3D Object following based on visual information for Unmanned Aerial Vehicles

Iván F. Mondragón, Pascual Campoy, Miguel A. Olivares-Mendez, Carol Martinez

Computer Vision Group

Centro de Automtica y Robtica (CAR) -UPM-CSIC

C. José Gutiérrez Abascal 2, 28006 Madrid, Spain

Email: imondragon@etsii.upm.es

web: www.vision4uav.com

Abstract—This article presents a novel system and a control strategy for visual following of a 3D moving object by an Unmanned Aerial Vehicle UAV. The presented strategy is based only on the visual information given by an adaptive tracking method based on the color information, which jointly with the dynamics of a camera fixed to a rotary wind UAV are used to develop an Image-based visual servoing IBVS system. This system is focused on continuously following a 3D moving target object, maintaining it with a fixed distance and centered on the image plane. The algorithm is validated on real flights on outdoors scenarios, showing the robustness of the proposed systems agains winds perturbations, illumination and weather changes among others. The obtained results indicate that the proposed algorithms is suitable for complex controls task, such object following and pursuit, flying in formation, as well as their use for indoor navigation

Keywords- Visual Servoing, UAV, Object Following.

I. INTRODUCTION

Our research interest focuses on developing computer vision techniques to provide UAVs with an additional source of information to perform visually guided tasks - this includes tracking and visual servoing, inspection, autonomous object following, pursuit and flying in formation among others.

Different works have been done where a vision system in conjunction with range sensor was used for Object Following test. On [1] an omnidirectional visual systems is used as bearing sensor, while the distance to the target is measured using a range sensor, for control a robotic wheelchair on indoors. Others systems have been proposed only based on visual information for cooperative robotics [2]. Visual information also have been proposed on aerial robotics for flying in formation [3]. Several approaches also have been proposed for fixed wind UAV flying at constant altitude following circular paths, in order to pursuit a moving object on a ground planar surface [4],[5]. In the same way, several approaches have been proposed for rotary wind UAV following a terrestrial target [6], [7].

Visual servoing also have been successfully implemented on aerial vehicles. Pose-based methods, in which is necessary to estimate the 3D position have been employed for applications like autonomous landing on moving objects [8], while image-based methods have been used for positioning [9], generally assuming a fixed distance to the object, reducing the complex-

ity of the derived controlled and the necessity to estimate the reference depth.

This paper presents a real time flying objects following method based only on visual information to generate a Dynamic Look and Move control architecture based on our previous visual control architecture developed for UAVs [10]. Section II present the flying object following problem statement. Section III explains how an adaptive color tracking method is used to identify and track the target object on the image plane. Then, this information is employed to derive an interaction matrix that relates the features changes on the image plane with the dynamics of the camera fixed to a rotary wind UAV as is presented on section IV. The integration of the developed system on a UAV electric quadcopter is presented in section V. Finally, section VI show the test results of the proposed algorithm running onboard a UAV, validating our approach for an autonomous object flying's following method based on visual information.

II. PROBLEM STATEMENT

Considering a flying object T moving with a unknown trajectory on the world space $\mathbb{R}^3$, and a flying robot O with an attached fixed calibrated pinhole camera, both having an idealized flying dynamics. The control goal is to command the flying robot in order to track the target object, maintaining it always onto the camera FOV with a fixed separation distance. Taking into account the figure 1 and considering the target object as a 3D spherical surface, it is projected on the camera image plane as a circular region that can be defined by its center of projection $\mathbf{x}_t = [x_t, y_t]^T = [\frac{u-c_u}{f}, \frac{v-c_v}{f}]^T$ and the circumference diameter $ø_t$, where (u, v) are the pixel location on projected center on the image, f is the focal length and (c_u, c_v) is the principal point obtained form camera calibration. Because the target is an ideally spherical surface, the projection point $(\mathbf{x}_t)$ can be considered as the image projection of target's sphere centroid with coordinates on the camera frame defined as $\mathbf{X}_{Tc} = [X_{Tc}, Y_{Tc}, Z_{Tc}]^T$. The detected circumference (with a diameter of $ø_t\, pixels$) on the image plane corresponds to the projections of the sphere perimeter (with a fixed diameter $Ø_T$) that results of the intersection of the plane which normal is parallel to the vector defined by the camera optical centre and the sphere centroid, that divides the target

in two hemispheres. The projected diameter also can be used to estimate the distance to the target, because it is inversely proportional to the distance from the camera to the object.
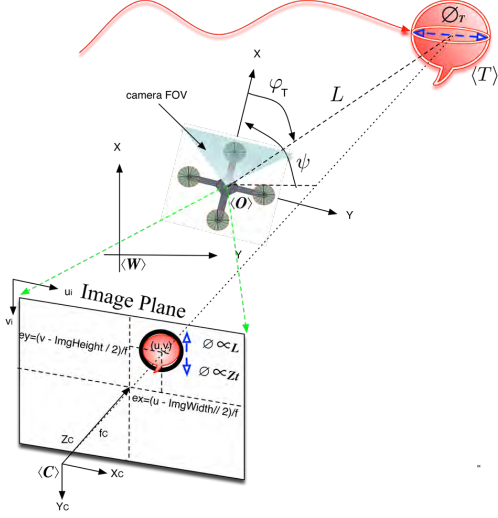

