Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

ROBUST CONTROL AND VISUAL SERVOING OF AN UAV

A. Dib, N. Zaidi and H. Siguerdidjane

SUPELEC, 3 Rue Joliot-Curie, 91192 Gif-sur-Yvette, France

Abstract: The purpose of this paper is to show that a combination of a nonlinear controller for an UAV of quadrotor type and visual servoing for trajectories generation leads to better stability results in a perturbed environment. The quadrotor is an underactuated system and highly nonlinear, which induces some difficulties in the design of the controller and in particular the application of visual servoing. The dynamic model of the quadrotor has been established by taking into account the gyroscopic effects of the rotors. The state-space representation of the system has been chosen in such a way that it enables to derive a backstepping controller. To generate the trajectories, three types of visual servoing have been investigated, 2D, 3D and 2D1/2, in order to draw the advantages and the drawbacks of each approach. Numerical simulations have been performed and have confirmed the validity of the theoretical results, besides the backstepping controller has shown the ability to control the quadrotor in the presence of relatively high perturbation conditions. *Copyright © 2008 IFAC.*

Keywords: UAV, Nonlinear systems, Backstepping, Visual servoing.

1. INTRODUCTION

The field of the UAVs is today in full rise due to the intensive interests they offer in the observation or the recognition areas. Unfortunately, inertial sensors (gyroscope, GPS…) do not allow to locate the UAV in its environment as expected, thus it is important to make it more autonomous while integrating an exteroceptive sensor such as a camera to provide an intrinsic capacity of perception and therefore using a visual servoing. The basic principle of visual servoing consists of taking into account visual information from a camera in order to control the movement of the system. The work on the visual control of the robots has first appeared, one decade ago, in for example (Chaumette, 1998; Malis, 1998). More recently, several research teams launched out in studies related to visual servoing of the machine wheels, one may see for instance (Chariette, 2001; Mahony and Hamel, 2005).

In this paper, our main goal is to ensure robustness in both tracking and trajectory generation. We used in this study an UAV of quadrotor type, but it can be, however, extended to include other kinds of systems. In order to extend our previous research work (Rontani and Siguerdidjane, 2007), we have used a combination of the backstepping based approach to design the controller, which might reach good performance in terms of stability and robustness, and visual servoing as well. This combination has not yet used in the literature, at least to our knowledge. Besides, we have here considered the gyroscopic effects of the UAV rotors in order to increase the validity of the modeling.

The paper is organized as follows. Section 2 describes the quadrotor's modeling, and the resulting model is analyzed in section 3. The Backstepping controller is derived in section 4. Section 5 is devoted to the theoretical study as well as the simulation results of the three types of visual applied controllers.

2. DYNAMIC MODELING OF THE QUADROTOR

The quadrotor consists of four cross parallel bars on which, at the ends of each extremity, are placed four engines controlling the rotors of vertical axes.

978-1-1234-7890-2/08/$20.00 © 2008 IFAC

10.3182/20080706-5-KR-1001.2951

Let's consider earth fixed frame *E*, and body fixed frame *B*, as shown in Fig. 1. Using Euler angles parameterization, the airframe orientation in space is given by a rotation *R* from *B* to *E*, where $R \in SO3$ is the rotation matrix.

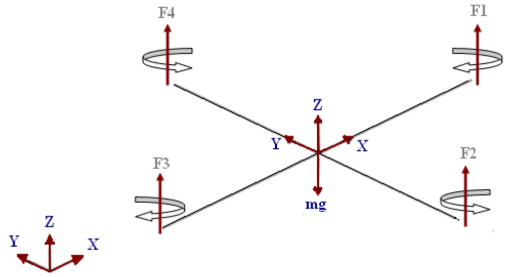

Fig. 1. Four engines generating the drag forces and the command torques

The four actuators generate four actions by means of four forces, which can equivalently be represented with one force and three moments.

