The Task Function Approach Applied to Vision-based Control

François Chaumette* Patrick Rives** Bernard Espiau***

* IRISA-INRIA Rennes, Campus de Beaulieu, 35042 Rennes-cedex, France
** INRIA Sophia Antipolis, 2004 Route des Lucioles, 06565 Valbonne, France
*** ISIA-ENSMP, rue Claude Daunesse, Sophia Antipolis, 06565 Valbonne, France

Abstract – **This paper describes some concepts and results related to the vision-based control approach in robotics. The basic idea consists in considering a vision system as a specific sensor dedicated to a task and included in a control servo-loop. Once the necessary modeling stage is performed, the framework becomes the one of automatic control, and, naturally, stability and robustness questions arise. Starting from the task function approach, the general framework of the control is described, and some stability results are recalled. The concept of hybrid task is also presented and applied to visual sensors. Some simulation and experimental results are finally given.**

I. INTRODUCTION

Let us consider a rigid robot, the state equation of which is given by:

$$\Gamma = M(q)\ddot{q} + N(q,\dot{q},t)\,, \dim(q) = \dim(\Gamma) = n \quad (1)$$

where Γ is the vector of applied external forces (actuator torques), M is the kinetics energy matrix, N gathers gravity, centrifugal, Coriolis and friction forces and where $(q,\dot{q})$, the joint position and velocity, is the state vector of the system. It is assumed that an actuator is associated to every degree of freedom of the robot. We will also assume here for simplicity that $n = 6$.

Decoupling and linearizing (1) in the joint space is trivial as soon as the dynamics is known and computed. However, control in joint space is generally of little interest for the user: it is at least wished to control the position (i.e. location and attitude) of the frame linked to the 'last' body of the robot. The ideal decoupling and linearizing control takes then the form:

$$\Gamma = M(q)J^{-1}(q)u + N(q,\dot{q}) - M(q)J^{-1}(q)\begin{bmatrix} \vdots \\ \dot{q}^T \frac{\partial J_i^T}{\partial q}(q)\dot{q} \\ \vdots \end{bmatrix} \quad (2)$$

where $J(q)$ is the robot jacobian (we do not consider here the case where $J(q)$ falls singular), $J_i(q), i = 1,6$, the i-th row of $J(q)$, and u the new control vector. Nevertheless, this kind of control is not suitable in more complex (and interesting) applications, especially when exteroceptive sensors are used. Another working space is then required. This situation is a particular case of the more general 'control in task space', developed in [1], which will be briefly stated in section III.

We will therefore try in the present paper to show how a control in sensory space, extending in some way the scheme (2), may be designed, and we will apply this approach to the case of visual sensors (other cases are examined in [1] and [2]). It should be emphasized that, in robotics, this area, known as 'visual servoing' or 'vision-based control', is not as largely investigated as classical robot vision. Some relevant references are [3], [4], [5]. The related works will not be discussed here, since done in [6], to which we refer the reader.

In fact, modeling aspects and design of the adequate task function (i.e. of the output space associated to (1)) are the most delicate points, and we shall focus the development on these aspects. Section II will be therefore devoted to general considerations on sensor-environment interactions; the concept of hybrid task and the specific case of visual features are presented in section III. Finally experimental results are presented in section IV.

II. MODELING OF SENSOR-ENVIRONMENT INTERACTION

Let us consider the three-dimensional affine euclidean space, the related vector space being $\mathbb{R}^3$. The configuration space of a rigid body, which is also the frame configuration space, is the Lie group of displacements, SE_3 (Special Euclidean Group). It is a six-dimensional differential manifold, an element of which (a 'position') is denoted as $\bar{r}$.

