2008 IEEE/RSJ International Conference on Intelligent Robots and Systems
Acropolis Convention Center
Nice, France, Sept, 22-26, 2008

Hovering flight and vertical landing control of a VTOL Unmanned Aerial Vehicle using Optical Flow

Bruno Herisse, Francois-Xavier Russotto, Tarek Hamel, Robert Mahony

Abstract— **This paper presents a nonlinear controller for hovering flight and touchdown control for a Vertical Take-off and Landing (VTOL) Unmanned Aerial Vehicle (UAV) using inertial optical flow. The VTOL vehicle is assumed to be a rigid body, equipped with a minimum sensor suite (camera and IMU), manoeuvring over a textured flat target plane. Two different tasks are considered in this paper: the first concerns the stability of hovering flight and the second one concerns regulation of automatic landing using the divergent optical flow as feedback information. Experimental results on a quad-rotor UAV demonstrate the performance of the proposed control strategy.**

I. INTRODUCTION

Recent advances in technology has lead to a growing interest in aerial robotic applications where, historically, payload constraints have severely limited autonomy of a range of micro aerial vehicles. The small size, highly coupled dynamics and 'low cost' implementation of such systems provide an ideal testing ground for sophisticated non-linear control techniques. A key issue arising is the difficulty of navigation through cluttered environments and close to obstructions (obstacle avoidance, take-off and landing). A vision system is a cheap, light, passive and adaptable sensor that can be used along with an Inertial Measurement Unit (IMU) to provide robust relative pose information and more generally allows autonomous navigation. Using a camera as the primary sensor for relative position leads to a visual servo control problem, a field that has been extensively developed over the last few years [16], [6], [1], [15], [9]. An alternate approach for the motion autonomy uses insight from the behavior of flying insects and animals to develop control strategies for aerial robots, in particular, techniques related to visual flow [17]. When a honeybee lands, neither velocity nor distance to the ground are used, the essential information needed is the time-to-contact that can be obtained from the optical flow field divergence [17]. This property has already been used for obstacle avoidance in mobile robotics [19], [11]. Recently control of flying vehicles have been inspired from models of flying insects [4], [13], [2], [19], [11]. It is rare that vehicle dynamics is mentioned in the theoretical developments or even in the experimental discussion of these prior works. The flight regime of insects is highly damped due to their high drag to mass ratios and the control strategies that have been observed in the various biological studies do not generalise to high-inertia, low-drag aerial vehicles.

In this paper, a control law for hovering flight and landing manoeuvre of a UAV, capable of quasi stationary flight, is considered by focusing just on the translational dynamics of a system. A 'high gain' controller can be designed to stabilise the orientation dynamics. This approach is classically known in aeronautics as guidance and control (or hierarchical control) [14]. The image feature considered is the translational optical flow obtained from the measurement of the optical flow of a textured target plane in the inertial frame using additional information provided by an embedded IMU. A PI-type controller is designed for hovering flight while a nonlinear controller, using the optical flow divergence as feedback information, is proposed for vertical landing. Lyapunov analysis is used to prove semi-global exponential stability of the closed-loop system for landing manoeuvres. Experimental results are obtained on a quad-rotor UAV capable of quasi-stationary flight developed at CEA (French Atomic Energy Commission). The proposed closed-loop control schemes demonstrate efficiency and performance for the hovering flight and vertical landing manoeuvre.

The body of the paper consists of six sections followed by a conclusion. Section II presents the fundamental equations of motion for an X4-flyer UAV. In Section III, fundamental equations of optical flow are presented. Sections IV and V present the proposed control strategies for hovering and vertical landing manoeuver respectively. Section VI describes the experimental results obtained on the quad-rotor vehicle.