The force $\hat{F} = \left[F_x \ F_y \ F_z\right]^T$ applied to the quadrotor is the thrust force created by all the rotors, and it can be expressed in the reference frame *B* as follows:

$$\begin{bmatrix} F_x \\ F_y \\ F_z \end{bmatrix}_B = \begin{bmatrix} 0 \\ 0 \\ U \end{bmatrix}_B = \begin{bmatrix} 0 \\ 0 \\ k(\omega_1^2 + \omega_2^2 + \omega_3^2 + \omega_4^2) \end{bmatrix}_B \tag{1}$$

In the inertial frame, we can write:

$$F = R(\phi, \theta, \psi)\hat{F} \tag{2}$$

where *R* denotes the rotation matrix.

The torque applied on the vehicle's body referential frame *B*, on one of the axes *x* and *y*, is the difference between the torques generated by two rotors on this axis. Both torques induce yaw and pitch movement along the (*x*, *y*) translations. The roll movement is due to the presence of a counter torque on each rotor's axis. The counter torque is generated by the aerodynamic drag force on the wings. Expressed in the reference frame *B*, the torque is given by:

$$\begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \end{bmatrix}_B = \begin{bmatrix} bl(\omega_4^2 - \omega_2^2) \\ bl(\omega_3^2 - \omega_1^2) \\ d(\omega_1^3 + \omega_3^2 - \omega_2^2 - \omega_4^2) \end{bmatrix}_B \tag{3}$$

where *l* is the characteristic length of the system, *k*, *b* and *d* are wing's geometry related coefficients and ω_i is the angular velocity of rotor *i*.

Let's also consider the torques resulting from the gyroscopic effects of the rotors, they can be written within the reference frame *B* as follows:

$$\begin{aligned} \tau_x &= I_r\dot{\theta}(\omega_3 + \omega_1 - \omega_2 - \omega_4) \\ \tau_y &= I_r\dot{\phi}(-\omega_3 - \omega_1 + \omega_2 + \omega_4) \end{aligned} \tag{4}$$

The dynamic model of the UAV is developed according to the Lagrangian approach, i.e. according to the potential and kinetic energies:

$$\Gamma_i = \frac{d}{dt}\left(\frac{\partial L}{\partial \dot{q}_i}\right) - \frac{\partial L}{\partial q_i} \qquad L = T - U_p \tag{5}$$

where Γ_i is the generalized forces given by the non preserving forces, *T* is the total kinetic energy, U_p is the total potential energy and q_i*'s* are the generalized coordinates.

Equations of translation: one may note that linear accelerations of the vehicle under the influence of the weight and the drag forces of the rotors are given by:

$$m\ddot{\xi} + \begin{pmatrix} 0 \\ 0 \\ mg \end{pmatrix} = R(\phi, \theta, \psi)\hat{F} \tag{6}$$

It is now easy to extract the expressions of these linear accelerations using (1) and (6):

$$\begin{aligned} \ddot{x} &= (\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi)(U/m) \\ \ddot{y} &= (\cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi)(U/m) \\ \ddot{z} &= -g + (\cos\phi \cos\theta)(U/m) \end{aligned} \tag{7}$$

Equations of rotation: The expression of the kinetic energy is given by:

$$\begin{aligned} T = \frac{1}{2}(I_x(\dot{\phi} - \dot{\psi}\sin\theta)^2 &+ I_y(\dot{\theta}\cos\phi + \dot{\psi}\cos\theta\sin\phi)^2 \\ &+ I_z(\dot{\theta}\sin\phi - \dot{\psi}\cos\phi\cos\theta)^2) \end{aligned} \tag{8}$$

where I_x, I_y and I_z are the inertia moments of the quadrotor.

