hierarchies are fused, making it able to tolerate large and strong frame-to-frame motions. We have also found that if the HMPMR strategy is adopted with the ICIA algorithm, it is possible to obtain robust estimations at real-time frame rates >16 FPS, even when estimating the homography at the highest resolution level. Therefore with the HMPMR-ICIA, we have been able to apply direct methods in a demanding real-time application: on-board UAVs, where robust and real-time motion estimations are required to develop vision-based control tasks (e.g. landing). Nevertheless, the HMPMR strategy can be extended to other applications (e.g. mosaicing, pose estimation, etc.), even using other image registration algorithms [15, 28].

In the paper, we have also proposed to use the smallest eigenvalue of the Hessian as a way to measure if the selected parameters in each level can be reliably estimated. With the ICIA algorithm we can use this criterion only when the template image is selected and every time it is updated, and not in every image (in the ICIA, the hessian is constant). Therefore, future work will focus on using the smallest eigenvalue criterion to dynamically reconfigure the MP structure of the algorithm every time the template is updated, and also will focus on analyzing other criteria that permit to analyze the number of parameters that can be reliably estimated in every image.

Future work will also focus on establishing criteria to robustly identify when the tracking algorithm has failed or

when the object to track is out of the FOV of the camera, to be able to resume the tracking task when the object reappears in the scene. In this sense, machine learning approaches will be explored for dealing with the recognition problem and also for dealing with the adaptation stage that is required to tackle occlusion problems.

Acknowledgments The work described in this paper is the result of several research stages conducted at the Computer Vision Group of the Universidad Politécnica de Madrid. The authors would like to thank the Universidad Politécnica de Madrid, the Consejería de Educación de la Comunidad de Madrid, and the Fondo Social Europeo (FSE) for the Ph.D. Scholarships of some of the Authors. This work has been supported by the Spanish Ministry of Science under grant MICYT DPI2010-20751-C02-01.

References

1. Anderson, C.H., Bergen, J.R., Burt, P.J., Ogden, J.M.: Pyramid methods in image processing. *RCA Eng.* **29**(6), 33–41 (1984)
2. Baker, S., Datta, A., Kanade, T.: Parameterizing Homographies. *Tech. Rep. CMU-RI-TR-06-11*, Robotics Institute, Pittsburgh (2006)
3. Baker, S., Matthews, I.: Equivalence and efficiency of image alignment algorithms. In: Proceedings of the 2001 IEEE Conference on Computer Vision and Pattern Recognition 1, pp. 1090–1097 (2001)
4. Baker, S., Matthews, I.: Lucas-kanade 20 years on: a unifying framework. *Int. J. Comput. Vis.* **56**(1), 221–255 (2004)
5. Bergen, J.R., Anandan, P., Hanna, K.J., Hingorani, R.: Hierarchical model-based motion estimation. In: *ECCV '92: Proceedings of the Second European Conference on Computer Vision*, pp. 237–252 (1992)
6. Bouguet, J.Y.: Pyramidal implementation of the Lucas Kanade feature tracker: description of the algorithm. Technical report, OpenCV Document, Intel Microprocessor Research Labs (2002)
7. Bradski, G., Kaehler, A.: *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly (2008)
8. Burt, P., Adelson, E.: The laplacian pyramid as a compact image code. *IEEE Trans. Commun.* **31**(4), 532–540 (1983). doi:[10.1109/TCOM.1983.1095851](https://doi.org/10.1109/TCOM.1983.1095851)
9. Can, A., Stewart, C., Roysam, B., Tanenbaum, H.: A feature-based, robust, hierarchical algorithm for registering pairs of images of the curved human retina. *IEEE Transactions on Pattern Analysis and Machine Intelligence* (2002)
10. Cao, X., Lan, J., Yan, P., Li, X.: Vehicle detection and tracking in airborne videos by multi-motion layer analysis. *Mach. Vis. Appl.* **23**, 921–935 (2012)
11. Corral, E.M.: Efficient model-based 3d tracking by using direct image registration. Ph.D. thesis, Facultad de Informática. Universidad Politécnica de Madrid, Spain (2012)
12. Dufaux, F., Konrad, J.: Efficient, robust, and fast global motion estimation for video coding. *IEEE Trans. Image Process.* **9**(3), 497–501 (2000)
13. Dupac, J., Matas, J., Naiser, F.: Ultra-fast tracking based on zero-shift points. *Image Vis. Comput.* **30**(12), 1016–1031 (2012)
14. García Carrillo, L., Rondon, E., Sanchez, A., Dzul, A., Lozano, R.: Stabilization and trajectory tracking of a quad-rotor using vision. *J. Intell. Robot. Syst.* **61**, 103–118 (2011)
15. Hager, G., Belhumeur, P.: Efficient region tracking with parametric models of geometry and illumination. *IEEE Trans. Pattern Anal. Mach. Intell.* **20**(10), 1025–1039 (1998). doi:[10.1109/34.722606](https://doi.org/10.1109/34.722606)
16. Hanna, K., Okamoto, N.: Combining stereo and motion analysis for direct estimation of scene structure. In: *Proceedings of Fourth*