A. The Interaction Matrix

We restrict our study to the case where a sensor is completely defined by a differentiable mapping from SE_3 to $\mathbb{R}^p$. This assumption implies in particular that, for a given sensor, relative environmental modifications of the geometrical kind between the sensor and the scene are the only ones allowed to make the sensor output varying. This is true for many kinds of proximity, range force and visual sensors. Let us now link a frame F_E to the part of environment observed by the sensor, and another, F_S, to the sensor itself. The sensor output s may then be written $s(F_E, F_S)$. Furthermore, let us assume that the sensor mobility is got through a generalized coordinate system, q, which constitutes a local chart of SE_3 (for example the joint angular positions of a rigid manipulator which handles a camera). Then, when the observed objects are autonomously mobile themselves, s may be also written $s = s(q,t)$, the independent time variable t representing the contribution of the objects velocity $\mathcal{T}$. We may finally write at $\bar{r}$ the basic relation:

$$\dot{s} = L_s^T\, T = L_s^T\, (J(q)\,\dot{q} - \mathcal{T}) \tag{3}$$

where $T = (V\ \Omega)^T$ represents the translational (V) and rotational (Ω) velocities of the frame F_E with respect to the frame F_S and where L_s^T depends both on the environment characteristics and on the sensor itself. It therefore fully characterizes the interaction between a sensor and its environment, and we thus call it *Interaction Matrix*.

B. Case of a Visual Sensor

Let us reduce a camera to a perspective projection model. Without loss of generality, the focal length is assumed to be equal to 1, such that any point m with coordinates $\underline{x} = (x\,y\,z)^T$ is projected on the image plane as a point M with coordinates $\underline{X} = (X\,Y\,1)^T$ with:

$$\underline{X} = \frac{1}{z}\,\underline{x} \tag{4}$$

Differentiating (4) leads to the well known optical flow equation which provides the interaction matrix related to X and Y:

$$\begin{array}{rcl} \dot{X} & = & L_X^T\, T \\ \dot{Y} & = & L_Y^T\, T \end{array} \tag{5}$$

with:

$$\begin{array}{l} L_X^T = (\ -1/z \quad 0 \quad X/z \quad XY \quad -1-X^2 \quad Y\ ) \\ L_Y^T = (\ 0 \quad -1/z \quad Y/z \quad 1+Y^2 \quad -XY \quad -X\ ) \end{array} \tag{6}$$

Various sensor signals may be generated from image points. Let us for example consider a 3D segment, limited by the points m_1 and m_2. It may be represented in the image either by the coordinates of its end points, M_1 and M_2 (we then have $L_{X_1}, L_{Y_1}, L_{X_2}, L_{Y_2}$ given by (6)), or by its length l, its orientation α and the coordinates X_c, Y_c of its center. In that case, the related interaction screws may also be easily derived ([7]):

$$\begin{array}{ll} L_l^T = (& \lambda_1 c_\alpha \qquad\qquad \lambda_1 s_\alpha \\ & \lambda_2 l - \lambda_1(X_c c_\alpha + Y_c s_\alpha) \qquad l[X_c c_\alpha s_\alpha + Y_c(1+s_\alpha^2)] \\ & -l[X_c(1+c_\alpha^2) + Y_c c_\alpha s_\alpha] \qquad 0 \quad) \\ L_\alpha^T = (& -\lambda_1 s_\alpha/l \qquad\qquad \lambda_1 c_\alpha/l \\ & \lambda_1(X_c s_\alpha - Y_c c_\alpha)/l \qquad -X_c s_\alpha^2 + Y_c c_\alpha s_\alpha \\ & X_c c_\alpha s_\alpha - Y_c c_\alpha^2 \qquad -1 \quad) \\ L_{X_c}^T = (& -\lambda_2 \qquad\qquad 0 \\ & \lambda_2 X_c - \lambda_1 l c_\alpha/4 \qquad X_c Y_c + l^2 c_\alpha s_\alpha/4 \\ & -(1 + X_c^2 + l^2 c_\alpha^2/4) \qquad Y_c \quad) \\ L_{Y_c}^T = (& 0 \qquad\qquad -\lambda_2 \\ & \lambda_2 Y_c - \lambda_1 l s_\alpha/4 \qquad 1 + Y_c^2 + l^2 s_\alpha^2/4 \\ & -X_c Y_c - l^2 c_\alpha s_\alpha/4 \qquad -X_c \quad) \end{array} \tag{7}$$

with $\lambda_1 = (z_1 - z_2)/z_1 z_2$, $\lambda_2 = (z_1 + z_2)/2z_1 z_2$, $c_\alpha = \cos\alpha$ and $s_\alpha = \sin\alpha$.

Other more complex primitives (lines, cylinders, spheres,...) are examined in [7].