(a)   (b)

Fig. 2. Camshift tracking of a colored red target on a image sequence. The white circle corresponds to the boundaries of the tracked colored area

Fig. 1. UAV object Following setup in 2D. The control goal is to follow a target object (T) with an unknown trajectory using only the pose $((u,v)_t)$ and projection diameter ($\varnothing_t$) on the image plane. The objective is to maintain the Target centered on the image plane with fixed distance or a proportional projection diameter on the image

III. DETECTION APPROACH

Using cameras in outdoor environments is a challenging task. Sudden changes and inconsistencies with outdoor illumination cause changes in the apparent colour as perceived by a camera. Next, we describe the details of the tracking approach used for detect the target.

A. Colour-based probabilistic tracking approach

We approach the problem of tracking by exploiting the colour characteristic of the target. We define a basic colour to the target by assuming a simple coloured mark to it and tracking this mark. Therefore, we rely on a suitable and consistent colour representation that allows us to keep colour distributions derived from video image sequences approximately constant (in outdoor settings). However, this process is not always perfect, and changes still occur in colour distributions over time. An algorithm that has proven to deal with this issue by dynamically adapting to changes in probability distributions is the *Continuously Adaptive Mean Shift* [11] (CamShift). This algorithm is based in the *mean shift* originally introduced by Fukunaga and Hostetler [12].

The Camshift algorithm is used to track a defined color on a images sequence, obtaining for each frame the center of the color region and the the circumference that involves the tracked colored area. Figure 2 shows an example of a color tracked sequence using the Camshift tracked of a red object.
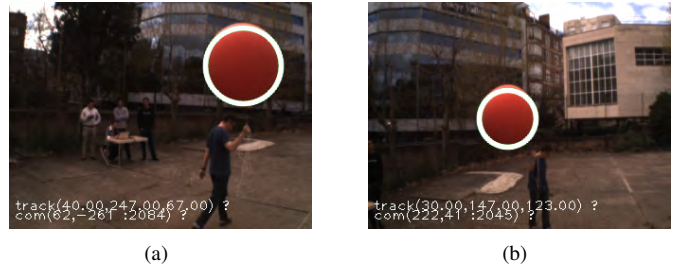
IV. UAV - OBJECT FOLLOWING AND VISUAL SERVOING

Following a similar approach to the one presented on IBVS literature [13],[14],[15], it is possible to define the projected features $\mathbf{s} = [x_t, y_t, \varnothing_t]^T$ as the images characteristics used to define the control. Because the control goal is to maintain alway the target object on the camera FOV with a fixed distance and considering that the projected diameter is inversely proportional to the target distance, the desire position on the image plane can be defined as $\mathbf{s}^* = [x^*, y^*, \varnothing^*]^T = [\frac{u^*-c_u}{f}, \frac{v^*-c_v}{f}, \varnothing^*]^T$, where (u^*, v^*) are the desired pixel location on the image. The error function can be defined as $(\mathbf{e} = \mathbf{s} - \mathbf{s}^*)$.

Considering a general pinhole camera and in order to design a velocity controller, the features are projected on the image plane as equation 1 shows:

$$x_t = \frac{X_{Tc}}{Z_{Tc}}; y_t = \frac{Y_{Tc}}{Z_{Tc}}; \varnothing_t = \frac{\varnothing_T}{Z_{Tc}} \qquad (1)$$

The time derivatives of projected target center and diameter (fixed value) on the image plane are defined on equation 2:

$$\dot{x}_t = \frac{\dot{X}_{Tc}}{Z_{Tc}} - X_{Tc}\frac{\dot{Z}_{Tc}}{Z_{Tc}^2} = \frac{\dot{X}_{Tc}}{Z_{Tc}} - x_t\frac{\dot{Z}_{Tc}}{Z_{Tc}}$$

$$\dot{y}_t = \frac{\dot{Y}_{Tc}}{Z_{Tc}} - Y_{Tc}\frac{\dot{Z}_{Tc}}{Z_{Tc}^2} = \frac{\dot{Y}_{Tc}}{Z_{Tc}} - y_t\frac{\dot{Z}_{Tc}}{Z_{Tc}} \qquad (2)$$

$$\dot{\varnothing}_t = -\varnothing_T\frac{\dot{Z}_{Tc}}{Z_{Tc}^2} = -\varnothing_t\frac{\dot{Z}_{Tc}}{Z_{Tc}}$$

In order to relate the velocity of the point $\mathbf{X}_{Tc}$ to the camera spatial velocity, lets define the spacial velocity of the camera as $\mathbf{V}_c = (\mathbf{v}_c, \omega_c)$, with $\mathbf{v}_c = [v_x, v_y, v_z]^T$ the camera's instantaneous linear velocity and $\omega_c = [\omega_x, \omega_y, \omega_z]^T$ the instantaneous angular velocity of the camera. The time derivative of the point $\mathbf{X}_{Tc}$ is defined as $\dot{\mathbf{X}} = -\mathbf{v}_c - \omega_c \times \mathbf{X}_{Tc}$ as is shown on equation 3

$$\dot{X} = -v_x - \omega_y Z + \omega_z Y_{Tc}$$
$$\dot{Y} = -v_y - \omega_z X + \omega_x Z_{Tc} \qquad (3)$$
$$\dot{Z} = -v_z - \omega_x Y + \omega_Y X_{Tc}$$