By using (5) and applying the approximation of the small angles, where velocities of Cardan angles are identical to the angular velocities in the body reference *B*, we can then write:

$$\begin{aligned} \Gamma_\phi &= I_x\ddot{\phi} + (I_z - I_y)\dot{\theta}\dot{\psi} = \tau_1 + \tau_x \\ \Gamma_\theta &= I_y\ddot{\theta} + (I_x - I_z)\dot{\phi}\dot{\psi} = \tau_2 + \tau_y \\ \Gamma_\psi &= I_z\ddot{\psi} + (I_y - I_x)\dot{\theta}\dot{\phi} = \tau_3 \end{aligned} \tag{9}$$

From which it comes out, using (4):

$$\begin{aligned} \ddot{\phi} &= \frac{I_r\dot{\theta}(\omega_3 + \omega_1 - \omega_2 - \omega_4)}{I_x} + \frac{(I_y - I_z)}{I_x}\dot{\theta}\dot{\psi} \\ &\quad + \frac{bl(\omega_4^2 - \omega_2^2)}{I_x} \\ \ddot{\theta} &= \frac{I_r\dot{\phi}(-\omega_3 - \omega_1 + \omega_2 + \omega_4)}{I_y} + \frac{(I_z - I_x)}{I_y}\dot{\phi}\dot{\psi} \\ &\quad + \frac{bl(\omega_3^2 - \omega_1^2)}{I_y} \\ \ddot{\psi} &= \frac{(I_x - I_y)}{I_z}\dot{\theta}\dot{\phi} + \frac{d(\omega_1^2 + \omega_3^2 - \omega_2^2 - \omega_4^2)}{I_z} \end{aligned} \tag{10}$$

3. UAV SYSTEM ANALYSIS

The model developed in section 2 can be rewritten in a state space form $\dot{X} = f(X, U)$ by introducing $X = (x_1 \dots x_{12})^T \in \Re^{12}$ as the state vector, one may see for instance (Bouabdallah and Siegwart, 2005), in other words:

$$\begin{aligned} &x_1 = \phi,\ x_2 = \dot{\phi},\ x_3 = \theta,\ x_4 = \dot{\theta},\ x_5 = \psi,\ x_6 = \dot{\psi}, \\ &x_7 = z,\ x_8 = \dot{z},\ x_9 = x,\ x_{10} = \dot{x},\ x_{11} = y,\ x_{12} = \dot{y} \end{aligned} \tag{11}$$

From (7) and (10), it yields:

$$f(X,U)=\begin{pmatrix} x_2 \\ a_1 x_4 x_6 + a_2 x_4 \omega + b_1 \tau_1 \\ x_4 \\ a_3 x_2 x_6 + a_4 x_2 \omega + b_2 \tau_2 \\ x_6 \\ a_5 x_2 x_4 + b_3 \tau_3 \\ x_8 \\ -g + (1/m)(\cos x_1 \cos x_3) U \\ x_{10} \\ u_x (U/m) \\ x_{12} \\ u_y (U/m) \end{pmatrix} \tag{12}$$

where:

$$\begin{aligned} &a_1 = (I_y - I_z)/I_x, \quad a_2 = -J_r/I_x \quad a_3 = (I_z - I_x)/I_y \\ &a_4 = J_r/I_y, \qquad a_5 = (I_x - I_y)/I_z \\ &b_1 = l/I_x, \qquad b_2 = l/I_y, \quad b_3 = l/I_z \\ &\omega = \omega_2 + \omega_4 - \omega_1 - \omega_3 \end{aligned} \tag{13}$$

$$\begin{aligned} &u_x = \cos\phi \sin\theta \cos\psi + \sin\phi \sin\psi \\ &u_y = \cos\phi \sin\theta \sin\psi - \sin\phi \cos\psi \end{aligned} \tag{14}$$

According to the dynamic model established in the last section, we note that rotations and their time derivatives do not depend on the translation components. On the other hand, the translations depend on rotations. We can then consider the system as composed of two subsystems: angular rotations and linear translations, as shown in Fig. 2.

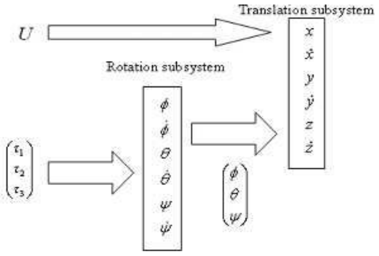

Fig. 2. Two subsystems: rotation and linear translation.