C. The Concept of Virtual Linkage

A set of *compatible* and *independent* constraints, $s - s_d = 0$, where s_d is a desired sensor value, constitutes a *virtual linkage* between the sensor (S) and the objects of the environment (E).

Let T^* be a motion at $\bar{r}$ keeping constant the sensor output component s_j, i.e. preserving the satisfaction of the jth constraint. T^* is solution of the equation $L_j^T T^* = 0$. Let us now return to the full sensor output vector, s, with dimension p. The set of motions T^* leaving s invariant is the subspace S^* such that $S^* = \text{Ker}\ L_s^T$. The dimension, N, of S^* is called the *class* of the virtual linkage in $\bar{r}$. Let $m = 6 - N$, when $m = p$, the dimension of the signal vector s is adequate, in the intuitive sense that it is indeed the number of 'degrees of freedom' to be controlled from s. However, the case $p > m$ often offers some practical advantages, for example because of the filtering effects or simplification it may induce. We will therefore include this case in the analysis.

The concept of virtual linkage may be related to the basic kinematics of contacts, as classically used in the theory of mechanisms and may include the physical linkage when contact sensors are used. The idea of virtual linkage will allow us to design the wished sensor-referenced robotics tasks in a simple way. This will also establish a connection with the approach known in the literature as 'hybrid control', which is traditionally used in control schemes involving contact force sensors. This finally shows that many types of sensors may be used within a single framework: the one of *hybrid tasks* which realize *virtual linkages*.

III. TASKS AND CONTROL DESIGN

A. The concept of task function

The dynamic behavior of a rigid manipulator is described by (1). The task to be performed may then be specified as an *output function* associated to (1). The problem is indeed well-posed if the passage between the 'control space' and the 'output space' is regular in some sense.

More precisely, it may be shown ([1]) that the user's objective may in general be expressed as the regulation to zero of some n-dimensional C^2 function, $e(q,t)$, called *task function*, during a time interval $[0, t_m]$. An immediate example of task functions is $e(q,t) = x(q) - x_d(t)$ where $x_d(t)$ is for example a parametrization of the desired position of a robot wrist in SE_3. When sensors are used, it appears that the sensor vector $s(q,t)$ has to contribute to the design of the task function, in a way explained later.

As detailed in [1], the problem of regulating e is well-posed if e has some specific properties. One of them is the existence and the unicity of a C^2 *ideal trajectory*, $q_r(t)$, such that $e(q_r(t),t) = 0$, $\forall t \in [0,t_m]$ and $q_r(0) = q_0$, where q_0 is a given initial condition. Another one, very important, is the non-singularity of the task-jacobian matrix $\frac{\partial e}{\partial q}(q,t)$, around $q_r(t)$. When all the required conditions are satisfied, which will be implicitly assumed in the following, the task function is said to be 'admissible'. Efficient control laws may then be designed.

B. Control and stability

We only give here an intuitive idea of the used approach and of the obtained results. All the related developments may be found in [1].

Let us consider the exact decoupling and feedback linearization in the task space: in a way similar to (2), it is easy to see that an adequate control is:

$$\Gamma = M\left(\frac{\partial e}{\partial q}\right)^{-1} u + N - M\left(\frac{\partial e}{\partial q}\right)^{-1} f \tag{8}$$

$$\text{with } f = \begin{bmatrix} \vdots \\ \dot{q}^T \ \frac{\partial E_i^T}{\partial q}(q,t)\ \dot{q} \\ \vdots \end{bmatrix} + 2\frac{\partial^2 e}{\partial q \partial t}(q,t)\dot{q} + \frac{\partial^2 e}{\partial^2 t}(q,t)$$

where $E_i, i = 1,n$, is the i-th row of $\frac{\partial e}{\partial q}$. Furthermore, the control vector u is, for example, a PD feedback of the form:

$$u = -\lambda\, G\, (\mu\, D\, e + \dot{e}) \tag{9}$$

λ and μ being positive scalars, G and D being positive matrices, all to be tuned by the user (a matrix $A(n \times n)$ is positive if $x^T A x > 0$, $\forall x \neq 0 \in \mathbb{R}^n$).

