Hovering quad-rotor control: A comparison of nonlinear controllers using visual feedback

Luis Rodolfo Garcia Carrillo, Alejandro Dzul, Rogelio Lozano

HAL Id: hal-00776033
https://hal.science/hal-00776033

Submitted on 15 Jan 2013

Hovering quad-rotor control: A comparison of nonlinear controllers using visual feedback

L.R. García Carrillo, A. Dzul, and R. Lozano

Abstract—This article presents a comparison of three control techniques: Nested Saturations, Backstepping and Sliding Modes. The control objective consists of obtaining the best control strategy to stabilize the position of a quad-rotor unmanned aerial vehicle (UAV) when using visual feedback. We propose a vision-based method to measure translational speed as well as the UAV 3D position in a local frame. The three selected controllers were implemented and tested in real-time experiments. The obtained results demonstrate the performance of such methodologies applied to the quad-rotor system.

Index Terms—Autonomous helicopter, nested saturations, backstepping, sliding modes, visual feedback, position stabilization.

I. INTRODUCTION

Fig. 1: The four-rotor aircraft experimental platform.

With a wide range of applications in both civilian and military scenarios, automatic control of flying machines has attracted the attention of many researches in recent years. Operations of search and rescue, inspection and sensing of remote areas, hazardous material recovery, real-time forest fire monitoring, disaster relief support, surveillance of sensitive areas (borders, ports, oil pipelines), etc., represent some of the tasks where UAV's have demonstrated to be very powerful tools. UAV's can fly autonomously or semi-autonomously and, in addition, they are expendable or recoverable [1].

Several earlier works have already proven the real-time embedded stabilization of a small quad-rotor [2]. In addition, testing the performance of different controls over such systems is a subject already studied. PID and LQR controllers

L.R. García Carrillo is with Heudiasyc, UMR CNRS 6599, Université de Technologie de Compiègne, Compiègne, FRANCE. e-mail: (lgarciac@hds.utc.fr).

A. Dzul is with Instituto Tecnologico de la Laguna, Torreon, Coahuila, MEXICO. e-mail: (dzul@faraday.itlalaguna.edu.mx).

R. Lozano is with Heudiasyc, UMR CNRS 6599, FRANCE and LAFMIA UMI 3175, Cinvestav, MEXICO. e-mail: (rlozano@hds.utc.fr).

are compared in [3], while in [4], the performance of a backstepping and a sliding modes controllers are tested. For the last two examples, experiments were performed over a quad-rotor platform, where 3 degrees of freedom are locked. Authors conclude that backstepping control technique is the most appropriate approach for their future works. In [5], two control methods are studied over a quad-rotor platform equipped with visual feedback. These methods are based on feedback linearization and a backstepping-like control.

Vision systems have become a popular choice for obtaining information that can be used in the feedback control loop of autonomous vehicles. This data is usually combined with an Inertial Measurement Unit (IMU) to provide robust relative attitude information and allowing autonomous positioning and navigation. Also, note that an on-board vision system increases the performance of an UAV equipped with a global positioning system (GPS), which provides position information relative to an inertial frame, but fails in indoor or in noisy environments. On-board computer vision systems provide information which is obtained, for example, from the detection of landmarks. This information allows the UAV to estimate its position in a local frame. Once the UAV knows its position, a control strategy could be implemented in order to achieve a desired position. Several research works aiming at the control of UAV's, using a camera as a vision sensor, can be found in [11], [7], [8], [9], [10]. A vision algorithm for visual navigation and landing of a gas-powered radio-controlled model helicopter, equipped with a PC- 104 is presented in [11]. Concerning electric-powered vehicles, an approach based on optical flow techniques has been applied to the real-time stabilization of an Eight-Rotor UAV in [12]. Also, a two cameras system is presented in [13], with the drawback that the cameras are not embedded on the UAV. A stereo vision system combined with inertial measurements and a laser range finder is proposed in [14], with the purpose of enabling relative localization and local navigation.

In this article, we are interested on experimentally evaluating the performance of three different control strategies for stabilizing a quad-rotor UAV. The aircraft, shown in Figure 1, is equipped with an embedded IMU and gyros, providing Euler angles and angular rates respectively. In addition, we have developed an algorithm for landmark detection and tracking, which estimates the UAV motion (relative position and linear velocity) with respect to a landing pad on the ground, using data provided by an onboard monocular camera [15].

The control objective of our experiments consists of validating the most effective controller for stabilizing the vehicle's position with respect to an artificial visual landmark on the ground. For our tests, we have chosen three controllers

considered between the most commonly reported in the literature: nested saturations [16], [17], backstepping [18], and sliding modes [19]. Experimental results have shown that the nested saturation controllers offer a smoother UAV behavior than the other two controllers, which leads to less energy consumption.

The paper is organized as follows. Section II presents the dynamical equations of the quad-rotor UAV. In Section III we give details of the visual system setup. The control strategies for hovering flight are described in Section IV. The complete quad-rotor experimental platform is presented in Section V. In order to evaluate the effectiveness of the implemented controllers, experimental results are shown in Section VI. Some final concluding remarks are presented in Section VII.

II. Quad-rotor dynamical model

A quad-rotor helicopter is an underactuated dynamic vehicle composed of four input forces (the thrust provided by each propeller) and six output coordinates (fully spatial movements). It could be considered as a vertical takeoff and landing vehicle (VTOL), able to move omnidirectionally and with the ability to fly in hover. The quad-rotor behavior is controlled by varying the angular speed of the rotors. Each rotor produces a thrust and a torque, whose combination generates the main thrust, the yaw torque, the pitch torque, and the roll torque acting on the rotorcraft. In the quad-rotor, the front and rear rotors rotate counter-clockwise while the left and right rotors rotate clockwise, canceling gyroscopic effects and aerodynamic torques in stationary trimmed flight. Vertical motion is controlled by the collective throttle input, that is, the sum of the thrusts of each motor. As we can see in Figure 2, forward/backward motion is achieved by controlling the differential speed of the front and rear motors. This causes the quad-rotor to tilt around the corresponding axis, generating a pitch angle. The left/right motion of the vehicle is achieved by controlling the differential speed of the right and left motors, tilting around the corresponding axis and producing a roll angle. Finally, yaw movement is obtained by taking advantage of the two sets of rotors rotating in opposite direction. Thus, a yaw angular displacement is obtained by increasing (or decreasing) the speed of the front and rear motors while decreasing (or increasing) the speed of the lateral motors. This is done keeping the total thrust constant, then the altitude remains unchanged.