The time derivative of the point $\mathbf{X}_{Tc}$ (defined on equation 3) is integrated on equation 2, and considering the features projection defined on equation 1, it is possible to obtain the integration matrix defined on equation 4

$$\mathbf{L}_s = \begin{bmatrix} -\frac{1}{Z_T} & 0 & \frac{x_t}{Z_T} & x_t y_t & -(x_t^2+1) & y_t \\ 0 & -\frac{1}{Z_T} & \frac{y_t}{Z_T} & (y_t^2+1) & -x_t y_t & -x_t \\ 0 & 0 & \frac{\phi_t}{Z_T} & \phi_t y_t & -\phi_t x_t & 0 \end{bmatrix} \quad (4)$$

Equation 4 relates the 6 DOF camera movement with the target projected features on the image plane. Because we only have three features on the image plane it constrains the number of camera DOF that can be controlled. In order to deal with these constrains the following assumptions are made:

- Vehicle roll and pitch are approximately constant, therefore camera angular velocities in X and Z are negligible. $\omega_{X_c} \approx \omega_{Z_c} \approx 0$
- Camera Velocity on axis X is set to zero initially, i.e, $v_X = 0$, $t = 0$ however $v_X \neq 0$, $\forall \; t > 0$

using this assumptions, equation 4 can be written as:

$$\mathbf{L}_s = \begin{bmatrix} 0 & \frac{x_t}{Z_T} & -(x_t^2+1) \\ -\frac{1}{Z_T} & \frac{y_t}{Z_T} & -x_t y_t \\ 0 & \frac{\phi_t}{Z_T} & -\phi_t x_t \end{bmatrix} \quad (5)$$

The relationship betwen the camera velocity $\mathbf{V}_c$ and the time variation of the error can be defined as $\dot{\mathbf{e}} = \mathbf{L}_e \mathbf{V}_c$, where $\mathbf{J}_e = \mathbf{J}(\mathbf{s}, L)$. Often, the error function is selected in such a way that there is an exponential decouple decrease of the error ($\dot{\mathbf{e}} + \lambda \mathbf{e} = 0; \lambda > 0$), allowing that the velocity commands transmitted to the low level controller, can be define as equation 6 shows:

$$\mathbf{V}_c = -\lambda \mathbf{L}_e^+ \mathbf{e} \quad (6)$$

Where $\mathbf{L}_e^+$ is the pseudo-inverse of 5 defined as equation 7 shows:

$$\widehat{\mathbf{L}_e}^+ = \begin{bmatrix} 0 & -Z_T & \frac{Z_T y_t}{\phi_t} \\ -Z_T x_t & 0 & \frac{Z_T(x_t^2+1)}{\phi_t} \\ -1 & 0 & \frac{x_t}{\phi_t} \end{bmatrix} \quad (7)$$

Equations 5 and 7 shows that there is an interaction between each one of the Camera DOF with all image features involved on the system. For control proposes, it is desirable to reduce the interaction between the different variables allowing to have a better tuning of the controller. In order to have a measure of the process interaction, the Relative Gain Array RGA defined as $\Lambda = L \otimes (L^{-1})^T$, (where $\otimes$ denotes element-by-element multiplication) is applied to the Interaction Matrix, given as a result the RGA presented on equation 8.

$$\Lambda = \begin{bmatrix} 0 & -x_t^2 & x_t^2+1 \\ 1 & 0 & 0 \\ 0 & x_t^2+1 & -x_t^2 \end{bmatrix} \quad (8)$$

The RGA matrix 8 is evaluated for the steady state ($x_t = 0$) given as a result:

$$\Lambda = \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad (9)$$

The RGA matrix for the steady state shows that the systems can be approximated by a series of "direct control" loops, in which the best input for the control of the Y_c axis velocity (v_y) is the y_t component of the projected point on the image plane. In the same way, the best input for the control of the Z_c axis velocity corresponds to the projected diameter on the image plane ϕ_t and finally ω_y is best controlled by the x_t component of the projected point on the image plane. In steady state other pairs interaction can be considered as negligible. For steady state the interaction matrix 10 can be approximated as

$$\mathbf{L}_s = \begin{bmatrix} 0 & 0 & -(x_t^2+1) \\ -\frac{1}{Z_T} & 0 & 0 \\ 0 & \frac{\phi_t}{Z_T} & 0 \end{bmatrix} \quad (10)$$

The pseudo-inverse of the Estimated Jacobian $\widehat{\mathbf{L}_e}^+$ is then as equation 11

$$\widehat{\mathbf{L}_e}^+ = \begin{bmatrix} 0 & -Z_T & 0 \\ 0 & 0 & \frac{Z_T}{\phi_t} \\ \frac{-1}{x_t^2+1} & 0 & 0 \end{bmatrix} \quad (11)$$

Equations 7 and 11 still needs to measure or estimate the depth at which is the target. However, the projected diameter on the image plane can be considered to be inversely proportional to the distance to the target and taking into account that the diameter range on the image varies from a minimum of one pixel to a maximum of *ImgWidth* pixels (the camera is touching the target), it it possible to make a lineal approximation of the object distance given by $Z_T = \frac{k_Z}{\phi_t}$, where k_Z is a constant value experimentally found.