II. UAV DYNAMIC MODEL AND TIME SCALE SEPARATION

The VTOL UAV is represented by a rigid body of mass m and of tensor of inertia $\mathbf{I}$. To describe the motion of the UAV, two reference frames are introduced: an inertial reference frame $\mathcal{I}$ associated with the vector basis $[e_1, e_2, e_3]$ and a body-fixed frame $\mathcal{B}$ attached to the UAV at the center of mass and associated with the vector basis $[e_1^b, e_2^b, e_3^b]$. The position and the linear velocity of the UAV in $\mathcal{I}$ are respectively denoted $\xi = (x, y, z)^T$ and $v = (\dot{x}, \dot{y}, \dot{z})^T$. The orientation of the UAV is given by the orientation matrix $R \in SO(3)$ from $\mathcal{B}$ to $\mathcal{I}$, usually parameterized by Euler's angles ψ, θ, ϕ (yaw, pitch, roll). Finally, let $\Omega = (\Omega_1, \Omega_2, \Omega_3)^T$ be the angular velocity of the UAV defined in $\mathcal{B}$.

A translational force F and a control torque Γ are applied to the UAV. The translational force F combines thrust, lift, drag and gravity components. For a miniature VTOL UAV in quasi-stationary flight one can reasonably assume that the aerodynamic forces are always in direction e_3^b, since the

B. Herisse and F.-X. Russotto are with CEA/List, Fontenay-Aux-Roses, France firstname.lastname@cea.fr

T. Hamel is with I3S, UNSA - CNRS, Sophia Antipolis, France thamel@i3s.unice.fr

R. Mahony is with Dep. of Eng., Australian Nat. Univ., ACT, 0200 Australia Robert.Mahony@anu.edu.au

978-1-4244-2058-2/08/$25.00 ©2008 IEEE.

lift force predominates on the other components [12]. The gravitational force can be separated from other forces and the dynamics of the VTOL UAV can be written as:

$$\dot{\xi} = v \tag{1}$$

$$m\dot{v} = -TRe_3 + mge_3 + \Delta \tag{2}$$

$$\epsilon\dot{R} = R\Omega_\times, \tag{3}$$

$$\epsilon\mathbf{I}\dot{\Omega} = -\Omega \times \mathbf{I}\Omega + \Gamma. \tag{4}$$

In the above notation, g is the acceleration due to gravity, and T a scalar input termed the thrust or heave, applied in direction e_3, the third-axis unit vector. The term Δ represents constant (or slowly time varying unmodeled) forces. The matrix $\Omega_\times$ denotes the skew-symmetric matrix associated to the vector product $\Omega_\times x := \Omega \times x$ for any x.

The positive parameter $\epsilon > 0$ ($\epsilon << 1$) is introduced for timescale separation between the translation and orientation dynamics. It means that the orientation dynamics of the VTOL UAV are compensated with separate high gain control loop. For this hierarchical control, the time-scale separation between the translational dynamics (slow time-scale) and the orientation dynamics (fast-time scale) can be used to design position and orientation controllers under simplifying assumptions. Although reduced-order subsystems can hence be considered for control design, the stability must be analyzed by considering the complete closed-loop system [14]. In this paper, however, we will focus on the control design for the translational dynamics.

III. OPTICAL FLOW EQUATIONS

In this section image plane kinematics and spherical optical flow are derived. The camera is assumed to be attached to the center of mass so that the camera frame coincides with the body-fixed frame.

A. Kinematics of an image point under spherical projection

Computation of the optical flow is one of the most important problems in computer vision, intensively studied by researchers and different nonlinear complex methods have been proposed. In order to simplify computation, we adopt a spherical image screen and arrange equations in linear form. It is shown in [18] that optical flow equations can be numerically computed from an image plane to a spherical retina. A Jacobian matrix relates temporal derivatives (velocities) in the spherical coordinate system to those in the image frame. Let $\mathcal{P}$ and $\mathcal{S}$ denotes the planar image frame and the spherical surface respectively.

Define $P = (X, Y, Z) \in \Re^3$ as a stationary visible target point expressed in the camera frame. The image point observed by the camera is denoted $p_\mathcal{P}$ (respectively $p_\mathcal{S}$) and is the projection of P onto the image surface $\mathcal{P}$ (respectively $\mathcal{S}$) of the camera. Thus,

$$p = \frac{P}{\sigma(P)} \tag{5}$$

Where the function σ represents a positive rescaling factor of any vector $x \in \Re^3$. For a camera with a flat image plane then is given by the perspective projection (fig. 1) $\sigma(P) = \frac{|Z|}{f}$ and for a virtual spherical camera (fig. 2) $\sigma(P) = \frac{|P|}{f}$ where f is the focal length. The time derivative $\dot{p}_\mathcal{P}$ and $\dot{p}_\mathcal{S}$ are respectively the dynamics of the image point, also called optical flow equations, on the flat image plane and the spherical one. The Jacobian matrix relating temporal derivatives in $\mathcal{S}$ to those in $\mathcal{P}$ is given by the following equation:

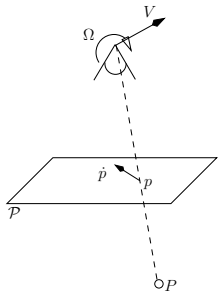

Fig. 1: Image dynamics for rectilinear camera image geometry

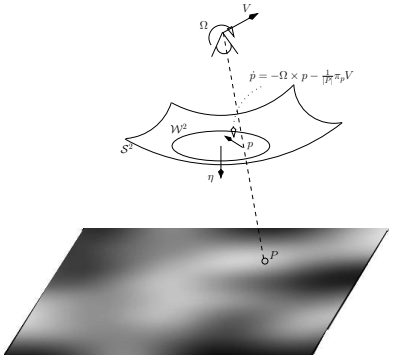

Fig. 2: Image dynamics for spherical camera image geometry

$$J = \frac{\partial p_\mathcal{S}}{\partial p_\mathcal{P}} \tag{6}$$

$$\dot{p}_\mathcal{S} = J\dot{p}_\mathcal{P} \tag{7}$$

Motivated by the preceding discussion, we make the following assumptions.

Assumption 3.1: The image surface of the camera is spherical with unit image radius (f=1).

Assumption 3.2: The target points are stationary in the inertial frame. Thus, motion of target points depend only on motion of the camera.

The dynamics of an image point for a spherical camera of image surface radius unity are (see [6])

$$\dot{p} = -\Omega \times p - \frac{\pi_p}{|P|}V, \tag{8}$$

Where $\pi_p = (I_3 - pp^T)$ is the projection $\pi_p : \Re^3 \to T_p\mathcal{S}^2$, the tangent space of the sphere $\mathcal{S}^2$ at the point $p \in \mathcal{S}^2$ ($p = p_\mathcal{S}$). The vector V represents the translational velocity

of the center of mass expressed in the body-fixed frame ($V = R^T v$).

Let $\eta' \in \mathcal{I}$ denote the unit normal to a target plane (see [9]) and $\eta = R^T \eta'$ its expression in body-fixed frame $\mathcal{I}$. Define $d := d(t)$, to be the orthogonal distance from the target surface to the origin of frame $\mathcal{B}$, measured as a positive scalar. Thus, for any point P on the target surface

$$d(t) = \langle P, \eta \rangle$$

where both P and η are expressed in the body-fixed frame. One can also write $d(t) = -\langle \eta', \xi \rangle$ where ξ is the position of the camera. For a target point, one has

$$|P| = \frac{d(t)}{\langle p, \eta \rangle} = \frac{d(t)}{\cos(\theta)}$$

where θ is the angle between the inertial direction η and the observed target point p. Substituting this relation into (8) yields

$$\dot{p} = -\Omega_\times p - \frac{\cos(\alpha)}{d(t)} \pi_p V \tag{9}$$

B. Translational optical flow computation

Measuring the optical flow is the key aspect of the practical implementation of the control algorithms proposed in the sequel. The optical flow $\dot{p}$ can be computed using a range of algorithms (correlation-based technique, features-based approaches, differential techniques, etc) [7]. Note that due to the rotational ego-motion of the camera, (9) involves the angular velocity as well as the linear velocity. For the control problem we define an inertial translational optical flow from the integral of all observed optical flow corrected for rotational angular velocity. When the observed world is a flat planar surface, inertial translational optical flow will have three components, flow in the two planar directions, analogous to classical optical flow, and flow in the normal direction to the plane, analogous to optical divergence.