The quad-rotor representation used for this paper is shown in Figure 2. The dynamic model of this aircraft is obtained by representing the quad-rotor as a solid body evolving in 3D and subject to one force and 3 moments [20]. The position of the vehicle's center of gravity, with respect to the inertial frame, is denoted by $\xi = [x\ y\ z]^T \in \mathbb{R}^3$, the three Euler angles (roll, pitch and yaw), which represent the orientation of the vehicle are expressed as $\eta = [\phi\ \theta\ \psi]^T \in \mathbb{R}^3$. The full model rotorcraft dynamics is obtained from Euler-Lagrange equations [16]:
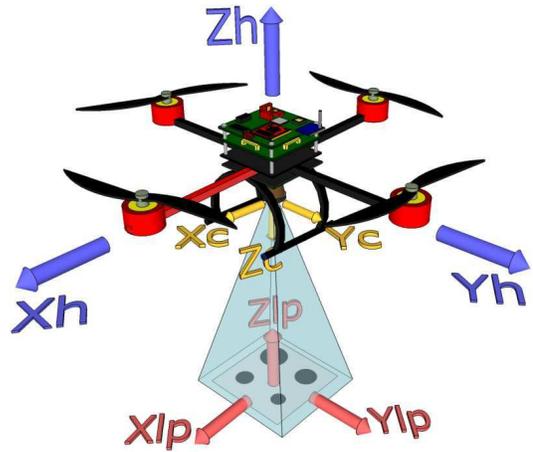

Fig. 2: Visual system setup.

$$m\ddot{x} = u(\sin\phi\sin\psi + \cos\phi\cos\psi\sin\theta) \quad (1)$$
$$m\ddot{y} = u(\cos\phi\sin\theta\sin\psi - \cos\psi\sin\phi) \quad (2)$$
$$m\ddot{z} = u\cos\theta\cos\phi - mg \quad (3)$$
$$\ddot{\psi} = \tilde{\tau}_\psi \quad (4)$$
$$\ddot{\theta} = \tilde{\tau}_\theta \quad (5)$$
$$\ddot{\phi} = \tilde{\tau}_\phi \quad (6)$$

where u is the main thrust directed out of the top of the aircraft, m is the mass of the quad-rotor, g denotes the gravity constant, x and y are coordinates in the horizontal plane, z is the vertical position, and $\tilde{\tau}_\psi$, $\tilde{\tau}_\theta$, and $\tilde{\tau}_\phi$ are the yawing moment, pitching moment and rolling moment respectively, which are related to the generalized torques τ_ψ , τ_θ, τ_ϕ.

III. Visual system setup

Controlling the 3-dimensional position of an UAV depends on the knowledge of the (x, y, z) vehicle coordinates and $(\dot{x}, \dot{y}, \dot{z})$ translational velocities with respect to a well-known reference frame. Such values are required data for the controller in order to generate the control inputs to stabilize the aircraft over a desired location. In order to fulfill this situation, a vision system can be implemented to provide the required position and velocities information. The vision system proposed consists of a calibrated camera onboard the UAV, a landing pad or artificial marker placed on ground, a vision algorithm running on a supervisory ground station PC, and a wireless link between the helicopter and the supervisory ground station. Figure 2 shows the proposed system which can be described as:

- **quadrotor UAV:** with a body fixed frame (X_h, Y_h, Z_h), assumed to be at its center of gravity. Z_h represents the yaw axis, and pointing upwards. X_h and Y_h are the roll and pitch axis respectively.
- **strapdown camera:** pointing downwards, with a reference frame (X_c, Y_c, Z_c). When moving, the camera surveys the scene passing below the quadrotor. Since $X_c - Y_c$ and $X_h - Y_h$ are considered as parallel planes, then visual information collected by the camera can be used to stabilize the vehicle.

- **landing pad:** artificial landmark of known dimensions, formed by four circles of known coordinates, painted on high contrast background and placed underneath the rotorcraft. The coordinates frame (X_{lp}, Y_{lp}, Z_{lp}) represents the inertial reference frame.

The planes formed by $(X_h - Y_h)$ and $(X_{lp} - Y_{lp})$ are considered to be parallel because we assume that the rotorcraft is in hover flight over the landing pad.

A. Vision-based position estimation

In order to estimate the UAV position relative to the landing pad, the extrinsic parameters of the camera are computed at every image frame. This is achieved by implementing an homography estimation technique, which provides the (x, y, z) position and (ψ, θ, ϕ) orientation of the camera with respect to the artificial landmark in the image scene. The action of the homography can be expressed as [21]:

$$\begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = sM \begin{bmatrix} r_1 & r_2 & r_3 & \mathbf{t} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (7)$$

where $[x \ y \ 1]^T$ represents the landing pad position in the camera image, s is a known scale factor, $M \in \mathbb{R}^{3 \times 3}$ represents the intrinsics parameters camera matrix, $R = [r_1 \ r_2 \ r_3] \in \mathbb{R}^{3 \times 3}$ are the extrinsics rotation parameters, $\mathbf{t} \in \mathbb{R}^{3 \times 1}$ is the extrinsics translation parameters vector, $[X \ Y \ Z \ 1]^T$ is the real landing pad position, and $H = sM[r_1 \ r_2 \ \mathbf{t}]$ is the homography matrix. The homography matrix H is divided in two parts: the physical transformation (which locates the observed object plane) and the projection (the camera intrinsic matrix).

Rotation R is described by three angles and translation $\mathbf{t}$ is defined by three offsets; hence there are six unknowns for each view. The known planar object (the artificial landmark) provides eight equations, that is, the mapping of a rectangle into a quadrilateral can be described by four (x, y) image points. For every instant, when the aerial vehicle is in hovering, it is possible to compute the H homography matrix using the a priori knowledge of the position of the four centroids of the circles [22]. Using this estimated transformation matrix and the intrinsic camera matrix previously identified by an off-line calibration based on the method in [23], we are able to calculate the camera extrinsic parameters, and therefore we have the vehicle's (x, y, z) position with respect to the landing pad on the ground.

B. Translational velocities

An optical flow computation procedure is applied to compute the $(\dot{x}, \dot{y}, \dot{z})$ translational velocities of the aerial vehicle with respect to the landing pad. In order to compute optical flow, we have implemented the Lucas-Kanade pyramidal algorithm [24] in combination with a feature-detecting algorithm. This approach provides an accurate estimation of the motion field since it does not take into account the non landing pad areas, where the motion field cannot be accurately determined.

Consider the camera moving with respect to a rigid scene. The velocities and rotation rates of the camera in the inertial frame are expressed by (V_x, V_y, V_z) and (w_x, w_y, w_z) respectively. To accurately estimate the pseudo-speeds of the engine, we define a tracking zone surrounding the landing pad, in a way that the centroid of the zone and the center of the landing pad coincide. The most representative features over the zone are selected as features to track for. These features are usually the circles perimeter. Once this group of features has been identified, a tracking process is performed over the entire image. The optical flow is estimated based on the displacements of the tracked features.