V. UAV ARCHITECTURE AND CONTROL SYSTEM

We performed experimental tests using a Pelican quadrotor [16] and a moving colored target. The testbed shown in Figure 3 has a low-level stability controller based on PID that uses information from GPS, IMU, pressure altimeter and magnetometer fused using a Kalman filter. This controller is embedded, closed, unmodifiable but gains are tunable. On-board vision processing is achieved using a dual core Atom 1.6 GHz processor with 1 GB RAM, wireless interface and support for several types of USB cameras (mono or stereo). This computer runs Linux OS working in a multi-client wireless 802.11(a,b,g) ad-hoc network, allowing it to communicate with a ground station PC used for monitoring and supervision.

Our vision-based controller (running in the Atom) communicates via serial interface with the low-level autopilot. This controller sends velocity references to the autopilot creating a *dynamic look-and-move* servoing architecture (figure 4). The vision process runs at 20 fps allowing faster responses from the low level controller to the generated vision commands.

Fig. 3. CVG-UPM [17] Pelican QuadCopter testbed used for sense and avoid experiments

VI. TEST AND RESULTS

Each one of the Image-based visual servoing matrixes defined on equations 7 and 11 are separately implemented on the Pelican UAV using a dynamic look-and-move architecture as is presented on figure 4. In this scheme, the velocity references (defined on equation 6) generated by each one of these controllers (running onboard aircraft) are used as a input references for the Pelican Low Level controlled. This low level controller allows to get as input, velocity commands, as well as direct control actions. For this test, the velocity commands generated by the visual system are directly send as input velocities for the Autopilot. This autopilot also allow to independently control the *X, Y, Z* and *Yaw*. *Roll* and *Pitch* angles are no directly controlled by velocity command, but it is possible to control it by means of motor direct control. So, in this control architecture, and considering that the camera is looking forward, the quadrotor *X* and *Z* as well as the *yaw* angle will be controlled by the generated references $(V_{Xq} = V_{Zc}, V_{Zq} = V_{Yc}, \omega_{Zq} = \omega_{Yc})$. The *Y* axis and *Pitch,Roll* angles are controlled by the low level autopilot.
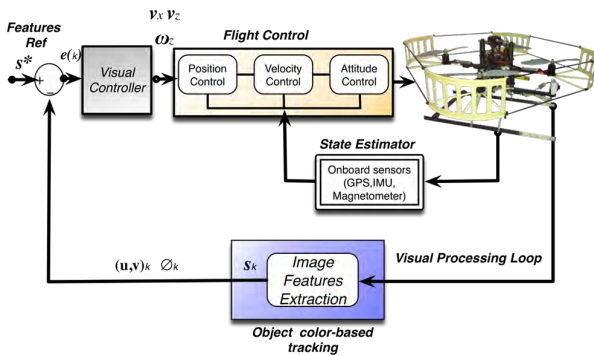
Fig. 4. Object Following *Dynamic image-based look-and-move* system architecture.

Several test have been done in which different balloons has been used as the target object (in order to reduce the complexity of an aerial moving object.). These balloons are moved with a random trajectory on the 3D space. The visual system composed of the camshift color tracked and the IBVS proposed runs onboard the aircraft in a Intel Atom Board PC.

The velocity commands are send to the low level autopilot through a serial interface. Figure 5 shows a external view of two different Object following tests and the onboard images used for the IBVS.
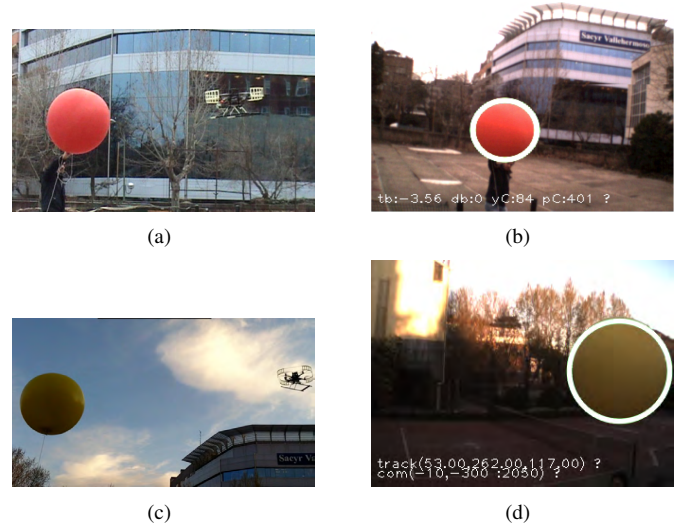

(a)   (b)

(c)   (d)

Fig. 5. Two different Object Following test using a Image-based Visual Servoing IBVS on a Pelican quadrotor: (a),(c) External view of the tests using two balloons with a different color. The balloons are manually move with a random trajectory. (b),(d) The onboard image captured, and the projected diameter on the image plane identified using the Camshift color tracked.