4. BACKSTEPPING CONTROL OF THE UAV SYSTEM

The complete control of this system may be divided into a controller of position and a controller of rotation (e.g. Bouabdallah and Siegwart, 2005), as shown in Fig. 3.

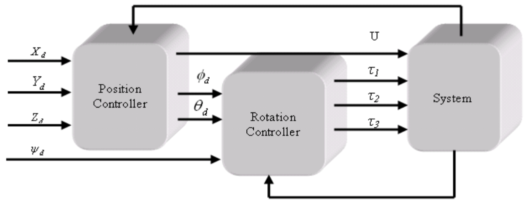

Fig. 3. Control block diagram.

4.1 Backstepping control of the rotation subsystem

To design a controller that forces the UAV system to track desired trajectories $x_{id}, i = 1 \cdots 8$, we have adopted the backstepping approach. Our objective is to force the tracking error to vanish, so we choose the following positive definite function as a Lyapunov function:

$$V(z_1) = \frac{1}{2} z_1^2 \tag{15}$$

where z_1 denotes the tracking error $z_1 = x_{1d} - x_1$.

To ensure that Lyapunov theory conditions are fulfilled, its time derivative must be negative semi-definite. To achieve this goal we introduce a virtual control input x_2

$$x_2 = \dot{x}_{1d} + \alpha_1 z_1 \qquad \alpha_1 > 0 \tag{16}$$

The time derivative of V is:

$$\dot{V}(z_1) = -\alpha_1 z_1^2 \tag{17}$$

Let's introduce a variable change by making:

$$z_2 = x_2 - \dot{x}_{1d} - \alpha_1 z_1 \tag{18}$$

Now consider the augmented Lyapunov function:

$$V(z_1, z_2) = \frac{1}{2}(z_1^2 + z_2^2) \tag{19}$$

and its time derivative is then:

$$\begin{aligned} \dot{V}(z_1, z_2) = \; & z_2 (a_1 x_4 x_6 + a_2 x_4 \omega + b_1 \tau_1) \\ & - z_2 (\ddot{x}_{1d} - \alpha_1 (z_2 + \alpha_1 z_1)) \\ & - z_1 z_2 - \alpha_2 z_1^2 \end{aligned} \tag{20}$$

In order to make $\dot{V}(z_1, z_2) < 0$, we can choose τ_1 in such a way that makes $\dot{V}(z_1, z_2)$ equal to $-\alpha_1 z_1^2 - \alpha_2 z_2^2$ with α_1, α_2 > 0. Thus one can extract the expression of τ_1 :

$$\tau_1 = \frac{1}{b_1}(\ddot{x}_{1d} + z_1 - a_1 x_4 x_6 - a_2 x_4 \omega - \alpha_1 (z_2 + \alpha_1 z_1) - \alpha_2 z_2) \tag{21.a}$$

Using the same way, one can extract τ_2 and τ_3, by introducing the variables $z_i ; i = 3, \cdots, 6$:

$$\begin{aligned} &\tau_2 = \frac{1}{b_2}(\ddot{x}_{3d} + z_3 - a_3 x_2 x_6 - a_4 x_2 \omega - \alpha_3 (z_4 + \alpha_3 z_3) - \alpha_4 z_4) \\ &\tau_3 = \frac{1}{b_3}(\ddot{x}_{5d} + z_5 - a_5 x_2 x_4 - \alpha_5 (z_6 + \alpha_5 z_5) - \alpha_6 z_6) \end{aligned} \tag{21.b}$$

with:

$$\begin{aligned} &z_3 = x_{3d} - x_3, \; z_4 = x_4 - \dot{x}_{3d} - \alpha_3 z_3, \; z_5 = x_{5d} - x_5 \\ &z_6 = x_6 - \dot{x}_{5d} - \alpha_5 z_5 \\ &\alpha_i > 0, \; i = 3, \cdots, 6 \end{aligned} \tag{22}$$