The optical flow computed at point (x_k^i, y_k^i) is composed of a translational and a rotational part as

$$\begin{bmatrix} OF_{x_k^i} \\ OF_{y_k^i} \end{bmatrix} = T_{OF_k} + R_{OF_k} \quad (8)$$

with the translational part

$$T_{OF_k} = \frac{1}{z} \begin{bmatrix} -f & 0 & x_k^i \\ 0 & -f & y_k^i \end{bmatrix} \begin{bmatrix} V_x \\ V_y \\ V_z \end{bmatrix} \quad (9)$$

and the rotational part

$$R_{OF_k} = \begin{bmatrix} \frac{x_k^i y_k^i}{f} & -(f + \frac{(x_k^i)^2}{f}) & y_k^i \\ (f + \frac{(y_k^i)^2}{f}) & -\frac{x_k^i y_k^i}{f} & -x_k^i \end{bmatrix} \begin{bmatrix} \omega_x \\ \omega_y \\ \omega_z \end{bmatrix} \quad (10)$$

where $OF_{j_i}^d$ is the optical flow component in the coordinate j of the point p_i, V_k and ω_k are the translation velocities and rotation rates, respectively, of the body in the coordinate k. Thus, the mean of the optical flow computed on all those points can be expressed as a function of the camera movement as follows

$$\bar{OF}_x = \bar{V}_{OF_x} + K_x \bar{V}_{OF_z} + \bar{R}_{OF_x} \quad (11)$$
$$\bar{OF}_y = \bar{V}_{OF_y} + K_y \bar{V}_{OF_z} v + \bar{R}_{OF_y} \quad (12)$$

Using the results from [25], the rotational optical flow is compensated and the pseudo-speeds $(\bar{V}_{OF_x}, \bar{V}_{OF_y}, \bar{V}_{OF_z})$ are deduced. Since the camera system and the helicopter share the same movements, it can be said that the deduced pseudo-velocities depend of the rotorcraft movement. Indeed, the camera is mounted onboard the quad-rotor and fixed in a way it has no freedom degree. Thus, it can be written that

$$\bar{V}_{OF_x} = -\frac{f\dot{x}}{z} \quad (13)$$

$$\bar{V}_{OF_y} = -\frac{f\dot{y}}{z} \quad (14)$$

$$\bar{V}_{OF_z} = \frac{\dot{z}}{z} \quad (15)$$

where $(\dot{x}, \dot{y}, \dot{z})$ is the speed vector of the rotorcraft center of gravity and z is the altitude. Thus, from these three equations the proposed optical flow vision system allows speed estimation of the rotorcraft up to a scale factor, when flying at constant altitude. Those estimations can be used to control the translational velocities of the rotorcraft.

If the landing pad is not successfully detected in the current image, the vision algorithm will fail. To overcome

this situation, optical flow values can be used to estimate the position of the four circles centroids as

$$\rho_x^{k+1} = \rho_x^k + \Delta_T \bar{V}_{OF_x} \qquad (16)$$
$$\rho_y^{k+1} = \rho_y^k + \Delta_T \bar{V}_{OF_y} \qquad (17)$$

where (ρ_x^k, ρ_y^k) represents the circle's centroid position and Δ_T is the working frequency of the algorithm. This estimated centroids positions are used to estimate the homography at each time the detection of the landing pad fails.

IV. CONTROL STRATEGY

In this section, we describe three control strategies applied to the quad-rotor: nested saturations control method, backstepping approach, and sliding modes controller. All control laws are designed to stabilize the x, y, θ and ϕ states. The altitude z and the yaw angle ψ are stabilized by PD controllers.

A. Altitude and yaw control

The control of the vertical position (3) can be obtained by using the following control input [16]:

$$u = (r_1 + mg)\frac{1}{\cos\theta\cos\phi} \qquad (18)$$

where

$$r_1 = -k_{vz}\dot{z} - k_{pz}e_z \qquad (19)$$

with $e_z = z_d - z$ as the z error position and z_d as the desired altitude. k_{pz} and k_{vz} are positive constants. Thus, for the altitude dynamics, r_1 is a PD controller. In the case of the yaw angular position (4), we can apply

$$\tilde{\tau}_\psi = -k_{v\psi}\dot{\psi} - k_{p\psi}e_\psi \qquad (20)$$

where $e_\psi = \psi_d - \psi$ denotes the yaw error, ψ_d represents the desired yaw angle, $k_{p\psi}$ and $k_{v\psi}$ denote the positive constants of a PD controller. Indeed, introducing equations (18) and (20) into set of equations (1)-(4) and provided that $\cos\theta\cos\phi \neq 0$, we obtain

$$m\ddot{x} = (r_1 + mg)\left(\frac{\sin\psi\tan\phi}{\cos\theta} + \cos\psi\tan\theta\right) \qquad (21)$$

$$m\ddot{y} = (r_1 + mg)\left(\sin\psi\tan\theta - \frac{\cos\psi\tan\phi}{\cos\theta}\right) \qquad (22)$$

$$m\ddot{z} = -k_{vz}\dot{z} - k_{pz}e_z \qquad (23)$$

$$\ddot{\psi} = -k_{v\psi}\dot{\psi} - k_{p\psi}e_\psi \qquad (24)$$

The control parameters $k_{p\psi}$, $k_{v\psi}$, k_{pz} and k_{vz} should be carefully chosen to ensure a stable well-damped response in the vertical and yaw axes [16]. From (23) and (24) it follows that $\psi \to \psi_d$ and $z \to z_d$.

B. Nested Saturations Control

Consider a system given by four integrators in cascade:

$$\dot{x}_1 = \alpha x_2, \quad \dot{x}_2 = \beta x_3, \quad \dot{x}_3 = \gamma x_4, \quad \dot{x}_4 = u \qquad (25)$$

where α, β, $\gamma \neq 0$ are constants. A nested saturations control input can be defined as [16], [17]:

$$u = -\sigma_{b_4}(k_4 z_4 + \sigma_{b_3}(k_3 z_3 + \sigma_{b_2}(k_2 z_2 + \sigma_{b_1}(k_1 z_1)))) \qquad (26)$$

where z_i, for $i = 1\dots4$, denotes a change of variables. $k_i > 0$ are constants, and σ_{b_i} represent saturation functions defined as:

$$\sigma_{b_i}(s) = \begin{cases} -b_i & ; & s < -b_i \\ s & ; & |s| \le b_i \\ b_i & ; & s > b_i \end{cases} \qquad (27)$$

where $b_i > 0$ are constants denoting the bounds of the saturation functions. The z_i's are given by:

$$z_1 = x_4 + \frac{k_4 + k_3 + k_2}{\gamma}x_3 + \frac{k_3 k_4 + k_2 k_3 + k_2 k_4}{\beta\gamma}x_2$$
$$+ \frac{k_2 k_3 k_4}{\alpha\beta\gamma}x_1 \qquad (28)$$

$$z_2 = x_4 + \frac{k_4 + k_3}{\gamma}x_3 + \frac{k_3 k_4}{\beta\gamma}x_2 \qquad (29)$$

$$z_3 = x_4 + \frac{k_4}{\gamma}x_3 \qquad (30)$$

$$z_4 = x_4 \qquad (31)$$

Note that from (19) and (23) $r_1 \to 0$. For a time T large enough, e_z and e_ψ are arbitrarily small, therefore, equations (21) and (22) reduce to

$$\ddot{x} = g\tan\theta \qquad (32)$$

$$\ddot{y} = -g\frac{\tan\phi}{\cos\theta} \qquad (33)$$

1) Control of the forward position and pitch angle: Consider the subsystem given by (5) and (32). Implementing a nonlinear control based on nested saturations allows in the limit a guarantee of arbitrary bounds for x, $\dot{x}$, θ and $\dot{\theta}$. To further simplify the analysis, we will impose a very small upper bound on $|\theta|$ in such a way that the difference $\tan(\theta) - \theta$ is arbitrarily small ($\theta \approx \tan(\theta)$). Therefore, the subsystem (5) and (32) becomes the following linearized system:

$$\ddot{x} = g\theta \qquad (34)$$

$$\ddot{\theta} = \tilde{\tau}_\theta \qquad (35)$$

which represents four integrators in cascade. Then, by using (25)-(26) the controller is given by

$$\tilde{\tau}_\theta = -\sigma_{b_4}\left(\dot{\theta} + \sigma_{b_3}\left(\dot{\theta} + \theta + \sigma_{b_2}\left(\dot{\theta} + 2\theta + \frac{\dot{x}}{g}\right.\right.\right.$$
$$\left.\left.\left. + \sigma_{b_1}\left(\dot{\theta} + 3\theta + 3\frac{\dot{x}}{g} + \frac{x}{g}\right)\right)\right)\right) \qquad (36)$$