International Conference on Computer Vision, pp. 357–365 (1993). doi:[10.1109/ICCV.1993.378192](https://doi.org/10.1109/ICCV.1993.378192)

17. Hartley, R., Zisserman, A.: *Multiple View Geometry in Computer Vision*. Cambridge University Press, New York (2003)
18. Hess, R.: SIFT feature detector implementation in C. <http://www.web.engr.oregonstate.edu/~hess/index.html> (2007)
19. Hess, R.: An open-source SIFT library. In: Proceedings of the International Conference on Multimedia, pp. 1493–1496 (2010). <http://portal.acm.org/citation.cfm?id=1874256>
20. Holzer, S., Ilic, S., Navab, N.: Multilayer adaptive linear predictors for real-time tracking. *IEEE Trans. Pattern Anal. Mach. Intell.* **35**(1), 105–117 (2013). doi:[10.1109/TPAMI.2012.86](https://doi.org/10.1109/TPAMI.2012.86)
21. Hwangbo, M., Kim, J.S., Kanade, T.: Inertial-aided klt feature tracking for a moving camera. In: IEEE/RSJ International Conference on Intelligent Robots and Systems, 2009. IROS 2009, pp. 1909–1916 (2009). doi:[10.1109/IROS.2009.5354093](https://doi.org/10.1109/IROS.2009.5354093)
22. Irani, M., Anandan, P.: About direct methods. In: *Vision Algorithms: Theory and Practice*, Lecture Notes in Computer Science, vol. 1883, pp. 267–277. Springer, Berlin (2000)
23. Irani, M., Rousso, B., Peleg, S.: Computing occluding and transparent motions. *Int. J. Comput. Vis.* **12**(1), 5–16 (1994). doi:[10.1007/BF01420982](https://doi.org/10.1007/BF01420982)
24. Jurie, F., Dhome, M.: Real time robust template matching. In: Rosin, P.L., Marshall, A.D. (eds.) *British Machine Vision Conference*, BMVC 2002, September, 2002, pp. 123–132. British Machine Vision Association, Cardiff (2002)
25. Kalal, Z., Mikolajczyk, K., Matas, J.: Tracking-learning-detection. *IEEE Trans. Pattern Anal. Mach. Intell.* **34**(7), 1409–1422 (2012). doi:[10.1109/TPAMI.2011.239](https://doi.org/10.1109/TPAMI.2011.239)
26. Kumar, R., Sawhney, H., Samarakera, S., Hsu, S., Tao, H., Guo, Y., Hanna, K., Pope, A., Wildes, R., Hirvonen, D., Hansen, M., Burt, P.: Aerial video surveillance and exploitation. *Proc. IEEE* **89**(10), 1518–1539 (2001)
27. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vis.* **60**(2), 91–110 (2004)
28. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proceedings of the International Joint Conference on Artificial Intelligence, pp. 674–679 (1981)
29. Malis, E., Benhimane, S.: A unified approach to visual tracking and servoing. *Robot. Auton. Syst.* **52**(1), 39–52 (2005). doi:[10.1016/j.robot.2005.03.014](https://doi.org/10.1016/j.robot.2005.03.014). (Advances in Robot Vision)
30. Martinez, C., Mejias, L., Campoy, P.: A multi-resolution image alignment technique based on direct methods for pose estimation of aerial vehicles. In: Proceedings of the International Conference on Digital Image Computing Techniques and Applications (DICTA), pp. 542–548 (2011). doi:[10.1109/DICTA.2011.97](https://doi.org/10.1109/DICTA.2011.97)
31. Martinez, C., Richardson, T., Thomas, P., du Bois, J.L., Campoy, P.: A vision-based strategy for autonomous aerial refueling tasks. *Robot. Auton. Syst.* **61**(8), 876–895 (2013). doi:[10.1016/j.robot.2013.02.006](https://doi.org/10.1016/j.robot.2013.02.006)
32. Mejias, L., Saripalli, S., Campoy, P., Sukhatme, G.: Visual servoing approach for tracking features in urban areas using an autonomous helicopter. In: Proceedings of IEEE International Conference on Robotics and Automation, pp. 2503–2508, Orlando (2006)