The ideal control scheme (8) requires a perfect knowledge of all its components, which is neither possible, nor even wished. A more realistic approach consists in generalizing the previous control as:

$$\Gamma = -\lambda \hat{M}\left(\widehat{\frac{\partial e}{\partial q}}\right)^{-1} G\left(\mu D e + \widehat{\frac{\partial e}{\partial q}}\dot{q} + \widehat{\frac{\partial e}{\partial t}}\right) + \hat{N} - \hat{M}\left(\widehat{\frac{\partial e}{\partial q}}\right)^{-1}\hat{f} \tag{10}$$

where the carets point out that models (approximations, estimates) are used instead of the true terms. The control (10) includes most of existing schemes: computed torque, resolved motion rate or acceleration control, indirect adaptive control,...

A stability analysis of the system (1) with control (10) was done by Samson ([1]) in a nonlinear framework. Two main classes of sufficient stability conditions (in the sense of the boundedness of $||e(t)||$) were then exhibited: **gain** conditions (these tuning parameters leave more or less possibilities to the user) and **modeling** conditions. Among them, those related to the robot dynamics are not too strong in practice, owing for example to the symmetric-positive definiteness of the kinetics energy matrix. Another sufficient one, much more critical, concerns the task itself, and has the form:

$$\frac{\partial e}{\partial q}\left(\widehat{\frac{\partial e}{\partial q}}\right)^{-1} > 0 \tag{11}$$

This essential condition allows to characterize the robustness of the task itself with regard to uncertainties and approximations. It may be noticed that, when we are interested in the motion of the end effector, we may write $\frac{\partial e}{\partial q} = \frac{\partial e}{\partial \bar{r}}\ J(q)$. When $J(q)$ is known and nonsingular, as we shall assume afterwards, the choice $\widehat{\frac{\partial e}{\partial q}} = \widehat{\frac{\partial e}{\partial \bar{r}}}\ J(q)$ allows the condition (11) to be reduced to:

$$\frac{\partial e}{\partial \bar{r}}\left(\widehat{\frac{\partial e}{\partial \bar{r}}}\right)^{-1} > 0 \tag{12}$$

C. Hybrid Tasks

In most cases, the space which has to be controlled during a robotics task can be splitted in two subspaces: one is devoted to sensor-based control, the other is used to satisfy a second objective like a trajectory tracking. Generally, the problem specification leads in a first step to defining a sensor-based task vector, $e_1(q,t)$, with $m \leq 6$ independent components, the regulation of which constitutes the part of the global task which requires the use of exteroceptive sensors. A second objective, for example a desired sensor motion, might be represented in a first glance by a second vector $e_2(q,t)$. However, e_1 and e_2

would be gathered in a single task vector $e(q,t)$ **admissible**. These two tasks would therefore be compatible and independent, which intuitively means, in terms of virtual linkage, that the secondary goal may be reached owing to all the realizable motions left available by the virtual linkage associated to the sensor-based task.

It may indeed be shown that a more efficient way of setting the problem consists in embedding it in the framework of ***task redundancy***. In this approach, e_1 is considered as priority, and e_2 is defined as the representation of the constrained minimization of a secondary cost function. Let us now recall some basic results in this domain, taken from [8].

We are interested in regulating s around a desired value s_d in order to realize a virtual linkage of class $N = 6 - m$. Let us recall that s is of dimension p and that the jacobian of s in SE_3 corresponds to the interaction matrix L_s^T (the dimension of L_s is $6 \times p$ and its rank is m for $s = s_d$). Let C be a 'combination matrix', with dimension $m \times p$, such that CL_s^T is of full rank m along the ideal trajectory $q_r(t)$. The main task may then be written:

$$e_1 = C\,(s(q,t) - s_d) \tag{13}$$

Let h, with gradient $\frac{\partial h}{\partial \bar{r}}$, be a secondary cost function to be minimized. Minimizing h under the constraint $e_1 = 0$ requires the subspace of motions left free by this constraint to be determined. This comes back to knowing the null space, Ker (J_1), of the jacobian matrix $J_1 = \frac{\partial e_1}{\partial \bar{r}}$ along $q_r(t)$. In other words, it has to be found any $m \times n$ full rank matrix W such that:

$$\text{Ker }(W) = \text{Ker }(J_1) \tag{14}$$

along $q_r(t)$. In our case, we have $J_1 = CL_s^T$ and Ker (J_1) = Ker $(L_s^T) = S^*$ for all positions such that $s = s_d$. Thus property (14) becomes Ker $(W) = S^*$.