It is proved in [16] that θ, $\dot{\theta}$, x and $\dot{x}$ converge to zero. To regulate x around a desired position, we rewrite the most inner term (associated to σ_{b_1}) as $\dot{\theta} + 3\theta + 3\frac{\dot{x}}{g} + \frac{e_x}{g}$, where e_x is the position error, expressed as $e_x = x_d - x$. Here, x_d represents the desired position reference for x.

2) Control of lateral position and roll angle: Consider the subsystem given by equations (6) and (33). Imposing a very small upper bound on $|\phi|$ in such a way that the difference $\tan(\phi) - \phi$ is arbitrarily small ($\phi \approx \tan(\phi)$), the subsystem (6) and (33) becomes

$$\ddot{y} = -g\phi \qquad (37)$$

$$\ddot{\phi} = \tilde{\tau}_\phi \qquad (38)$$

Using a similar procedure to the one proposed for the pitch control, we obtain

$$
\tilde{\tau}_\phi = -\sigma_{b_4}\left(\dot{\phi} + \sigma_{b_3}\left(\dot{\phi} + \phi + \sigma_{b_2}\left(\dot{\phi} + 2\phi - \frac{\dot{y}}{g}\right.\right.\right.
$$
$$
\left.\left.\left. + \sigma_{b_1}\left(\dot{\phi} + 3\phi - 3\frac{\dot{y}}{g} - \frac{y}{g}\right)\right)\right)\right) \tag{39}
$$

In order to regulate y around a desired position, we rewrite the most inner term (associated to σ_{b_1}) as $\dot{\phi} + 3\phi - 3\frac{\dot{y}}{g} - \frac{e_y}{g}$, where e_y is the position error, expressed as $e_y = y_d - y$. Here, y_d represents the desired position reference for y.

C. Backstepping Control

The Backstepping technique provides a systematic method to obtain a control law from a chain of integrators. This methodology was introduced in [18].

1) Control of the forward position and pitch angle: Rewrite subsystem given by (34)-(35) as

$$
\dot{x}_1 = x_2 \tag{40}
$$
$$
\dot{x}_2 = g\theta_1 \tag{41}
$$
$$
\dot{\theta}_1 = \theta_2 \tag{42}
$$
$$
\dot{\theta}_2 = \tilde{\tau}_{B\theta} \tag{43}
$$

where $\tilde{\tau}_{B\theta}$ will define the final backstepping control input. In order to obtain this control input, consider each equation as a new subsystem, where the next state is taken as the input and it is defined as a virtual control to stabilize such a subsystem. Then, for our case, we begin with

$$
\dot{x}_1 = x_2 \tag{44}
$$
$$
\zeta_1 = x_1 \tag{45}
$$

where x_2 represents the input, and ζ_1 the output. Let us propose a positive definite function $V_1 = \frac{1}{2}x_1^2$, whose time derivative is given by

$$
\dot{V}_1 = x_1\dot{x}_1 = x_1x_2 \tag{46}
$$

and consider a virtual input $\alpha_1 = (x_2)^v = -k_1x_1$, where k_1 is a positive constant. Then $\dot{V}_1 = -k_1x_1^2$. Now, let ζ_2 be the new output:

$$
\zeta_2 = x_2 - \alpha_1 \tag{47}
$$

The new subsystem that we are trying to stabilize is written as

$$
\dot{x}_1 = \zeta_2 + \alpha_1 \tag{48}
$$
$$
\dot{\zeta}_2 = g\theta_1 - \dot{\alpha}_1 \tag{49}
$$

and let us propose a positive definite function $V_2 = V_1 + \frac{\zeta_2^2}{2}$, then

$$
\dot{V}_2 = \dot{V}_1 + \zeta_2\dot{\zeta}_2 = \dot{V}_1 + \zeta_2(g\theta_1 - \dot{\alpha}_1) \tag{50}
$$

Define a virtual input $\alpha_2 = (g\theta_1) = -k_2\zeta_2 + \dot{\alpha}_1$, where k_2 is a positive constant. Then, $\dot{V}_2 = \dot{V}_1 - k_2\zeta_2^2$. Now, let ζ_3 be a new output:

$$
\zeta_3 = g\theta_1 - \alpha_2 \tag{51}
$$

The new subsystem to be stabilized is written as

$$
\dot{x}_1 = \zeta_2 + \alpha_1 \tag{52}
$$
$$
\dot{\zeta}_2 = g\theta_1 - \dot{\alpha}_1 \tag{53}
$$
$$
\dot{\zeta}_3 = g\theta_2 - \dot{\alpha}_2 \tag{54}
$$

and let us propose the positive definite function $V_3 = V_2 + \frac{\zeta_3^2}{2}$, then

$$
\dot{V}_3 = \dot{V}_2 + \zeta_3\dot{\zeta}_3 = \dot{V}_2 + \zeta_3(g\theta_2 - \dot{\alpha}_2) \tag{55}
$$

Define a virtual input $\alpha_3 = (g\theta_2) = -k_3\zeta_3 + \dot{\alpha}_2$, where k_3 is a positive constant. Then, $\dot{V}_3 = \dot{V}_2 - k_3\zeta_3^2$. Let ζ_4 be the new output:

$$
\zeta_4 = g\theta_2 - \alpha_3 \tag{56}
$$

The new subsystem to be stabilized is written as

$$
\dot{x}_1 = \zeta_2 + \alpha_1 \tag{57}
$$
$$
\dot{\zeta}_2 = g\theta_1 - \dot{\alpha}_1 \tag{58}
$$
$$
\dot{\zeta}_3 = g\theta_2 - \dot{\alpha}_2 \tag{59}
$$
$$
\dot{\zeta}_4 = g\dot{\theta}_2 - \dot{\alpha}_3 \tag{60}
$$

and let us propose the Lyapunov candidate function $V_4 = V_3 + \frac{\zeta_4^2}{2}$, then

$$
\dot{V}_4 = \dot{V}_3 + \zeta_4\dot{\zeta}_4 = \dot{V}_3 + \zeta_4(g\tilde{\tau}_{B\theta} - \dot{\alpha}_3) \tag{61}
$$