33. Mondragon, I.F., Campoy, P., Correa, J., Mejias, L.: Visual model feature tracking for UAV control. In: IEEE International Symposium on Intelligent Signal Processing, 2007. WISP 2007, pp. 1–6 (2007). doi:[10.1109/WISP.2007.4447629](https://doi.org/10.1109/WISP.2007.4447629)
34. Mondragón, I.F., Campoy, P., Martinez, C., Olivares-Mendez, M.: 3D pose estimation based on planar object tracking for UAVs control. In: Proceedings of IEEE International Conference on Robotics and Automation 2010 ICRA2010, Anchorage (2010)
35. Rao, C., Guo, Y., Sawhney, H., Kumar, R.: A heterogeneous feature-based image alignment method. In: 18th International Conference on Pattern Recognition, 2006. ICPR 2006 (2006)
36. Sawhney, H., Kumar, R.: True multi-image alignment and its application to mosaicing and lens distortion correction. *IEEE Trans. Pattern Anal. Mach. Intell.* **21**(3), 235–243 (1999). doi:[10.1109/34.754589](https://doi.org/10.1109/34.754589)
37. Sawhney, H.S., Hsu, S., Kumar, R.: Robust video mosaicing through topology inference and local to global alignment. In: Proc. European Conference on Computer Vision, pp. 103–119 (1998)
38. Sheikh, Y., Khan, S., Shah, M.: Feature-based georegistration of aerial images. In: International Conference on Geosensor Networks (2004)
39. Shi, J., Tomasi, C.: Good features to track. In: 1994 IEEE Conference on Computer Vision and Pattern Recognition (CVPR'94), pp. 593–600 (1994)
40. Shum, H.Y., Szeliski, R.: Panoramic vision. chap. Construction of Panoramic Image Mosaics with Global and Local Alignment, pp. 227–268. Springer, Secaucus (2001). <http://portal.acm.org/citation.cfm?id=378207.378241>
41. Szeliski, R.: Image alignment and stitching: a tutorial. *Found. Trends. Comput. Graph. Vis.* **2**(1), 1–104 (2006). doi:[10.1561/0600000009](https://doi.org/10.1561/0600000009)
42. Szeliski, R., Shum, H.Y.: Creating full view panoramic image mosaics and environment maps. In: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH '97, pp. 251–258. ACM Press/Addison-Wesley Publishing Co., New York (1997). doi:[10.1145/258734.258861](https://doi.org/10.1145/258734.258861)
43. Teuliere, C., Eck, L., Marchand, E.: Chasing a moving target from a flying uav. In: 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 4929–4934 (2011). doi:[10.1109/IROS.2011.6094404](https://doi.org/10.1109/IROS.2011.6094404)
44. Torr, P.H.S., Zisserman, A.: Feature based methods for structure and motion estimation. In: Proceedings of the International Workshop on Vision Algorithms: Theory and Practice, ICCV '99, pp. 278–294. Springer, London (2000)
45. Tsai, Y., Averbuch, A.: Automatic segmentation of moving objects in video sequences: a region labeling approach. *IEEE Trans. Circuits Syst. Video Technol.* **12**(7), 597–612 (2002)
46. Turcajova, R., Kautsky, J.: A hierarchical multiresolution technique for image registration. In: Proceedings of SPIE Mathematical Imaging: Wavelet Applications in Signal and Image Processing (1996)
47. Ye, G.: Image Registration and Super-resolution Mosaicing. Ph.D. thesis, The University of New South Wales (2005)
48. Zhang, H., Yuan, F.: Vehicle tracking based on image alignment in aerial videos. In: EMMCVPR'07: Proceedings of the 6th International Conference on Energy Minimization Methods in Computer Vision and Pattern Recognition, pp. 295–302. Springer, Berlin (2007)
49. Zimmermann, K., Matas, J., Svoboda, T.: Tracking by an optimal sequence of linear predictors. *IEEE Trans. Pattern Anal. Mach. Intell.* **31**(4), 677–692 (2009)
50. Zitová, B., Flusser, J.: Image registration methods: a survey. *Image Vis. Comput.* **21**(11), 977–1000 (2003)