Once this matrix is determined, it may rather easily be shown that a task function minimizing h under the constraint $e_1 = 0$ is:

$$e = W^+ e_1 + \beta\,(\mathbb{I}_6 - W^+ W)\,\frac{\partial h}{\partial \bar{r}}^T \tag{15}$$

where β is a positive scalar, W^+ is the pseudo-inverse of W and where $(\mathbb{I}_6 - W^+W)$ is an orthogonal projection operator on the null space of W, i.e. on that of J_1.

It clearly appears that the computation of the jacobian matrix related to (15), required in the control scheme, may be complex. The positivity condition (12) may then be of some interest. It may indeed be shown (see [1]) that if the property:

$$J_1 W^T = CL_s^T W^T > 0 \tag{16}$$

is satisfied along $q_r(t)$, then, under 'normal circumstances', the condition (12) is satisfied by taking $\frac{\widehat{\partial e}}{\partial \bar{r}} = \mathbb{I}_6$.

For example, (16) may be satisfied by selecting $C = WL_s^{T+}$, or even $C = W\hat{L}_s^{T+}$ where $\hat{L}_s$ is an approximation of L_s. The possible choices for $\hat{L}_s$ are discussed in [6].

IV. RESULTS

Several examples, obtained in simulation or with an experimental testbed, are reported in [6] and [7]. We only give here a simple illustration of the proposed approach.

Let us suppose that it is wished to set the camera with respect to a plane object which may be characterized by four points defining a square. Let us choose as sensor signals $s = (X_c, Y_c, l_1, l_2, l_3, l_4)$ where X_c and Y_c are the coordinates of the projection in the image of the center of gravity of the square and where $l_1, \ldots l_4$ are the length in the image of the four segments limited by a square vertex and by the center of gravity of the square.

We choose $s_d = (0, 0, l, l, l, l)$ in order that, at a desired position of the camera, the image of the square is a centered square with an unconstrained orientation. From (7), we may easily compute the interaction matrix related to s_d:

$$L^T_{|s=s_d} = \begin{pmatrix} -1/z_d & 0 & 0 & 0 & -1 & 0 \\ 0 & -1/z_d & 0 & 1 & 0 & 0 \\ 0 & 0 & l/z_d & l^2 s_{\alpha_1} & -l^2 c_{\alpha_1} & 0 \\ 0 & 0 & l/z_d & l^2 s_{\alpha_2} & -l^2 c_{\alpha_2} & 0 \\ 0 & 0 & l/z_d & l^2 s_{\alpha_3} & -l^2 c_{\alpha_3} & 0 \\ 0 & 0 & l/z_d & l^2 s_{\alpha_4} & -l^2 c_{\alpha_4} & 0 \end{pmatrix} \tag{17}$$

where z_d is the desired range from camera to square.

$L^T_{|s=s_d}$ is always of rank 5 and $s - s_d$ constitutes a virtual linkage of class 1: $S^* = (0\;0\;0\;0\;0\;1)^T$.

Let us now apply the approach of the previous section for the derivation of e. The matrix W may be chosen as $(\mathbb{I}_5\;\;0)$ and the combination matrix as:

$$C = \begin{pmatrix} -z_d & 0 & z_d c_{\alpha_1}/2l^2 & z_d c_{\alpha_2}/2l^2 & z_d c_{\alpha_3}/2l^2 & z_d c_{\alpha_4}/2l^2 \\ 0 & -z_d & z_d s_{\alpha_1}/2l^2 & z_d s_{\alpha_2}/2l^2 & z_d s_{\alpha_3}/2l^2 & z_d s_{\alpha_4}/2l^2 \\ 0 & 0 & z_d/4l & z_d/4l & z_d/4l & z_d/4l \\ 0 & 0 & s_{\alpha_1}/2l^2 & s_{\alpha_2}/2l^2 & s_{\alpha_3}/2l^2 & s_{\alpha_4}/2l^2 \\ 0 & 0 & -c_{\alpha_1}/2l^2 & -c_{\alpha_2}/2l^2 & -c_{\alpha_3}/2l^2 & -c_{\alpha_4}/2l^2 \end{pmatrix} \tag{18}$$

since the positivity condition (16) is satisfied in the neighborhood of the desired positions: indeed, we have $CL^T_{|s=s_d} W^T = \mathbb{I}_5$ if, $\forall \alpha_1 \in \mathbb{R}$, $\alpha_2 = \alpha_1 + \pi/2$, $\alpha_3 = \alpha_1 + \pi$ and $\alpha_4 = \alpha_1 - \pi/2$.