Let us propose the backstepping control input $\tilde{\tau}_{B\theta}$ as

$$
\tilde{\tau}_{B\theta} = \frac{1}{g}(-k_4\zeta_4 + \dot{\alpha}_3) \tag{62}
$$

where k_4 is a positive constant. Then

$$
\dot{V}_4 = \dot{V}_3 - k_4\zeta_4^2 = -k_1x_1^2 - k_2\zeta_2^2 - k_3\zeta_3^2 - k_4\zeta_4^2 \tag{63}
$$

With the proposed backstepping control input $\tilde{\tau}_{B\theta}$ we have $\dot{V}_4 < 0$, and then the system (40)-(43) is globally asymptotically stable. In order to express $\tilde{\tau}_{B\theta}$ as a function of x_1, x_2, θ_1 and θ_2, we need to rewrite ζ_4 and $\dot{\alpha}_3$ as a function of such variables:

$$
\zeta_4 = g\theta_2 + (k_1 + k_2 + k_3)g\theta_1 + (k_1k_2 + k_1k_3 + k_2k_3)x_2
$$
$$
+ k_1k_2k_3x_1 \tag{64}
$$
$$
\dot{\alpha}_3 = -(k_1 + k_2 + k_3)g\theta_2 - (k_1k_2 + k_1k_3 + k_2k_3)g\theta_1
$$
$$
- k_1k_2k_3x_2 \tag{65}
$$

The final control input for the forward position and pitch angle is rewritten as

$$
\tilde{\tau}_{B\theta} = -\frac{\overline{k}_1}{g}x_1 - \frac{\overline{k}_2}{g}x_2 - \overline{k}_3\theta_1 - \overline{k}_4\theta_2 \tag{66}
$$

where

$$
\overline{k}_1 = k_1k_2k_3k_4 \tag{67}
$$
$$
\overline{k}_2 = k_1k_2k_3 + k_1k_2k_4 + k_1k_3k_4 + k_2k_3k_4 \tag{68}
$$
$$
\overline{k}_3 = k_1k_2 + k_1k_3 + k_1k_4 + k_2k_3 + k_2k_4 + k_3k_4 \tag{69}
$$
$$
\overline{k}_4 = k_1 + k_2 + k_3 + k_4 \tag{70}
$$

2) Control of lateral position and roll angle: Rewrite the subsystem given by (37)-(38) as $\dot{y}_1 = y_2$, $\dot{y}_2 = -g\phi_1$, $\dot{\phi}_1 = \phi_2$ and $\dot{\phi}_2 = \tilde{\tau}_{B\phi}$. Using a similar procedure to the one proposed for the pitch control, the backstepping roll control input can be obtained as

$$\tilde{\tau}_{B\phi} = \frac{\overline{k}_1}{g}y_1 + \frac{\overline{k}_2}{g}y_2 - \overline{k}_3\phi_1 - \overline{k}_4\phi_2 \quad (71)$$

D. Sliding Modes Control

Consider the system (25) of integrators in cascade, and rewrite this system in the form [19]:

$$\begin{bmatrix} \dot{\varphi} \\ \dot{\varepsilon} \end{bmatrix} = \begin{bmatrix} f_a(\varphi, \varepsilon) \\ f_b(\varphi, \varepsilon) + G(x)E(x)u + \delta(t, x, u) \end{bmatrix} (72)$$

then, we have that $\varphi = [x_1 \ x_2 \ x_3]^T$ and $\varepsilon = x_4$. Also $f_a(\varphi, \varepsilon) = \begin{bmatrix} \alpha x_2 & \beta x_3 & \gamma x_4 \end{bmatrix}^T$ and $f_b(\varphi, \varepsilon) = 0$, $G(x) = E(x) = 1$, $\delta(t, x, u) = 0$. Consider $x_4 = -c_0 x_1 - c_1 x_2 - c_2 x_3 = \phi(\varphi)$ in order to stabilize the origin. The partial derivative of ϕ, with respect to φ, is given by

$$\frac{\partial \phi}{\partial \varphi} = [-c_0 \ -c_1 \ -c_2] \quad (73)$$

then

$$\frac{\partial \phi}{\partial \varphi} f_a = -c_0 \alpha x_2 - c_1 \beta x_3 - c_2 \gamma x_4 \quad (74)$$

Define a sliding surface

$$s = \varepsilon - \phi(\varphi) = x_4 + c_0 x_1 + c_1 x_2 + c_2 x_3 = 0 \quad (75)$$

whose time derivative is given by

$$\dot{s} = -\frac{\partial \phi}{\partial \varphi} f_a(\varphi, \varepsilon) + u \quad (76)$$

then, the control input can be expressed as

$$u = -c_0 \alpha x_2 - c_1 \beta x_3 - c_2 \gamma x_4 + v \quad (77)$$

where $\dot{s} = v$ and $v = -\nu(x)\tanh\left(\frac{s}{\varepsilon}\right)$. By definition, we have that

$$\dot{s} = g(x)v + \Delta(t, x, u) \quad (78)$$

thus $g(x) = 1$ and $\Delta(t, x, u) = 0$. Also, we need to satisfy

$$\left| \frac{\Delta(t, x, u)}{g(x)} \right| \leq \rho(x) + k_0 \|v\| \quad (79)$$

then

$$\left| \frac{0}{1} \right| \leq B \quad (80)$$

where B is a positive constant, thus $\rho(x) = B$ and $k_0 = 0$. We have then

$$\nu(x) \geq \frac{\rho(x)}{1 - k_0} \geq B \quad (81)$$

Now we can determine that $v = -\nu(x)\tanh\left(\frac{s}{\varepsilon}\right)$ or $v = -B\tanh\left(\frac{s}{\varepsilon}\right)$. Note that $\tanh(s/\varepsilon)$ is a smooth approximation of the function $\text{sign}(s)$, which is used in order to reduce the chattering effect. ε is selected as a small constant. Finally, it

TABLE I: Characteristics of the quad-rotor

Parameter	Value
Diameter between rotors	40 cm
Weight	800 grams
Autonomy	15 minutes
Power	12 V, 2200 mAh Li-Po battery
Motor	1200 Brushless

is now possible to completely write the input signal equation as

$$u = -c_0 \alpha x_2 - c_1 \beta x_3 - c_2 \gamma x_4 - B\tanh\left(\frac{s}{\varepsilon}\right) \quad (82)$$

For the forward position and pitch angle subsystem equations, given by (34)-(35), and by analogy with the system (25), we obtain that $\alpha = 1$, $\beta = g$ and $\gamma = 1$, also, we have that $x_1 = x$, $x_2 = \dot{x}$, $x_3 = \theta$ and $x_4 = \dot{\theta}$. With this information, the sliding mode surface is given by

$$s = c_0 x + c_1 g\dot{x} + c_2 \theta + \dot{\theta} \quad (83)$$

and the forward position and pitch angle control input can be expressed as

$$\tilde{\tau}_\theta = -c_0 \dot{x} - c_1 \theta - c_2 \dot{\theta} - B\tanh\left(\frac{s}{\varepsilon}\right) \quad (84)$$

The constant terms c_i's should be carefully selected to obtain a stable output. In order to stabilize x in a position outside of the origin, we must place $x_1 = e_x$. A similar procedure should be followed to generate the lateral position and roll angle control input.
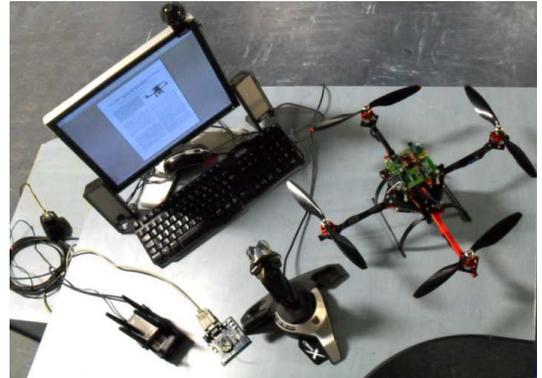
V. QUAD-ROTOR SYSTEM

Fig. 3: The quad-rotor experimental platform.