Both controllers (full matrix and reduced version) are tuned using step references with the balloons on static position. In this test the low level controller is commanded by the references generated by the IBVS controller. Figures 6 and 7 shows the response of the controllers presented on equation 7 and 11 to step changes on u^* with the balloon on a static position (some noise is caused by low wind, making the balloon oscillate around the anchor point).

In the same way, figure 8 shows the response of the controllers presented on equation 7 and 11 to step changes on ϕ^* with the balloon on a static position.

Because the dynamics of the quadrotor, the control system can execute negative displacements on the quadrotor *X* axis, allowing to make backward movements, when the desired distance reference is increased (the diameter on the image plane is reduced), as well as when the target object moves towards the follower UAV. This behavior is presented on figure 8.

Figure 9 shows the 2D reconstruction of the flight test presented on figures 7(b) and 8(b). Superimposed images corresponds to captures images from the onboard systems.

Both controllers have also been tested with a moving target following and unknown trajectory. The tests begins when the visual systems correctly identifies and tracks the balloon on the image plane, then the balloon is moved with and almost random trajectory on the 3D space. The command generated by the IBVS controller are transmitted to the UAV low level controller. Figure 10 presents two different tests employing the complete and approximated interaction matrix.
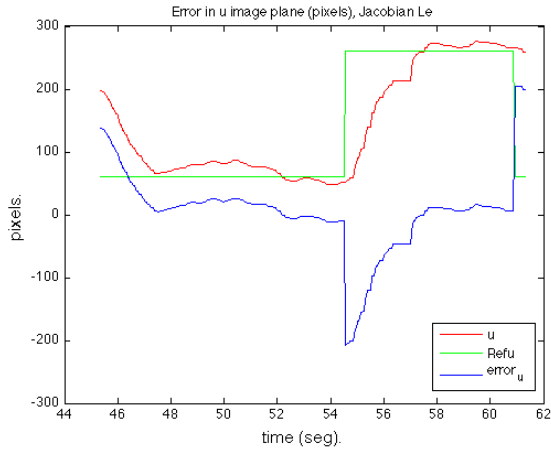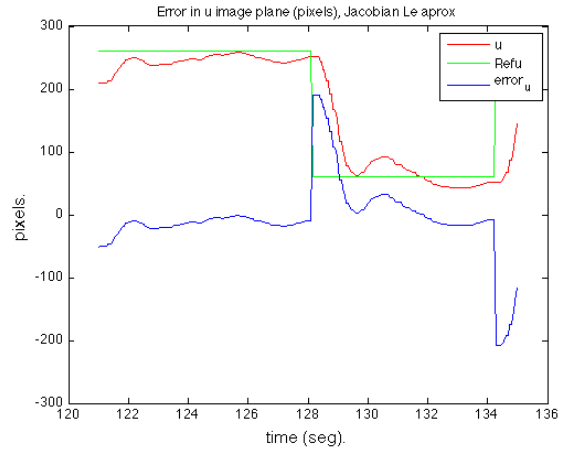
(a)
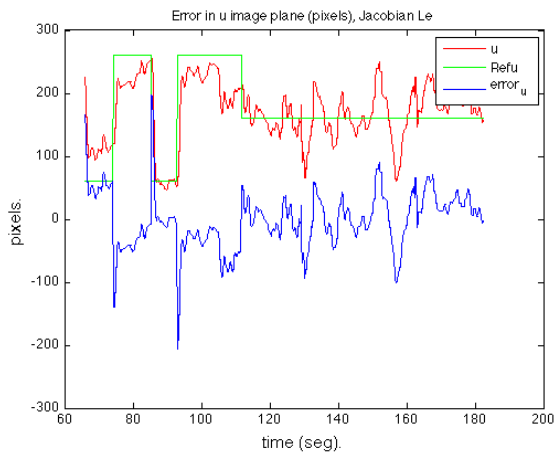
(b)

Fig. 6. $\widehat{\mathbf{L}_e}^+$ IBVS step response on u^* change: (a),(b) Two tests using the Full version of the IBVS presented on equation 7.

(a)

(b)

Fig. 7. $\widehat{\mathbf{L}_e}^+$ IBVS step response on u^* change: (a),(b) Two tests using the approximated version of the IBVS presented on equation 11

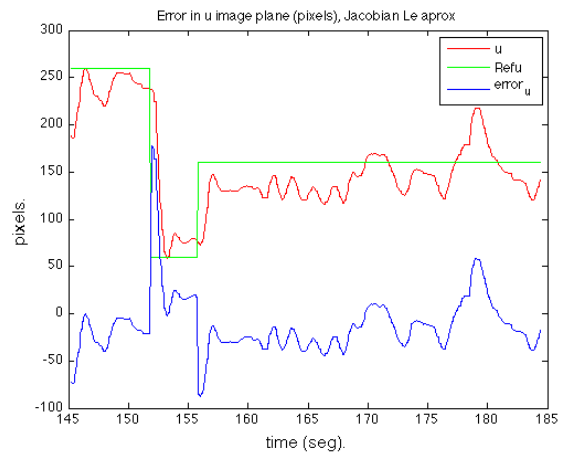Figures 11 and 12 shows the IBVS response given by both controllers for test presented on figure 10.