4.2 Backstepping control of the linear translation subsystem

- *Attitude control*: The attitude control U is obtained using the same approach as described above:

$$U = \frac{m}{\cos x_1 \cos x_3}(\ddot{x}_{7d} + z_7 + g - \alpha_7 (z_8 + \alpha_7 z_7) - \alpha_8 z_8) \tag{23}$$

with: $z_7 = x_{7d} - x_7$, $z_8 = x_8 - \dot{x}_{7d} - \alpha_7 z_7$

- *Linear x and y motion control:* we first use Backstepping approach to calculate u_x and u_y for a given command U, we obtain:

$$\begin{aligned} &u_x = (m/U)(\ddot{x}_{9d} + z_9 - \alpha_9 (z_{10} + \alpha_9 z_9) - \alpha_{10} z_{10}) \\ &u_y = (m/U)(\ddot{x}_{11d} + z_{11} - \alpha_{11} (z_{12} + \alpha_{11} z_{11}) - \alpha_{12} z_{12}) \end{aligned} \tag{24}$$

Then we use (14) to extract θ_d and φ_d, so we can write:

$$x_1 = \phi_d = \arcsin(u_x \sin\psi_d - u_y \cos\psi_d)$$
$$x_3 = \theta_d = \arcsin\left(\frac{u_x \cos\psi_d + u_y \sin\psi_d}{\cos\phi_d}\right) \tag{25}$$

4.3 Simulation results

We have performed a set of simulations using dynamic model (12) with 12 parameters $(\alpha_1 ....\alpha_{12})$ backstepping controller. We used a simple stop-stop trajectory, and we have applied a horizontal force on the UAV for a short period of time (2 s) to simulate a gust of wind. Fig. 4 shows a slight deviation from the reference during the wind but within acceptable range. It has been observed that the trajectory tracking remained unchanged when introducing structural uncertainties up to 30% with respect to the variables b, d, l, k, I_x, I_y and I_z.

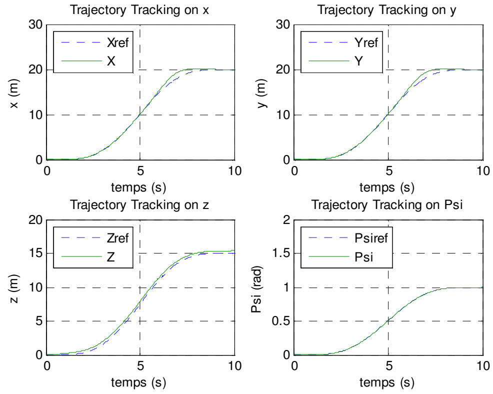

Fig. 4. Trajectory tracking of the UAV using backstepping controller and a simple stop-stop trajectory.

5. VISUAL SERVOING OF THE UAV SYSTEM

Visual servoing has been classified according to the controlled value:

- Visual control 2D: the reference, the measurement and the command law are defined in the image plane, in the forms of visual primitives. A primitive is an elementary geometrical form (point, segment…etc). It is used to model the projection of an object in the image plane.
- Visual control 3D: the reference and the measurement are defined by an attitude in Cartesian space. An object geometrical model of interest is necessary to estimate the measurement.
- Visual control 2D1/2: this type of control uses a combination of information expressed for some of them in the image and for others in the camera reference frame.

Visual information modeling: we define visual information as a particular geometrical form containing useful information. The whole visual information makes it possible to define the objective to be reached in terms of position and orientation. In this work, we consider visual information of type point only. Thus for a given point *i*, visual information s_i is defined by the following vector, for which the unit is the pixel

$$s_i = [u_i \quad v_i]^T \tag{26}$$

The time derivative of s makes it possible to connect the variations of visual information to the relative movement between the camera and the scene:

$$\dot{s} = \frac{\partial s}{\partial r}\dot{r} = L_S \dot{r} = L_S .W.T \tag{27}$$

where $r(t)$ is the situation, at time instant t, between the camera and its environment, L_s is a Jacobian matrix of dimension $2n \times 6$, where n is the number of visual information. It is called the Jacobian image or the interaction matrix associated to s, T is the relative kinematic torsor between the camera and the scene, and W is the transformation matrix of the kinematic torque from its expression in the camera reference frame to UAV reference frame.