The vision algorithm and the controllers presented previously have been tested over a system composed by a four-rotor helicopter, a supervisory ground station, and a wireless video and data link. The whole system is shown in Figure3.

A. Aerial vehicle

The quad-rotor helicopter shown in Figure 1 was built using a group of off-the-shelf components. Some of its characteristics are resumed in Table I. Onboard electronics consist of two interconnected cards: the first board is the control unit, while

the second one deals with the motor's speed controllers. Both cards are shown in Figure 4. The control unit card performs the essential tasks of sensing, communicating and computing the control law for stabilizing the UAV attitude during fly. The characteristics of this board can be summarized as follows.

- a Texas Instruments® **TMS320F2812 DSP** module reads the signals of the embedded sensors and computes the control law for stabilizing the aircraft. Its working frequency is 500 Hz.
- an **MIDG II INS/GPS IMU** from Microbotics Inc® measures the angular position of the rotorcraft at a frequency of 100 Hz.
- three **ADXRS150 analog gyroscopes** measure the angular rates at 500 Hz. We chosen analog rate measurement rather than IMU based measurements, since we can obtain a faster refresh of angular rates which, enables a better attitude stabilization of the UAV.
- **battery voltage measurement circuit** intended to provide the actual tension level of the supply battery. This information is used for several goals: perform a safety landing and turn-off before an unwanted discharge of tension (avoiding accidents).
- an XBee ZB ZigBee PRO® **Radio Modem** is used to link the ground station and the aerial vehicle. This communication link can be used to introduce external control inputs, send the sensors information to the ground station, etc.

The second board contains:

- Signal conditioning circuitry: In this stage, the motor's control signals are decoupled from the rest of the electronic systems. PWM signals are also filtered and conditioned.

B. Supervisory ground station

The supervosory ground station consists of a desktop PC, a flight simulator Cyborg-X® joystick, an XBee ZB ZigBee PRO Radio Modem and a Diversity video receiver system. This ground station runs a supervisory control application allowing an user to send information to the helicopter and to chose between a manual control or an autonomous vision-based position hold. The supervisory ground station receives and saves data sent by the vehicle in order to debug and analyze the flight experiments. The control feedback between the supervisory ground station and the helicopter is performed at 30 Hz.

C. Vision system

The UAV vision system is shown in Figure 5. Real-time video is obtained by means of a high definition CTDM-5351® camera, with a resolution of 640×480 pixels. It is installed in the lower part of the helicopter and is placed pointing downwards. The camera is connected to a 200mW micro video and audio HF transmitter. Images from the vision system are recovered on ground by a 4-Antenna Diversity® System Receiver. This receiver is connected to the supervisory ground station PC throughout a USB frame grabber. The frequency of the video transmission is performed at a rate of 30 Hz.
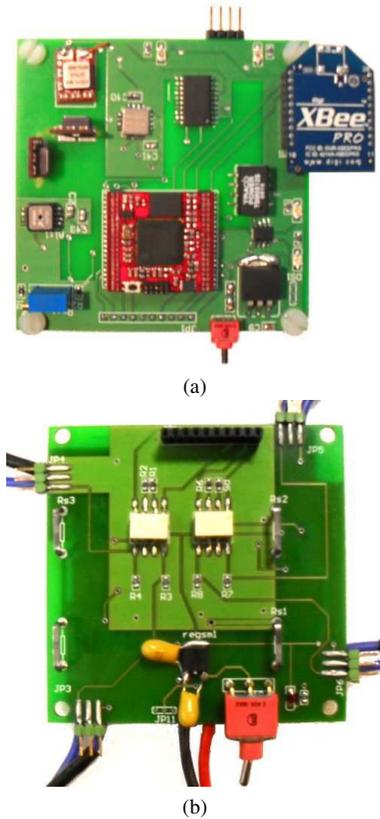

(a)

(b)

Fig. 4: The electronics on board: (a) Electronic card for the DSP, analog rate gyros, IMU connections and wireless modem; (b) Signal conditioning board.

Once received, real-time video is processed by a computer vision application developed specifically for computing the 3-dimensional position and the translational speed of the helicopter (see Section III). The computer vision application is programmed in Visual C++ and is based on OpenCV functions [26].

VI. EXPERIMENTAL RESULTS

The purpose of our experiments consists on validating the most appropriate control strategy for stabilizing the position of a quad-rotor UAV in hover flight. The quad-rotor is equipped with a vision system for estimating its relative 3-dimensional position (x, y, z) as well as its translational velocity $(\dot{x}, \dot{y}, \dot{z})$, with respect to a landing pad on the ground. Three similar experiments were performed, which are explained next.

Once the UAV is located exactly on top of the landing pad, the operator uses the supervisory ground station to define the current vehicle's position as the desired (x_d, y_d) position reference. The desired altitude z_d is always fixed at 150cm, and the desired yaw angle ψ is fixed at 0 degrees. The parameter values used for the altitude and the yaw controller are: $k_{pz} = 0.68$, $k_{vz} = 1.6$, $k_{p\psi} = 38$, $k_{p\psi} = 1350$. The parameters used for the nested saturations controller are: $\sigma_{b_4} = 0.4700$, $\sigma_{b_3} = 0.2349$, $\sigma_{b_2} = 0.1174$ and $\sigma_{b_1} = 0.0287$. The parameters used for the backstepping controller are: $\overline{k}_1 = 0.002$, $\overline{k}_2 = 0.004$, $\overline{k}_3 = 9$ and $\overline{k}_4 = 4$. The parameters
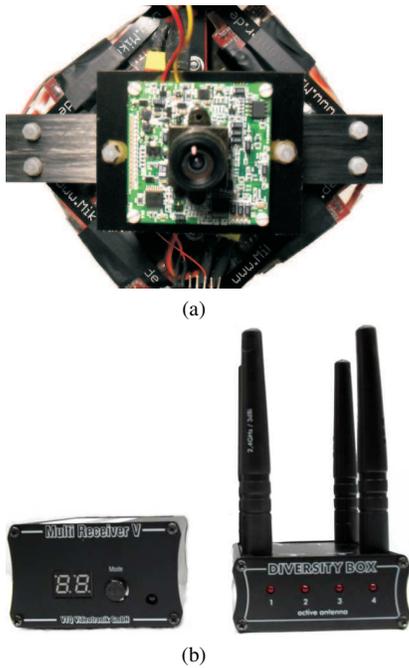
Fig. 5: The UAV vision system: (a) CTDM-5351 camera on board; (b) 4-Antenna Diversity System Receiver.

used for sliding mode controller are: $c_0 = 0.3$, $c_1 = 0.25$, $c_2 = 0.15$, $B = 0.011$ and $\epsilon = 0.05$. Those control parameters were found by trial and error.