Tests have shown that there is not a significative difference between the full IBVS interaction matrix presented on equation 7 and the version generated by the analyzes of the Relative Gain matrix for steady stated presented on equation 11. According to this results is also possible to assume that the different degrees of control can be decoupled, allowing to design separately "perfect control" schemes for each one of the involved DOF (exception of Roll and Pitch that are controlled directly by the aircraft Autopilot).

The proposed estimation of the object depth based on the projected diameter on the image plane, is a good approximation allowing to generate an operative IBVS. Both controllers were tested agains step changes on controller references and also using a moving object with an undefined trajectory (including perturbations caused by wind), showing the robustness of the presented controllers.

The video sequences of the test presented on this article and additional test are available at the Vision4UAV project web page [17].

VII. CONCLUSION

In this paper a method for flying object following based only on the visual information provided by a adaptive color tracking algorithm have been presented. The method have been validated experimentally on real test using a rotary wind UAV on outdoor scenarios.

The proposed method has demonstrated to be robust agains weather, wind and light variations performing a correct tracking and following action of the target objective.

The proposed algorithm permits to tracks and follows a moving object on a tridimensional space, without any kind of dynamics restriction. The control task is performed in such a way that a tridimensional control is achieved, allowing to have change on altitude on both, the target and the follower vehicle.

This paper also have derived a decoupled controller based on the analysis of the RGA matrix for the steady state response of the gain matrix derived using the classic IBVS approach.
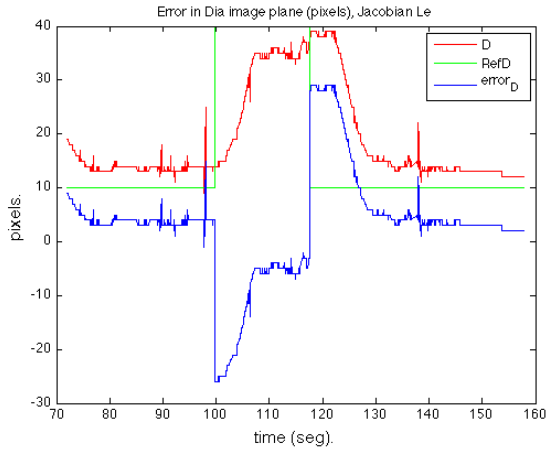
(a)
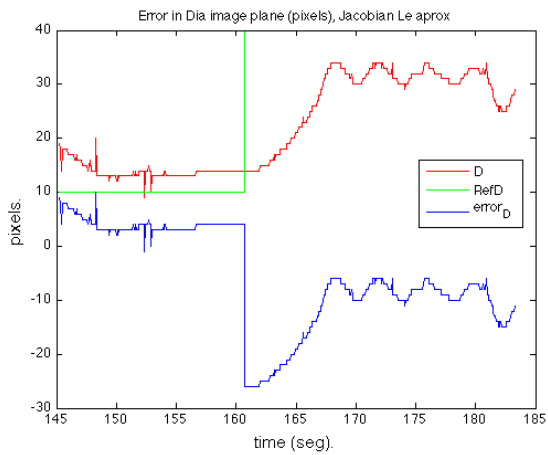
(b)

Fig. 8. $\widehat{\mathbf{L}_e}^+$ IBVS step response on ϕ^* change: (a) test using the Full version of the IBVS presented on equation 7. (b), test using the Approximated version of the IBVS presented on equation 11. (this test corresponds with the test presented on figure 7(b))

Both controllers have been evaluated, showing similar behaviors and demonstrating the effectiveness of this decoupling method.