Assume that the target plane is textured and the available data are $\dot{p}$, η and Ω where η and Ω are estimated from the IMU data (see [10]). The translational optical flow is obtained by integrating the observed optical flow over a section $\mathcal{W}^2$ of the sphere around the pole normal to the target plane (fig. 2). The average of the optical flow along the window $\mathcal{W}^2$ is given by ([9]):

$$\phi = \iint_{\mathcal{W}^2} \dot{p} = -\beta \Omega \times \eta - \frac{QV}{d}, \tag{10}$$

where the parameter β and the matrix Q depend on the size of the window $\mathcal{W}^2$. It can be verified that β represents the angle of the field of view of the window $\mathcal{W}^2$ and $Q = R^T(R_t \Lambda R_t^T)R$ is a symmetric positive definite matrix. The matrix Λ is a constant diagonal matrix depending on the window $\mathcal{W}^2$ parameters and R_t represents the orientation matrix of the target plane with respect to the inertial frame. For instance, if $\mathcal{W}^2$ is the hemisphere centered at η, corresponding to the visual image of the infinite target plane, it can be shown that

$$\beta = \pi, \quad \Lambda = \frac{\pi}{4} \begin{pmatrix} 3 & 0 & 0 \\ 0 & 3 & 0 \\ 0 & 0 & 2 \end{pmatrix} \tag{11}$$

From (10) it is straightforward to obtain the translational optical flow ($w = \frac{v}{d}$).

$$w = -(R_t \Lambda^{-1} R_t^T) R(\phi + \beta \Omega \times \eta) = \frac{v}{d} \tag{12}$$

Knowing that $d(t) = -\langle \eta', \xi \rangle$, it is straightforward to show that the dynamics of $d(t)$, is given by the following equation:

$$\dot{d} = -\langle \eta', v \rangle = -d \langle \eta', w \rangle \tag{13}$$

Note that measurable variables are w and $\frac{d}{d_0}$ where d_0 represents the initial value of d. The term $\frac{d}{d_0}$ is obtained from the exponential integration over time of equation (13)

$$\frac{d(t)}{d_0} = \exp\left(\log \frac{d(t)}{d_0}\right) = \exp\left(-\int_0^t \langle \eta', w \rangle d\tau\right)$$

IV. Hovering flight

In this section a control design insuring hovering flight is proposed. The control problem considered is the stabilisation of the linear velocity about zero despite unmodeled constant (or slowly time varying) dynamics. A PI-type controller depending on measurable variables $(w, \frac{d}{d_0})$ is developed for the translational dynamics (2). The full vectorial term TRe_3 will be considered as control input for these dynamics. We will assign its desired value $u \equiv (TRe_3)^d = f(w, \frac{d}{d_0})$. Assuming that actuator dynamics can be neglected before the rigid body dynamics of the UAV, the value T_d is considered to be instantaneously reached by T. Therefore, we have $(TRe_3)^d = TR^d e_3$, where R^d is the desired orientation of the vehicle. This vector can be split into its magnitude, $T = ||f(w, \frac{d}{d_0})||$, representing the first control input, and its direction $R^d e_3 = \frac{f\left(w, \frac{d}{d_0}\right)}{T}$.

For the orientation dynamics of (3)-(4), a high gain controller is used to insure that the orientation R of the UAV converges to the desired orientation R^d. This common approach is used in practice and may be justified theoretically using singular perturbation theory. In the sequel we will focus only on the translational dynamics.

Proposition 4.1: Consider the dynamics (2) and assume that the thrust vector ($u \equiv TR^d e_3$) is the control input chosen as follows:

$$u = k_P \frac{d}{d_0} w + k_I \int_0^t \left(\frac{d}{d_0} w\right) d\tau + mge_3, \quad k_P, k_I > 0 \tag{14}$$

Then, the linear velocity converges asymptotically towards zero.

Proof: Simple computation shows that the controller (14) is a PI controller that may be written as follows:

$$TR^d e_3 = \frac{1}{d_0} \left(k_P v + k_I \int_0^t v d\tau\right) + mge_3$$

Since k_P, k_I and d_0 are positive parameters, exponential stability of v to zero is insured. ■

Note that, if the orientation of the target plane with respect to the inertial frame $R_t = I_3$, the normal direction to the target plane becomes e_3 (observed from the camera-frame as a vector pointing towards the plane). Moreover, if one assumes that the target plane is in the plane x-y of the inertial frame, the variable d becomes the height h (or $|z|$) from the camera to the target and therefore the dynamics of the closed-loop system are:

$$\ddot{\xi} = -\frac{k_1}{h_0}\dot{\xi} - \frac{k_2}{h_0}(\xi - \xi_0) + \Delta \tag{15}$$

From the classical theory of linear system one can insure that ξ converges exponentially to $\xi_0 + \frac{h_0}{k_2}\Delta$. Consequently, if $\Delta = 0$, the position of the system will stabilise at its initial position.