Carol Martínez is a post-doctoral researcher at the Computer Vision Group at the Universidad Politécnica de Madrid (UPM). She received the degree in Mechatronics Engineering from the Universidad Autónoma de Bucaramanga (UNAB), Colombia, in June 2004. In 2007, she joined the Computer Vision Group of the UPM, and since then she has been conducting research in the area of computer vision for Unmanned Aerial Vehicles. In May 2009, she obtained the M.Sc. in Robotics and Automation from the UPM, and in the same year she started her PhD studies. During her studies, in July 2010, she was a research visitor at the Australian Research Center for Aerospace and Automation ARCAA, where she worked in a visual odometry strategy for aerial vehicles; and in June 2011, she was a research visitor at the University of Bristol, UK, where her work was focused on visual tracking for autonomous air-to-air refuelling tasks. In July 2013, she obtained her PhD in Robotics and Automation from the UPM with a thesis on “Visual Tracking, Pose Estimation, and Control for Aerial Vehicles”. Currently, she is a post-doctoral researcher at the UPM working in the I2L intelligent power line inspection project, whose main objective is to reduce the inspection time by means of computer vision and machine learning techniques. Her main research area is the use of computer vision techniques for UAV civilian applications.



Pascual Campoy is Chair Professor on Automatics at the Universidad Politécnica Madrid (UPM), where he received his PhD in Engineering in 1988. He presently lectures on Control, Machine Learning and Computer Vision at graduate and post-graduate level. He has been in charge of a wide variety of research projects concerning technology transfer of Computer Vision to the industry. These projects can be clustered into three main application areas: Automated Visual Inspection, Vision for Robotics and Visual Information Management System. He has been leader of 12 projects funded by the European Commission, 7 National R&D projects and over projects contracted directly by the industry. He is inventor of 9 patents in the field of Computer Vision, three of them registered internationally in several countries all over the word. His research activities have been gathered in more than 100 international publications in technical journals and symposia contributions, having 652 cite and an H number 13. He was the founder of the company “I4: Innovation in Industrial Images Inspection”, which is a spin-off of the U.P.M. and it is aimed to transfer R&D results to the industry in the field of Machine Vision. He is presently leading the Research Group on Computer Vision at U.P.M. He has a large activity in using Vision for Unmanned Aerial Vehicles (UAV),

which includes object tracking, visual servoing, visual control and guidance, visual SLAM, stereo and omni-directional vision. He is presently working on new learning paradigms for bio-inspired Neural Networks that are aimed to be used in image segmentation and detection recognition. He strongly believes in active learning and collaborative learning, being pioneer in using the SCALE-UP methodology in ETSII-UPM since 2007, when the collaborative classrooms were implemented. He is the funder of the ISA Student Association at UPM and its advisor Professor in the period 2004–2012.



Iván F. Mondragón (Colombia, 1980) studied Electrical Engineering at Universidad Nacional de Colombia, obtaining the degree of Electric Engineer (BSEE) in October 2002. He joined the master program at Universidad de los Andes (Colombia) obtaining a M.Sc. in Electronics and Computers Engineering in May 2005. From 2005 to 2006, he worked as a Power Transformers Test Field Engineer at Siemens Andina S.A. (Colombia). After that, he moved to Computer Vision Group at DISAM-ETSII-Universidad Politécnica de Madrid (Spain) obtaining a PhD degree in Automatic and Robotics in November 2011. While completing his PhD, he gained extensive experience on unmanned aerial vehicles, in particular, vision techniques for control and navigation of Autonomous Helicopter and Multi-rotor platforms. Since 2013, he is an assistant professor and researcher at the Industrial Automation Technology Center (CTAI), Department of Industrial Engineering at Pontificia Universidad Javeriana. He is currently working on computer vision applied to Unmanned Aerial Vehicles as well as Flexible Manufacturing Systems FMS and Quality Inspection.



José Luis Sánchez-López is a PhD candidate at the Computer Vision Group of the joint venture Technical University of Madrid (UPM) and Spanish National Research Council (CSIC). He received the degree in Industrial Engineering (September 2010) and the Master in Automation and Robotics (October 2012) at the Technical University of Madrid. In 2009, he joined the Computer Vision Group, and since then, he has been conducting research in the area of Autonomous Unmanned Aerial Systems (UASs). His research goal is to provide UASs with the maximum level of autonomy allowing them to perform different missions without human operation. His research interests are visual localization and mapping, visual control, trajectory planning and multi UAS swarming.

**Dr. Miguel A. Olivares-Méndez**

He got his PhD in 2013 at the Technologic University of Madrid with the Thesis dissertation “Soft Computing based Visual Control for Unmanned Vehicles”. He got his M.Sc. degree in 2009 at Technologic University of Madrid in 2009 on Robotics and Automatics with the dissertation on “Visual Servoing using Fuzzy Controllers on an Unmanned Aerial Vehicle”. Previously, he got his B.S. on Computer Engineer in 2006 at

the University of Malaga with the dissertation on “User Adaptive Fuzzy Control for a Robotic Wheel-chair”. He has done different research stays in the Australian Research Centre for Aerospace Automation (ARCAA) at the QUT, and in the Laboratory of Intelligent System (LIS) at the EPFL. He has more than 7 years experience as a researcher in the design and implementation of fuzzy control system approaches based on vision for different kinds of autonomous vehicles, such as helicopters, quadrotors, and a commercial car. He has also long experience on the uses of computer vision algorithms applied to robotics. He is one of the pioneers in the optimization of fuzzy controllers using the Cross-Entropy method. He has developed a software tool in C++ and ROS for the design, and implementation of fuzzy controllers (MOFS), and a simulator testbed tool using V-REP and ROS for unmanned aerial vehicles (UAVs) and vision-based control.