Figures 6-9 show the obtained behavior when applying the three controllers to the UAV. We can observe that all the controllers achieve hovering flight, however, smoother translational and angular behaviors are obtained when using the nested saturation controller. Table II and Table III show the mean and standard deviation values for the position and Euler angles signals respectively. Note that Table II and Table III were computed with only one experiment for each controller, considering that the UAV is in steady state response. We have also computed the Mean Square Errors for the Euler angles. The values are show in Table IV. This results shown that the Nested Saturations controller is the method that induces less angular corrections, which can be considered as less control inputs generated during flight, and consequently, less energy consumption.

Note in Table II and Table III that the values for the backstepping controller are closer to the desired reference values. However, if the important objective concerns the energy consumption, the nested saturations controller should be considered as the best option. A video of the experiments can be seen at

http://www.youtube.com/watch?v=SQlSXruTnj0

VII. CONCLUSIONS

Three control strategies were implemented and tested in a real-time application for a quad-rotor UAV by using visual feedback. The helicopter uses a combination of an IMU and three analog rate gyros to measure its angular dynamics. Using an homography estimation technique, an onboard monocular

vision system was used to estimate the (x, y, z) 3-dimensional position of the aerial vehicle with respect to a landing pad. It was also implemented an optical flow technique for estimating the vehicle's $(\dot{x}, \dot{y}, \dot{z})$ translational velocities. The control algorithms were implemented onboard to stabilize the UAV's 3-dimensional position and attitude. All control algorithms ensure that the Euler angles of the vehicle remain very close to the desired values. The experimental results show that the nested saturations control approach is the most appropriated strategy for our system, since it ensures a smoother vehicle behavior and reduces the energy consumption with respect to the other two controllers.
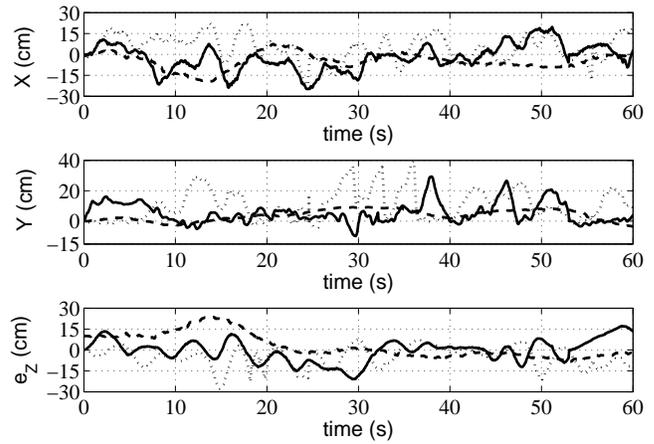

Fig. 6: Behavior of the X, Y and altitude error signals. The Backstepping, Nested Saturation and Sliding Modes are represented by solid line, dashed line and dotted line respectively.
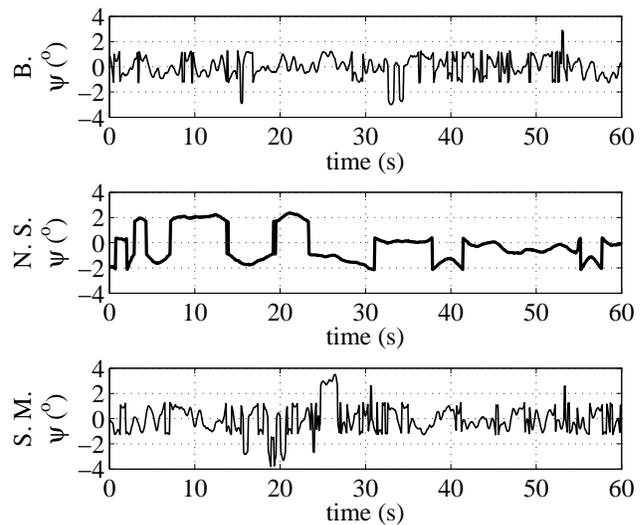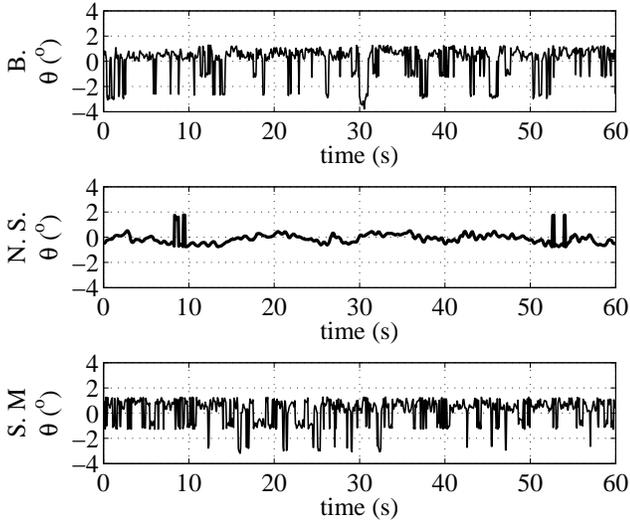

Fig. 7: Behavior of the yaw angle.
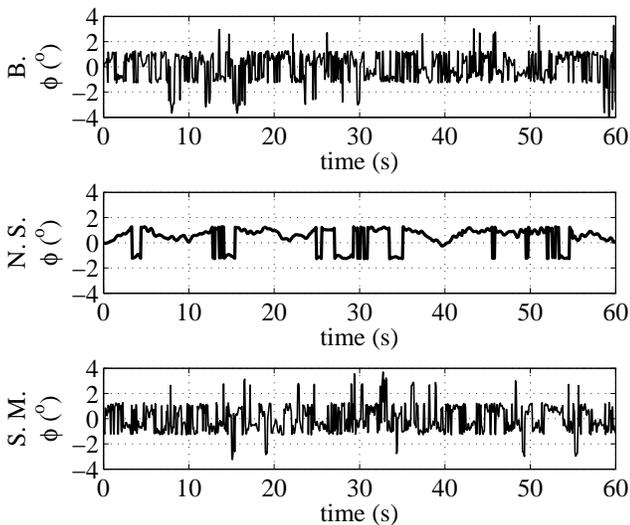
Fig. 8: Behavior of the pitch angle.

Fig. 9: Behavior of the roll angle.

ACKNOWLEDGEMENTS

REFERENCES

[1] J. R. Reinhardt, J. E. James and E. M. Flannagan. "Future Employment of UAVS: Issues of Jointness", *Joint Force Quarterly*, vol. 22, pp. 36-41, Summer 1999.