V. Touchdown control

We are interested in the case where the aerial robot comes in to land on a horizontal plane. Thus, we assume that the target plane belongs to the plan x-y of the inertial frame so that $d \equiv |z| \equiv h$. Two nonlinear controllers insuring vertical landing are proposed. Define

$$w^d = (0, 0, \omega^*)^T, \;\; \omega^* > 0,$$

as the desired translational optical flow. Note that the third component of the inertial translational optical flow acts analogously to optical flow divergence. It is straightforward to show that when $w = w^d$ one has $(v_x, v_y) = (0, 0)$ and $v_z = h_0\omega^* \exp(-\omega^* t)$ which converges to zero and $h = h_0 \exp(-\omega^* t)$ insuring a smooth vertical landing. Therefore, previous control law (14) may be used to stabilise $(v_x, v_y) = (0, 0)$. We still need to provide the control scheme for the remaining degree of freedom ($h \equiv |z|$). In particular, we consider the desired set point, ω^*, for the flow divergence (the flow in the normal direction to the target plane) and look for a control law that achieves regulation of $\dot{h}/h + \omega^*$. Two controllers are proposed. The first one is a direct application of the controller proposed in [3], along with a more concise proof.

Proposition 5.1: Consider the dynamics of the third component of (2) and assume that the third component of the thrust vector u_z of u ($u_z \equiv e_3^T T R e_3$) is the control input. Choose u_z as follows:

$$u_z = mk(w_z - \omega^*) + mg \tag{16}$$

Define $\delta = \dot{h} + \alpha h$ and the following Lyapunov function candidate $\mathcal{L}$ by

$$\mathcal{L} = \frac{1}{2}h^2 + \frac{1}{2\alpha^2}\delta^2 \tag{17}$$

where $\alpha = (\omega^* + \frac{\Delta_z}{mk})$. Choose the control gain k such that:

$$k > max\left(\frac{|\Delta_z^{\max}|}{m\omega^*}, \frac{\sqrt{2\mathcal{L}(0)}\omega^*}{2}(1+\sqrt{1 + \frac{4|\Delta_z^{\max}|}{m\sqrt{2\mathcal{L}(0)}\omega^{*2}}})\right) \tag{18}$$

Then for all initial conditions such that $h_0 > 0$ ($h_0 \equiv |z_0|$), the closed-loop trajectory exists for all time and $h(t) > 0$ remains positive and converges exponentially to zero.

Proof: Since the dynamics of the considered system are decoupled, recall the dynamics of the third component of (2):

$$m\dot{v}_z = -u_z + mg + \Delta_z \tag{19}$$

The control law is well defined and smooth for $h > 0$. First, show that $h(t)$ remains positive for all time. Knowing that $v_z = -\dot{h}$ and $w_z = -\frac{\dot{h}}{h}$, integrate equation (19), it yields:

$$h \exp\left(\frac{1}{k}\left(\dot{h} - \dot{h}_0\right)\right) = h_0 e^{-\alpha t} \tag{20}$$

This insures that $h(t)$ is positive for all time. It remains to show that $\dot{h}$ is bounded and that $h(t)$ converges exponentially to zero. Differentiating δ and recalling u_z (16), it yields:

$$\dot{\delta} = -\alpha^2 h - (\frac{k}{h} - \alpha)\delta \tag{21}$$

Differentiating $\mathcal{L}$ and recalling the above equation, (21), it yields

$$\dot{\mathcal{L}} = -\alpha h^2 - \frac{1}{\alpha^2}(\frac{k}{h} - \alpha)\delta^2 \tag{22}$$

Choosing k such that (18) is satisfied, it is straightforward to show that $\alpha > 0$ and that it exists a positive constant $\epsilon > 0$ such that:

$$\dot{\mathcal{L}} \leq -\alpha h^2 - \frac{1}{\alpha^2}\epsilon\delta^2 \tag{23}$$

This insures that $\mathcal{L}$ converges exponentially to zero and, therefore, $\dot{h}$ remains bounded for all time. Consequently, from (20), one can insure that h converges to zero with α as exponential decay constant. ■

The second controller is similar to the previous one augmented with an integrator. The integral correction is introduced to compensate for the unmodeled forces Δ_z and to insure that h converges to zero with ω^* as exponential decay constant. Thus, w_z converges to ω^*.