ACKNOWLEDGMENT

REFERENCES

[1] A. Argyros, P. Georgiadis, P. Trahanias, and D. Tsakiris, "Semi-autonomous navigation of a robotic wheelchair," *Journal of Intelligent and Robotic Systems*, vol. 34, pp. 315–329, 2002.
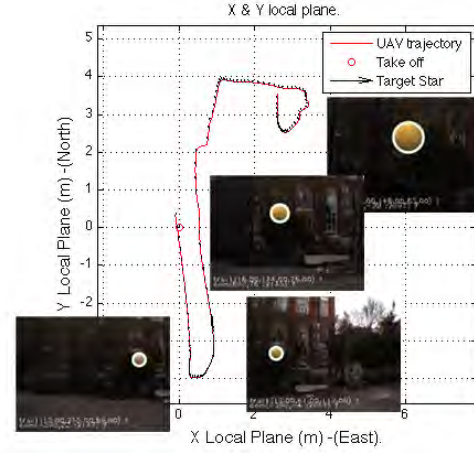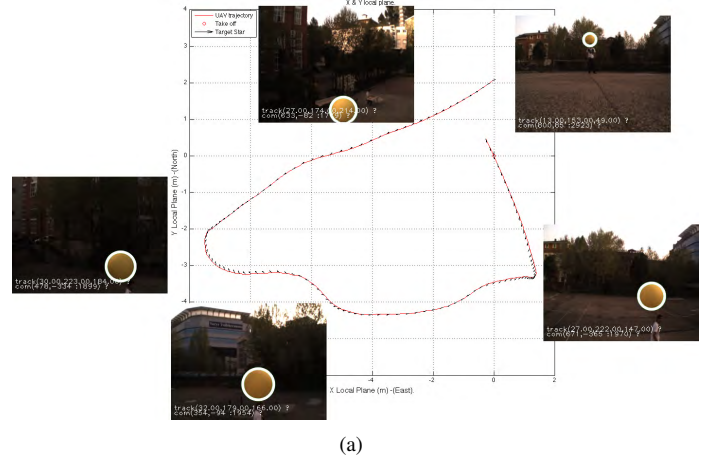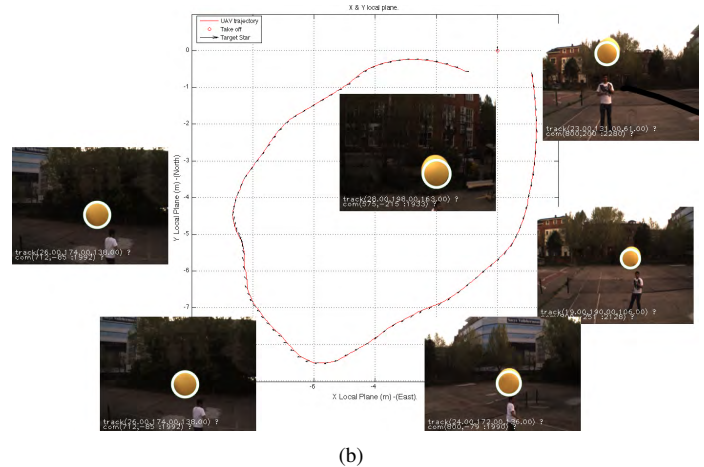
Fig. 9. 2D reconstruction for test with steps changes on u^* and ϕ^*. presented on figures 7(b) and 8(b). This step test was done with an anchor balloon. First, a reference step is applied on u^*, then, the step change is applied on ϕ^*. Superimposed images shows the captured images at different moments of this test.
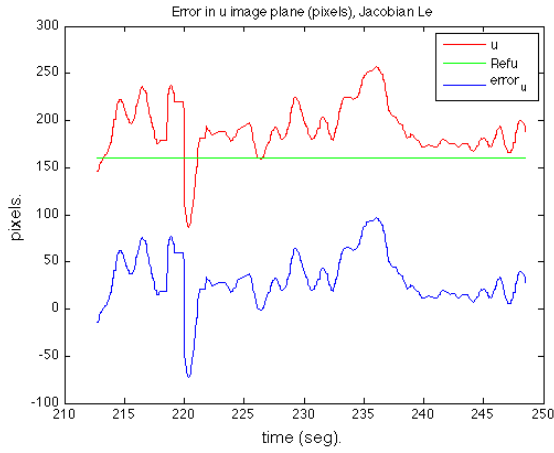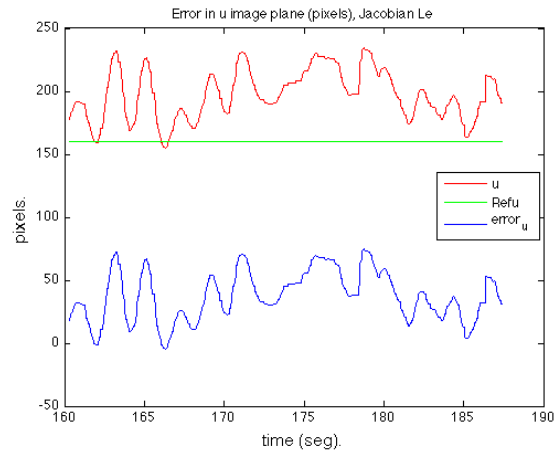

(a)

(b)

Fig. 10. 2D reconstruction for Object following tests: (a) Complete interaction matrix $\widehat{\mathbf{L}_e}^+$. (b) approximated interaction matrix $\widehat{\mathbf{L}_e}^+$.

[2] A. Korodi, A. Codrean, L. Banita, and C. Volosencu, "Aspects regarding the object following control procedure for wheeled mobile robots,"
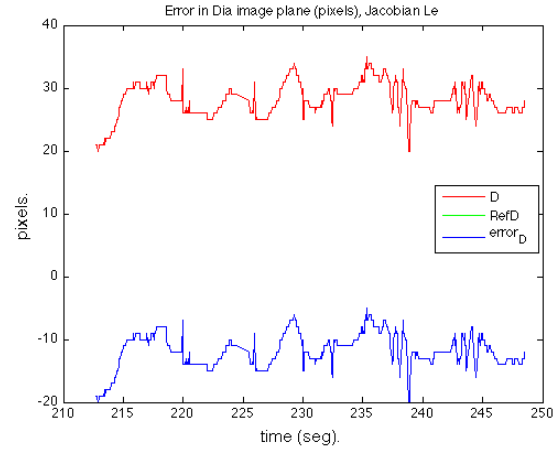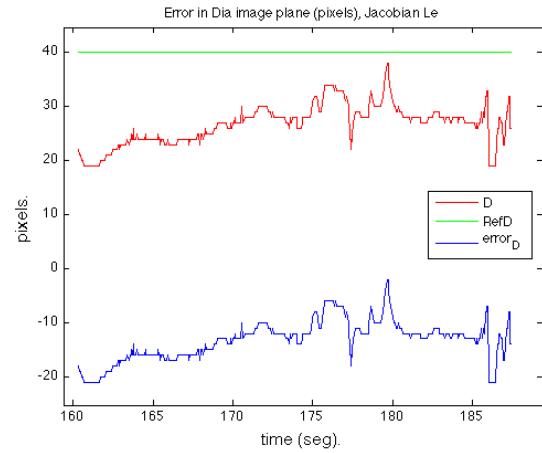
(a)