[2] S. Salazar, J. Escareno, D. Lara and R. Lozano, *Embedded control system for a four rotor UAV*, International Journal of Adaptive Control and Signal Processing, vol. 21, issue 2-3, pp. 189-204, March-April 2007.

[3] S. Bouabdallah and A. Noth, *PID vs LQ control techniques applied to an indoor micro quadrotor*, IEEE International Conference on Intelligent Robots and Systems (IROS), Sendai, Japan, October, 2004.

[4] S. Bouabdallah and R. Siegwart, *Backstepping and Sliding-mode Techniques Applied to an Indoor Micro Quadrotor*, Proceedings of IEEE International Conference on Robotics and Automation, Barcelona, Spain, April 2005.

[5] E. Altug, J. Ostrowski, anc C. Taylor, *Control of a Quadrotor Helicopter Using Dual CameraVisual Feedback*, The International Journal of Robotics Research, vol. 24, issue 5, pp.329-341, May 2005.

[6] S. Saripalli, J. M. Roberts, P. I. Corke and G. Sukhatme, *A Tale of Two Helicopters*, In IEEE/RSJ International Conference on Intelligent Robots and Systems, Las Vegas, USA, 2003.

[7] S. Salazar, H. Romero, R. Lozano and P. Castillo, *Modeling and Real-Time Stabilization of an Aircraft Having Eight Rotors*, Journal of Intelligent and Robotic Systems, vol. 54, issue 1-3, pp. 455 - 470, March 2009.

[8] E. Rondon, L. R. Garcia Carrillo, and I. Fantoni. *Vision-based altitude, position and speed regulation of a quad-rotor rotorcraft*, In International Conference on Intelligent Robots and Systems, Taipei, Taiwan. October 2010.

[9] S. Salazar, J. E. Gomez, J. A. Guerrero and R. Lozano, *Vision Based Autonomous Hover of a Mini-rotorcraft*, in International Conference on Unmanned Aerial Vehicles, Dubai, UAE, June 2010.

[10] J. E. Gomez, S. Salazar, J. A. Guerrero and R. Lozano, *Vision Based Tracking for a Mini-Rotorcraft Using Vanishing Points*, 18th IFAC World Congress, Milan, Italy, August 2011.

[11] S. Saripalli, J. Montgomery and G. Sukhatme, *Visually-Guided Landing of an Unmanned Aerial Vehicle*, IEEE Transactions on Robotics and Automation, vol. 19, no. 3, pp. 371-381, Jun 2003.

[12] H. Romero, S. Salazar, R. Lozano, *Real-Time Stabilization of an Eight-Rotor UAV Using Optical Flow*, In IEEE Transactions on Robotics, vol. 25, issue 4, pp.809-817, August 2009.

[13] E. Rondon, S. Salazar, J. Escareno and R. Lozano, *Vision-Based Position Control of a Two-rotor VTOL miniUAV*, Journal of Intelligent and Robotic Systems, vol. 57, no. 1-4, pp. 49-64, DOI: 10.1007/s10846-009-9370-6, 2009.

[14] M. Achtelik, A. Bachrach, R. He, S. Prentice and N. Roy. *Stereo vision and laser odometry for autonomous helicopters in GPS-denied indoor environments*, in Unmanned Systems Technology XI, Proc. of SPIE, vol. 7332, April 2009.

[15] L. R. Garcia Carrillo, E. Rondon, A. Sanchez, A. Dzul and R. Lozano. *Stabilization and trajectory tracking of a quad-rotor UAV using vision*, Journal of Intelligent and Robotic Systems, vol. 61, issue 1-4, pp. 103-118, 2011.

[16] P. Castillo, A. Dzul and R. Lozano, *Real-Time Stabilization and Tracking*

TABLE II: Mean values of position and Euler angles.

Parameter	Backstepping	Nested saturations	Sliding modes
Yaw angle	$-0.0108°$	$-0.2504°$	$-0.0400°$
Pitch angle	$0.0903°$	$-0.0706°$	$0.2033°$
Roll angle	$-0.0315°$	$0.4371°$	$-0.0782°$
e_Z error	0.0176 cm	1.4550 cm	-6.8905 cm
X position	-1.5662 cm	-2.7342 cm	4.3211 cm
Y position	5.6848 cm	4.2451 cm	10.1918 cm

TABLE III: Standard deviation of position and Euler angles.

Parameter	Backstepping	Nested saturations	Sliding modes
Yaw angle	$0.7968°$	$1.2606°$	$1.0260°$
Pitch angle	$1.1588°$	$0.4473°$	$0.9833°$
Roll angle	$1.1673°$	$0.7635°$	$1.0739°$
e_Z error	8.0868 cm	8.2162 cm	7.7071 cm
X position	9.2075 cm	6.2887 cm	10.2763 cm
Y position	7.1207 cm	4.1411 cm	9.2603 cm

TABLE IV: Mean square error for Euler angles.

Parameter	Backstepping	Nested saturations	Sliding modes
Yaw angle	$0.6350°$	$1.5910°$	$1.0853°$
Pitch	$1.3519°$	$0.1476°$	$1.0144°$
Roll	$1.3499°$	$0.7623°$	$1.1343°$

of a Four-Rotor Mini Rotorcraft, IEEE Transactions on Control Systems Technology, vol. 12, no. 4, pp. 510 - 516, July 2004.

[17] A.R. Teel, *Global stabilization and restricted tracking for multiple integrators with bounded controls*, Systems and control letters, vol. 18, issue 3, pp. 165-171, 1992.

[18] A. Saberi, P.V. Kokotovic, and H.J. Sussmann, *Global Stabilization of Partially Linear Composite Systems*, SIAM Journal on Control and Optimization, vol. 28, issue 6, pp. 1491-1503, November 1990.

[19] H. K. Khalil, *Nonlinear Systems*, Prentice-Hall, third edition, 2002.

[20] B. Etkin, and L.D Reid, *Dynamics of Flight*, Wiley, New York, 1959, ISBN 0471034185.

[21] G. Bradski and A. Kaehler, *Learning OpenCV, Computer Vision with the OpenCV Library*, O'Reilly Media, 1st edition, ISBN 9780596516130, 2008.

[22] R. Horaud, B. Conio, O. Leboulleux and B. Lacolle, *An Analytic Solution for the Perspective 4-Point Problem*, Computer Vision, Graphics and Image Processing, Vol. 47, pp. 33–44, 1989.

[23] J. Y. Bouguet, *Camera Calibration Toolbox for Matlab*, http://www.vision.caltech.edu/bouguetj/calib_doc/ Accessed December 2010.

[24] J. Y. Bouguet, *Pyramidal Implementation of the Lucas Kanade Feature Tracker - Description of the algorithm*, Intel Corporation - Microprocessor Research Labs, 2002.

[25] E. Rondon, I. Fantoni-Coichot, A. Sanchez, and G. Sanahuja, *Optical flow-based controller for reactive and relative navigation dedicated to a four rotor rotorcraft*, in IEEE/RSJ International Conference on Intelligent Robots and Systems, pp. 684-689, St. Louis, USA, October 2009.

[26] *Open Computer Vision Lib*, http://sourceforge.net/projects/opencvlibrary/