Proposition 5.2: Consider the dynamics of the third component of (2) and assume that the third component of the thrust vector u_z of u ($u_z \equiv e_3^T T R e_3$) chosen as follows:

$$u_z = k_P(w_z - \omega^*) + k_I \int_0^t (w_z - \omega^*)d\tau + mg \tag{24}$$

Then for all initial conditions such that $h_0 > 0$, the closed-loop trajectory exists for all time and $h(t)$ remains positive and converges to zero with ω^* as exponential decay constant. Moreover, contrary to the previous controller developed in this paper, w_z converges exactly towards ω^*.

The stability proof of the closed-loop system is much more difficult. It is omitted here for brevity, but may be obtained from the authors by request.

VI. EXPERIMENTAL RESULTS

In this section, experimental results of the above algorithms designed for the full dynamics of the system are presented. The UAV used for the experimentation is the quadrotor, made by the CEA, (Fig. 3) which is a vertical take off and landing vehicle ideally suited for stationary and quasi stationary flight [5].

A. Prototype description

The X4-flyer is equipped with a set of four electronic boards designed by the CEA. Each electronic board includes a micro-controller and has a particular function. The first board integrates motor controllers which regulate the rotation speed of the four propellers. The second board integrates an Inertial Measurement Unit (IMU) constituted of 3 low cost MEMS accelerometers, which give the gravity components in the body frame, 3 angular rate sensors and 2 magnetometers. On the third board, a Digital Signal Processing (DSP), running at 150 MIPS, is embedded and performs the control algorithm of the orientation dynamics and filtering computations. The final board provides a serial wireless communication between the operator's joystick and the vehicle. An embedded camera with a view angle of 70 degrees pointing directly down, transmits video to a ground station (PC) via a wireless analogical link of 2.4GHz. A Lithium-Polymer battery provides nearly 10 minutes of flight time. The loaded weight of the prototype is about 650g. The images sent by the embedded camera are received by the ground station at a frequency of 25Hz. In parallel, the X4-flyer sends the inertial data to the station on the ground at a frequency of 15Hz. The data is processed by the ground station PC and incorporated into the control algorithm. Desired orientation and desired thrust are generated on the ground station PC and sent to the drone. A key difficulty of the algorithm implementation lies in the relatively large time latency between the inertial data and visual features. For orientation dynamics, an embedded 'high gain' controller in the DSP running at 166Hz, independently ensures the exponential stability of the orientation towards the desired one.

B. Experiments

The considered target plane is the ground and textures are made of random contrasts (Fig. 3). A Pyramidal implementation of the Lucas-Kanade [8] algorithm is used to compute the optical flow. The efficiency of the algorithm is increased by defocusing the camera to low-pass filter images.

The field of view of the window of computation is $45°$ by $70°$ around η. Optical flow is computed at 504 points on this window. Thus, a discrete integration on the window provides the following estimate of the matrix Λ and the parameter β:

$$\beta = 0.878 \tag{25}$$

$$\Lambda = \begin{pmatrix} 0.839 & 0 & 0 \\ 0 & 0.781 & 0 \\ 0 & 0 & 0.134 \end{pmatrix} \tag{26}$$

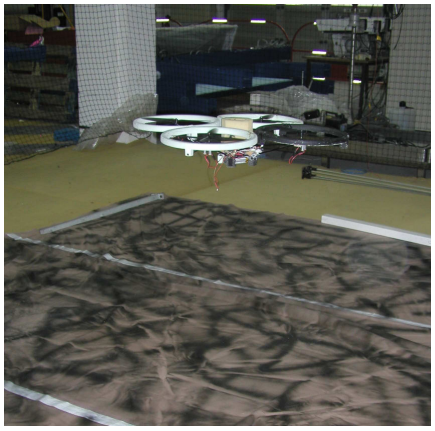

Fig. 3: hovering flight above a textured ground

Knowing that the divergent flow is relatively low compared to the translational flow in the front and back directions [4] and that only the divergent flow is used for landing manoeuver, the control approach is split into two phases. First the hovering flight is performed to insure that the translational flow is set to zero and then the vertical landing manoeuver is applied.