5.1 Visual servoing 2D

For this type of visual servoing we choose as a task function:

$$e(r(t)) = C.(s(r(t)) - s^*) \quad C_{6\times 2n} = L^+_{S|S=S^*} \tag{28}$$

The matrix C is a constant matrix which is selected as a pseudo inverse of an approximation of the interaction matrix at the desired position.

The command must force the task function to tend to zero. So then, one may simply consider a task function with an exponential decreasing:

$$\dot{e} = -ge \tag{29}$$

g is a positive scalar value (note that we can choose g as a time variable parameter, it can be small at the initial condition to make sure that the command signal doesn't saturate, after which, it might be large enough for a fast convergence, as for example, a choice of $g(t) = g_{max}(1 - -e^{-t/a})$ with g_{max}, $a > 0$).

The variation of the error can be written using (27) and (28):

$$\dot{e} = C\ L_S\ WT \tag{30}$$

We can finally write, using (29):

$$T = (CL_S W)^{-1} g(s^* - s) \tag{31}$$

and the control loop is shown in Fig. 5.

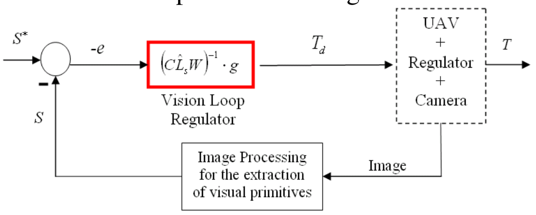

Fig. 5. Block diagram of the 2D regulator.

5.2 Visual servoing 3D

The principle of visual servoing 3D is to control the displacement of the UAV in the Cartesian space. It is possible to choose visual information that is not directly expressed in the image, but rather resulting from a reconstruction phase or localization 3D.

This information 3D can be obtained by a simple calculation of *pose* which is commonly used in visual servoing 3D.

The task function, in this type of visual control 3D, is (Malis, 1998):

$$e = r(s) - r(s^*) \tag{32}$$

Where r is the attitude of the object reference frame seen by the camera compared to UAV reference frame, this attitude is estimated using visual information.

One will adopt the same exponential decrease of the task function (29) by using the same adaptive gain. By following the same steps as done in the last section, the final expression of the kinematic torsor of the UAV is:

$$T = -g(LW)^{-1}e \tag{33}$$

L is the matrix of interaction which links the variation of the task function with respect to the speed of the camera.

Now let's consider the following task function:

$$e = \begin{bmatrix} {}^cT_o - {}^dT_o \\ {}^d\theta u_c \end{bmatrix} \tag{34}$$

Where ${}^cT_o$ is the homogeneous matrix of transformation between the current reference frame of the camera R_c, and the reference frame of the target R_o, expressed in the camera reference frame R_c, ${}^dT_o$ is the homogeneous matrix of transformation between the desired reference frame of the camera R_d, and the reference frame of the target R_o, expressed in the camera reference frame R_c, ${}^d\theta u_c$ is the minimal representation of the matrix ${}^dT_c$ (Khalil and Dombre, 1999), which represents the transformation between the desired reference frame of the camera R_d, and the current reference frame of the R_c one , and we have:

$${}^dT_c = {}^dT_o \cdot {}^cT_o^{-1} \tag{35}$$

The matrix of interaction associated with the selected task function is:

$$L = \begin{bmatrix} -I_3 & \left[{}^cP_o\right]_X \\ 0_3 & J_{c\omega} \end{bmatrix} \tag{36}$$

$$J_{c\omega} = I_3 - \frac{\theta}{2}[u]_x + \left(1 - \frac{\sin c\theta}{\sin c^2 \frac{\theta}{2}}\right)[u]_x^{\,2} \tag{37}$$

Where θ and u are the elements of the minimal representation of the rotation ${}^cR_o$, of the transformation homogeneous matrix. The control loop is shown in Fig. 6.