- **Remark:** If the dimension of the square is unknown, it is not possible to set the final range z_d from camera to object. Positioning at an unknown range is however possible by setting the desired length l in the image and by setting, for example, $z_d = 1$ in C. Indeed, we thus have:

$$CL^T_{|s=s_d} W^T = \begin{pmatrix} \mathbb{I}_3/z_d & 0 \\ 0 & \mathbb{I}_2 \end{pmatrix} > 0 \tag{19}$$

The secondary task may consist in specifying a time rotation around axis $\vec{z}$ of the camera, for example at a constant velocity ω_z. The associated secondary cost to be minimized is $h = \frac{1}{2}(\theta_z(t) - \theta_{z0} - \omega_z t)^2$ with $\theta_z(0) = \theta_{z0}$. Therefore $\frac{\partial h}{\partial \bar{r}} = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & (\theta_z(t) - \theta_{z0} - \omega_z t) \end{pmatrix}$. Note that tasks e_1 and $e_2 = \theta_z(t) - \theta_z 0 - \omega_z t$ are then compatible and independent since:

$$e = \begin{pmatrix} \mathbb{I}_5 \\ 0 \end{pmatrix} e_1 + \beta \begin{pmatrix} 0 \\ \theta_z(t) - \theta_{z0} - \omega_z t \end{pmatrix} \quad (20)$$

In the control law (10), since the condition (16) is satisfied, we can choose $\widehat{\frac{\partial e}{\partial q}} = J(q)$. An expression of the term $\widehat{\frac{\partial e}{\partial t}}$ is also needed. Considering the task function given by (20), we have:

$$\frac{\partial e}{\partial t} = \begin{pmatrix} \mathbb{I}_5 \\ 0 \end{pmatrix} \frac{\partial e_1}{\partial t} + \beta \begin{pmatrix} 0 \\ -\omega_z \end{pmatrix} \quad (21)$$

Vector $\frac{\partial e_1}{\partial t}$ represents the contribution of a possible autonomous target motion and is in general unknown. The choice made in many cases is $\widehat{\frac{\partial e_1}{\partial t}} = 0$. If the target moves, this choice may lead to a tracking error. On the other hand, since in trajectory tracking, the used secondary cost function allows to know $\frac{\partial}{\partial t}\left(\frac{\partial h}{\partial \bar{r}}\right)$, we may choose:

$$\widehat{\frac{\partial e}{\partial t}} = \beta \begin{pmatrix} 0 \\ -\omega_z \end{pmatrix} \quad (22)$$

Fig. 1 gives an example of the obtained behavior with $\omega_z = -1.25$ dg/s and with $\beta = 1$. Left and middle top windows show respectively initial and final positions of the camera (symbolized by a pyramid) with respect to the target. Left and middle bottom windows represent the associated images. On right windows, the time variation of $\|s - s_d\|$ and of the components (in cm/s and dg/s) of the camera velocity are respectively plotted. The exponential decreasing of $\|s - s_d\|$ and the convergence of the control law are ensured even for an initial position of the camera far away from the desired one.

Finally, Fig. 2 presents a sequence of real images acquired during the realization of the same task. The corresponding plots, close to the one obtained in simulation, show the robustness of the approach.