(b)

Fig. 11. $\widehat{\mathbf{L}_e}^+$ IBVS response on u^* for a moving target following tests: (a) Complete interaction matrix $\widehat{\mathbf{L}_e}^+$.(b) approximated interaction matrix $\widehat{\mathbf{L}_e}^+$.

(a)

(b)

Fig. 12. $\widehat{\mathbf{L}_e}^+$ IBVS response on u^* for a moving target following tests: (a) Complete interaction matrix $\widehat{\mathbf{L}_e}^+$.(b) approximated interaction matrix $\widehat{\mathbf{L}_e}^+$.

WSEAS Trans. Sys. Ctrl., vol. 3, pp. 537–546, June 2008. [Online]. Available: http://portal.acm.org/citation.cfm?id=1482140.1482147

[3] A. Betser, P. Vela, G. Pryor, and A. Tannenbaum, "Flying in formation using a pursuit guidance algorithm," in *American Control Conference, 2005. Proceedings of the 2005*, june 2005, pp. 5085 – 5090 vol. 7.

[4] A. Savkin and H. Teimoori, "Bearings-only guidance of an autonomous vehicle following a moving target with a smaller minimum turning radius," in *Decision and Control, 2008. CDC 2008. 47th IEEE Conference on*, dec. 2008, pp. 4239 –4243.

[5] K. S. Fahd Rafi, Saad Khan and M. Shah, "Autonomous target following by unmanned aerial vehicles," in *Proc. SPIE*, Orlando (Kissimmee), FL, USA, April 2006.

[6] H. Deng, X. Zhao, and Z. How, "A vision-based ground target tracking system for a small-scale autonomous helicopter," *Image and Graphics, International Conference on*, vol. 0, pp. 739–743, 2009.

[7] F. Lin, B. Chen, K. Y. Lum, and T. Lee, "A robust vision system on an unmanned helicopter for ground target seeking and following," in *Intelligent Control and Automation (WCICA), 2010 8th World Congress on*, july 2010, pp. 276 –281.

[8] S. Saripalli and G. S. Sukhatme, "Landing a helicopter on a moving target," in *Proceedings of IEEE International Conference on Robotics and Automation*, Rome, Italy, April 2007, pp. 2030–2035.

[9] O. Bourquardez, R. Mahony, N. Guenard, F. Chaumette, T. Hamel, and L. Eck, "Image-based visual servo control of the translation kinematics of a quadrotor aerial vehicle," *Robotics, IEEE Transactions on*, vol. 25, no. 3, pp. 743 –749, june 2009.

[10] P. Campoy, J. F. Correa, I. Mondragón, C. Martínez, M. Olivares, L. Mejías, and J. Artieda, "Computer vision onboard UAVs for civilian tasks," *Journal of Intelligent and Robotic Systems.*, vol. 54, no. 1-3, pp. 105–135, 2009.

[11] G. R. Bradski, "Computer vision face tracking for use in a perceptual user interface," *Intel Technology Journal*, no. Q2, 1998. [Online]. Available: http://citeseer.ist.psu.edu/585206.html

[12] K. Fukunaga and L. Hostetler, "The estimation of the gradient of a density function, with applications in pattern recognition," *Information Theory, IEEE Transactions on*, vol. 21, no. 1, pp. 32 – 40, Jan. 1975.

[13] F. Chaumette and S. Hutchinson, "Visual servo control. I. basic approaches," *Robotics & Automation Magazine, IEEE*, vol. 13, no. 4, pp. 82–90, 2006. [Online]. Available: http://dx.doi.org/10.1109/MRA.2006.250573

[14] ——, "Visual servo control, part ii: Advanced approaches," *IEEE Robotics and Automation Magazine*, vol. 14, no. 1, pp. 109–118, March 2007.

[15] S. Hutchinson, G. D. Hager, and P. Corke, "A tutorial on visual servo control," in *IEEE Transaction on Robotics and Automation*, vol. 12(5), October 1996, pp. 651–670.

[16] AscendingTechnologies, "Ascending technologies," http://www.asctec.de, 2010. [Online]. Available: http://www.asctec.de

[17] CVG-UPM, "Universidad Politécnica de Madrid. Computer Vision Group. Vision for UAV Project," http://www.vision4uav.com, 2010.