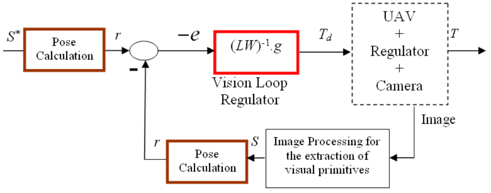

Fig. 6. Block diagram of the 3D regulator

5.3 Visual servoing 2D½

The problem of visual information 3D lies on the assumption that these data can be measured in a reliable way. In practice, they are more sensitive to measurement errors than visual information 2D, since they are obtained using this later along with the calculation of *pose*. Thus, it is interesting to combine visual information 2D and 3D to gain in robustness with respect to the measurement errors while preserving good properties of the decoupling.

The approach 2D½ is based on the decoupling between the feedback control loop in translation and the feedback control loop in rotation. It is therefore possible to combine a partial control of the trajectories in Cartesian space and in the image.

The task function in the case of a visual servoing 2D1/2 is defined as:

$$e = \begin{pmatrix} x - x^* & y - y^* & Ln(\frac{Z}{Z_d}) & \theta_u{}^T \end{pmatrix}^T \tag{38}$$

$(x, y)^T$ and $(x^*, y^*)^T$ are the current and desired metric coordinates of a characteristic point in the image, respectively. Z/Z_d is the ratio between the current and desired distance of this point about the camera, and θ_u is the minimal representation of rotation to be realized.

Using the same way as described in the last two sections we can write:

$$T = -g(LW)^{-1}e \tag{39}$$

According to chosen task function (38), the interaction of information 2D1/2 is (Malis, 1998; Khalil, 2002).

$$L = \begin{pmatrix} \frac{1}{Z}.L_{ev} & L_{ev\omega} \\ 0_3 & I_3 \end{pmatrix} \tag{40}$$

The control loop is shown in Fig. 7.

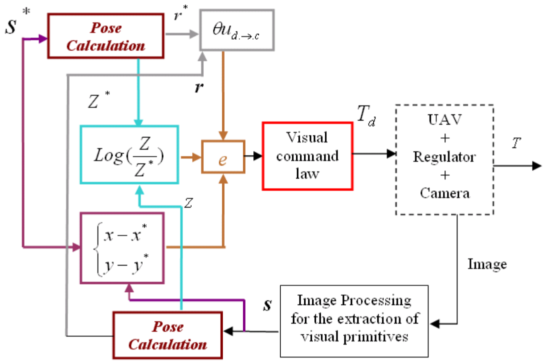

Fig. 7. Block diagram of the *2D½* regulator

5.4 Simulation results.

The feedback loop of the visual command and of the UAV one are in cascaded structure as described in the block diagrams of the various types of visual control, Figs. 5-7. Let us note that the cascaded structure forces the UAV command loop to being faster than the one of visual servoing, which implies to use larger values of the tuning parameters, while remaining within the limits tolerated by engines. Figures 8-13 show the trajectories evolution and satisfactory tracking.