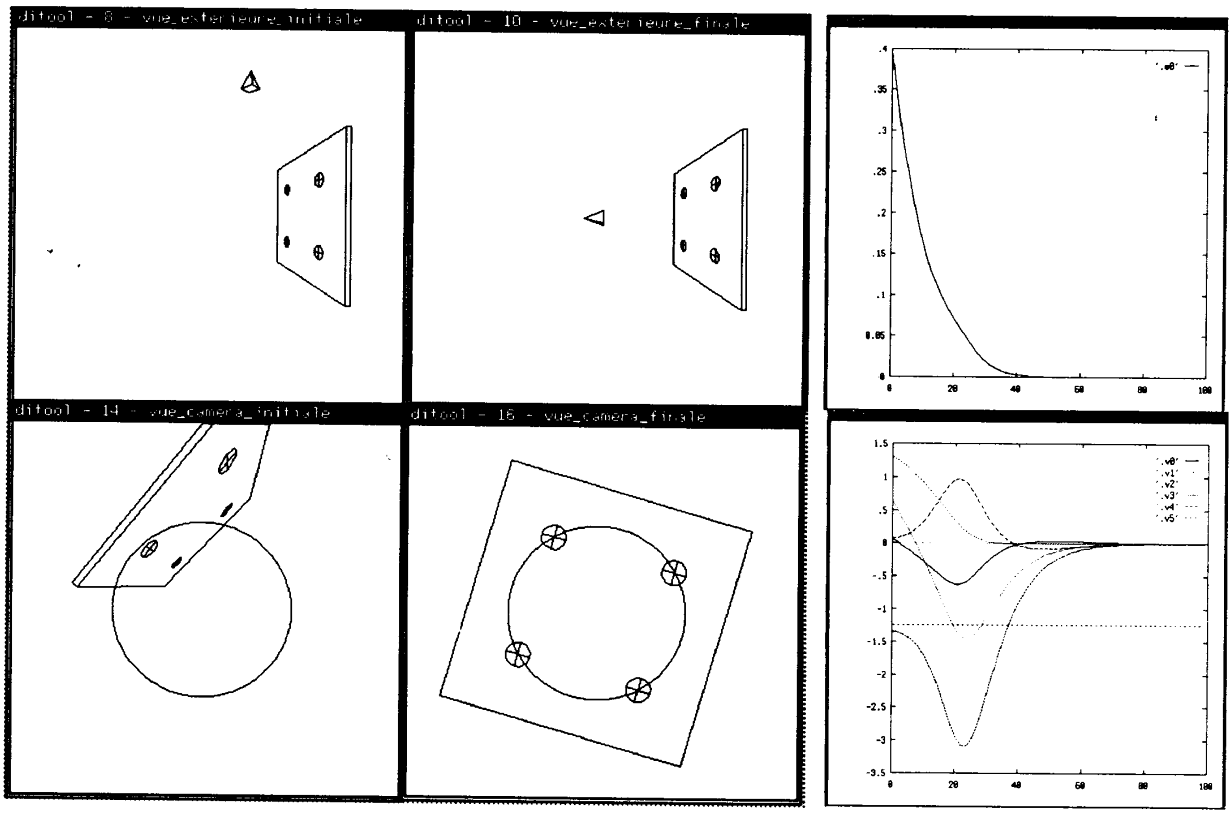

Fig. 1. Simulation results

V. CONCLUSION

We described in that work some problems and solutions related to the control of a robot in a space of visual sensors. One possible development of this work lies in the use of an adaptive approach of the control scheme. Indeed, if we may consider that intrinsic system parameters (inertia, kinematics, camera parameters...) which are not liable to large variations, may be computed or estimated off-line, on the contrary, uncertainties on the environment, which have a strong influence on the control behavior, have to be considered carefully, especially when objects have autonomous motion or when estimation algorithms are necessary in the control law. Finally, a natural extension will be to revisit the *structure and motion problem* from this new point of view and to embed it in a more general scheme based on an *active vision* approach.

REFERENCES

[1] C. Samson, B. Espiau, M. Le Borgne, *Robot control: the task function approach*, Oxford University Press, 1991.

[2] B. Espiau, J.P. Merlet, C. Samson, "Force feedback control and non-contact sensing: a unified approach," *8th CISM-IFTOMM Symp. on Theory and Practice of Robots and Manipulators*, 2-6 July 1990, Cracow-Poland.

[3] P.I. Corke, R.P. Paul, "Video-rate visual servoing for robots," *1st Int. Symp. on Experimental Robotics*, Montreal, Canada, June 1989.

[4] J.T. Feddema, O.R. Mitchell, "Vision-guided servoing with feature-based trajectory generation," *IEEE Trans. on Robotics and Automation*, Vol. 5, n. 5, October 1989.