During the experiments, the yaw velocity is set to zero. It has no effect on the proposed control scheme. The drone is teleoperated near the target, so that textures are visible. Figure 4 shows the results for automatic Hovering flight. Estimation of the UAV relative position is computed from the optical flow as follows:

$$\frac{\xi}{h_0} = \int_0^t w\gamma(\tau)\, d\tau = \int_0^t w \exp\left(-\int_0^\tau w_z\, d\delta\right) d\tau \tag{27}$$

For the hovering flight, one can observe the effective convergence of the position with a satisfactory behaviour.

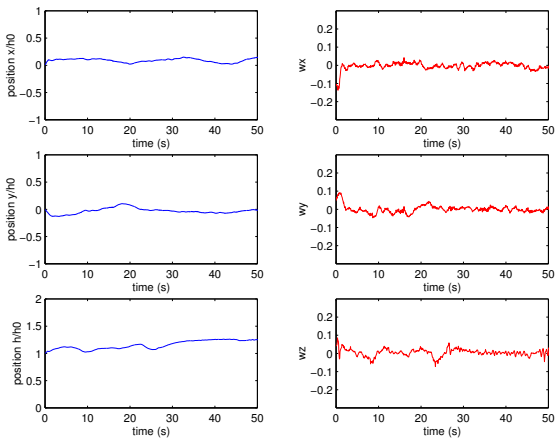

Fig. 4: Hovering flight

For the vertical landing, the desired set point w^d is set to $(0, 0, 0.04)^T$, it insures a relatively rapid descent

(approximatively 50s). Figures 5 and 6 show the exponential convergence of the position with a good behavior using the two controllers of the vertical landing manoeuvre. Note that, due to the landing gear, the final position is not $h \equiv 0$. Due to the fact that w_z converge to α, the first controller (16) produces a less rapid descent, it can be concluded that $\alpha < \omega^*$.

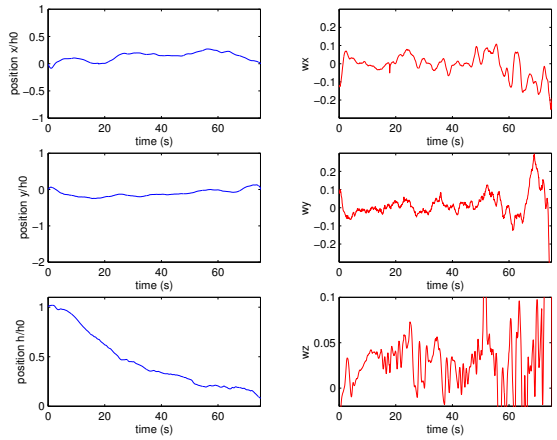

Fig. 5: Vertical landing using the controller (16)

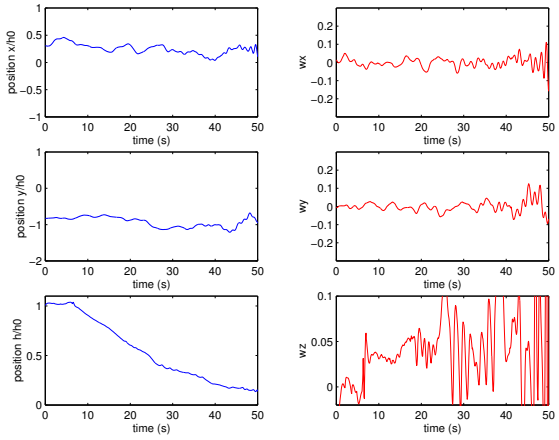

Fig. 6: Vertical landing using the controller (24)

VII. Concluding remarks

This paper presented a rigorous nonlinear controller for vertical landing of a VTOL UAV using the measurement of translational optical flow on a spherical camera along with the IMU data. Different controllers corresponding to different control objectives (hovering and vertical landing) have been proposed and the stability of the closed-loop systems has been analysed. Experimental results have been presented to show the performance of the approach considered.