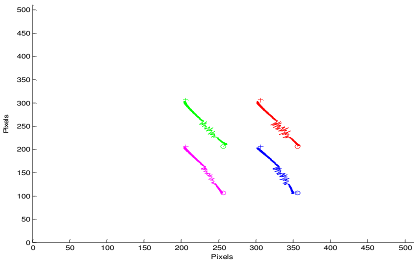

Fig. 8. Trajectory evolution in the image with disturbances, Visual servoing 2D

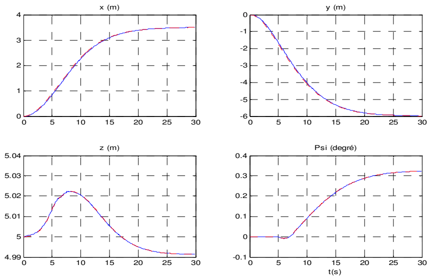

Fig. 9. Trajectory tracking with disturbances, Visual servoing 2D

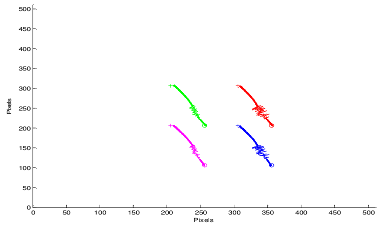

Fig. 10. Trajectory evolution in the image with disturbances, Visual servoing 3D

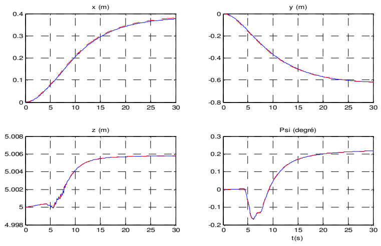

Fig. 11. Trajectory tracking with the presence of disturbances, Visual servoing 3D

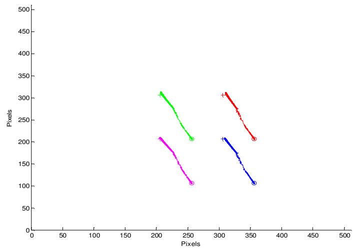

Fig. 12. Trajectory evolution in the image with disturbances, Visual servoing 2D1/2

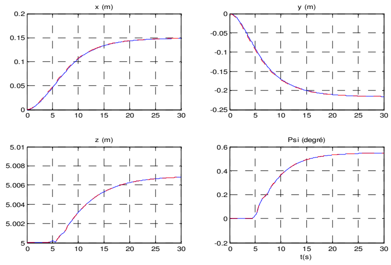

Fig. 13. Trajectory tracking with disturbances, Visual servoing 2D1/2

6. CONCLUSION

The controller derived, on the basis of the backstepping approach, with the objective of tracking trajectories generated by using visual servoing. This combination has led to satisfactory results in terms of stabilization and robustness. The advantage of visual servoing 2D is that it does not require a 3D reconstruction because the reasoning is directly realized on the level of the sensor, but the disadvantage is that it takes into account the current and desired images only, without taking into account the situation of the UAV, which can result in undesirable movements like the rotational ones. The system behavior in the 3D field is satisfactory from the point of view of the camera trajectory. The drawback is that no real control in the image is carried out, which implies that the object of interest can move out of the field of camera vision during its displacement. The main advantage of the visual control 2D1/2 technique is that the only 3D information used in command is the approximate desired depth of an object point, which means that the need for initial information is less useful than the case of visual servoing 2D and 3D.

REFERENCES

Bouabdallah S. and R. Siegwart (2005). Backstepping and Sliding mode Techniques Applied to an Indoor Micro Quadrotor. Proceeding *IEEE Int. Conf. on Robotics and Automation*, Barcelona.

Chariette A. (2001). Contribution à la commande et à la modélisation des hélicoptères : Asservissement visuel et commande adaptative. PhD thesis, Université d'Evry.

Chaumette F. (1998). Potential problems of stability and convergence in image-based and position-based visual servoing. *The confluence of vision and control*, Springer-Verglas, LNCIS 237, pp. 66-78.

Khalil W. and E. Dombre (1999). *Modélisation, identification et commande des robots.* Hermès, Paris, 2ème édition.

Khalil W. (2002). – *Commande des robots manipulateurs.* Hermès Science.

Mahony R. and T. Hamel (2005). Image-Based Visual Servo Control of Aerial Robotic Systems Using Linear Image Features. *IEEE Transactions On Robotics*, Vol. 21, No. 2.

Malis E. (1998). *Contributions à la modélisation et à la commande en asservissement visuel.* Thèse de l'Université de Rennes I, IRISA.

Rontani D. and H. Siguerdidjane (2007). Robust flatness based control and motion planning of an MUAV. CD-Rom Proceedings *17th IFAC Symposium au Automatic Control in Aerospace,* Toulouse.