[5] L.E. Weiss, A.C. Sanderson, "Dynamic sensor-based control of robots with visual feedback," *IEEE J. of Robotics and Automation*, Vol. RA-3, n. 5, October 1987.

[6] B. Espiau, F. Chaumette, P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. on Robotics and Automation*, in press.

[7] F. Chaumette, "La relation vision-commande: théorie et application à des tâches robotiques, *Ph-D. Thesis* Rennes I University, France, July 1990.

[8] C. Samson, B. Espiau, "Application of the task function approach to sensor-based control of robot manipulators," *IFAC*, Tallin, USSR, July 1990.

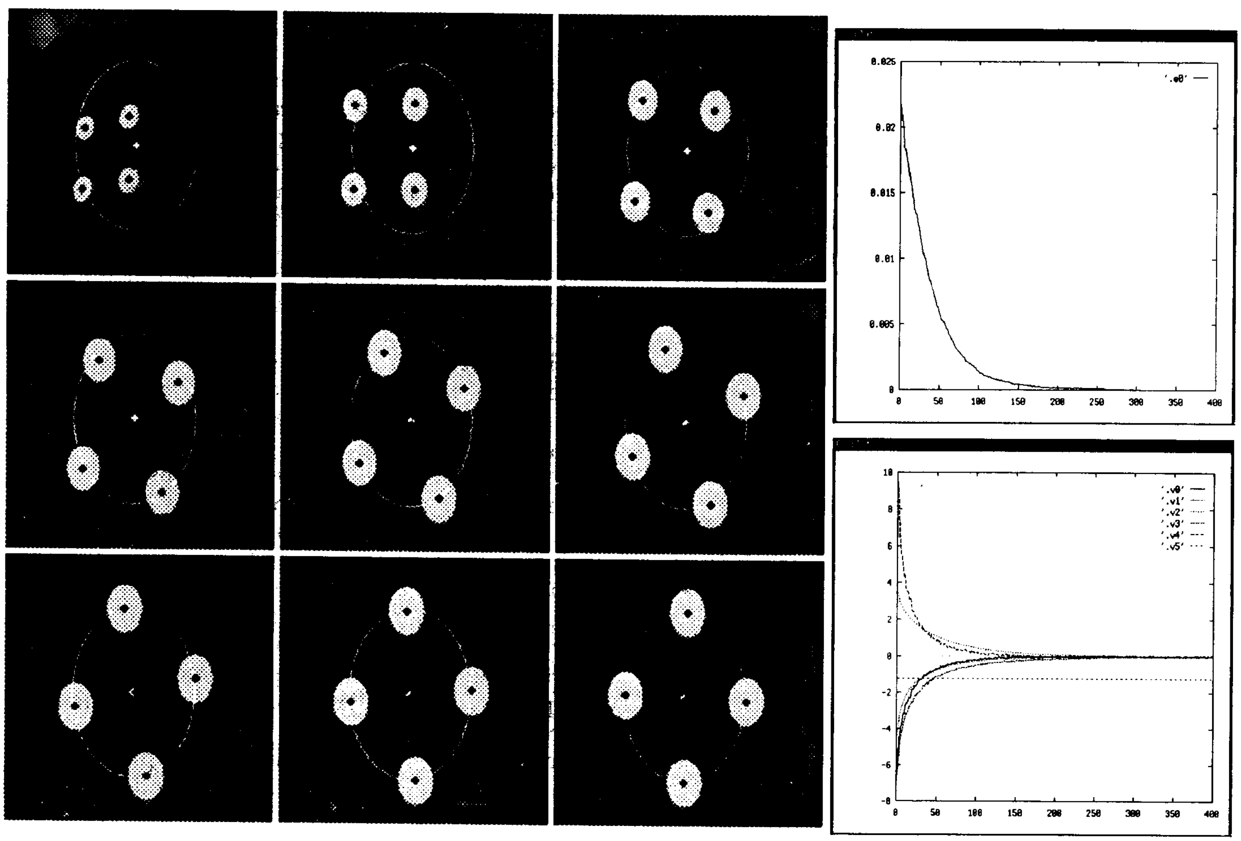

Fig. 2. Experimental results