VIII. Acknowledgments

This work was partially funded by Naviflow grant and by ANR project SCUAV (ANR-06-ROBO-0007).

References

[1] E. Altug, J. Ostrowski, and R. Mahony. Control of a quadrotor helicopter using visual feedback. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington DC, Virginia, June 2002.

[2] A. Beyeler, C. Mattiussi, J.-C. Zufferey, and D. Floreano. Vision-based altitude and pitch estimation for ultra-light indoor microflyers. In *Proceedings of the 2006 IEEE International Conference on Robotics & Automation*, Orlando, Florida, May 2006.

[3] McCarthy C., Barnes N., and Mahony R. A robust docking strategy for a mobile robot using flow field divergence. *Submitted to IEEE Transactions on Robotics*.

[4] J. S. Chahl, M. V. Srinivasan, and S. W. Zhang. Landing strategies in honeybees and applications to uninhabited airborne vehicles. *The International Journal of Robotics Research*, 23(2):101–110, 2004.

[5] N. Guenard, T. Hamel, and R. Mahony. A practical visual servo control for an unmanned aerial vehicle. *IEEE Transactions on Robotics*, 24:331–340, 2008.

[6] T. Hamel and R. Mahony. Visual servoing of an under-actuated dynamic rigid-body system: An image based approach. *IEEE Transactions on Robotics*, 18(2):187–198, 2002.

[7] Barron J. L, Fleet D. J., and Beauchemin S. S. Performance of optical flow techniques. *International Journal of Computer Vision*, 12(1):43–77, 1994.

[8] B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, pages 674–679, Vancouver, 1981.

[9] R. Mahony, P. Corke, and T. Hamel. Dynamic image-based visual servo control using centroid and optic flow features. *Journal of Dynamic Systems Measurement and Control*, 130(1,011005), 2008.

[10] N. Metni, J.M. Pflimlin, T. Hamel, and P. Souares. Attitude and gyro bias estimation for a flying uav. In *Proceedings of the IEEE International Conference on Intelligent Robots and Systems*, Edmonton, Canada, 2005.

[11] Laurent Muratet, Stéphane Doncieux, Yves Briere, and Jean-Arcady Meyer. A contribution to vision-based autonomous helicopter flight in urban environments. *Robotics and Autonomous Systems*, 50(4):195–209, 2005.

[12] Mahony R. and Hamel T. Robust trajectory tracking for a scale model autonomous helicopter. *International Journal of Non-linear and Robust Control*, 14:1035–1059, 2004.

[13] F. Ruffier and N. Franceschini. Visually guided micro-aerial vehicle: automatic take off, terrain following, landing and wind reaction. In *Proceedings of international conference on robotics and automation*, LA, New Orleans, April 2004.

[14] Bertrand S., Hamel T., and Piet-Lahanier H. Stability analysis of an uav controller using singular perturbation theory. In *Proceedings of the 17th IFAC World Congress (to appear)*, 2008.

[15] S. Saripalli, J.F. Montgomery, and G.S. Sakhatme. Vision based autonomous landing of an unmanned aerial vehicle. In *Proceedings of the IEEE International Conference on Robotics and Automation*, Washington DC, Virginia, June 2002.

[16] O. Shakernia, Y. Ma, T. J. Koo, and S. Sastry. Landing an unmanned air vehicle: vision based motion estimation and nonlinear control. *Asian Journal of Control*, 1(3):128–146, 1999.

[17] M.V. Srinivasan, S.W. Zhang, J. S. Chahl, E. Barth, and S. Venkatesh. How honeybees make grazing landings on flat surfaces. *Biological Cybernetics*, 83:171183, 2000.

[18] R.F. Vassallo, J. Santos-Victor, and H.J. Schneebeli. A general approach for egomotion estimation with omnidirectional images. In *OMNIVIS'02*, Copenhagen, Denmark, June 2002.

[19] Jean-Christophe Zufferey and Dario Floreano. Fly-inspired Visual Steering of an Ultralight Indoor Aircraft. *IEEE Transactions on Robotics*, 22(1):137–146, 2006.