Application of moment invariants to visual servoing

O. Tahri, François Chaumette

Application of Moment Invariants to Visual Servoing

Omar Tahri
IRISA/INRIA Rennes
Campus de Beaulieu
35 042 Rennes-cedex, France
E-mail: otahri@irisa.fr

François Chaumette
IRISA/INRIA Rennes
Campus de Beaulieu
35 042 Rennes-cedex, France
E-mail: chaumett@irisa.fr

***Abstract*— In this paper, we present how moment invariants can be used to design a decoupled 2D visual servoing scheme and to minimize the nonlinearity of the interaction matrix related to the selected visual features. Experimental results using a 6 dof eye-in-hand system to position a camera parallel to planar objects of complex shape are presented to demonstrate the efficiency of the proposed method.**

I. Introduction

In 2D visual servoing [8], [10], the control of the robot motion is performed using data directly extracted from the images acquired by one (or several) camera(s). Let $\mathbf{s}$ be a set of k features that characterize the image of the considered object. Once $\mathbf{s}$ is given by a differentiable function, the time derivative of $\mathbf{s}$ and the relative motion between the camera and the object can be related by the classical equation:

$$\dot{\mathbf{s}} = \mathbf{L_s T} \qquad (1)$$

where $\mathbf{T}$ is the relative velocity between the camera and the object. The matrix $\mathbf{L_s}$ is called the features Jacobian or the interaction matrix related to $\mathbf{s}$. The control scheme consists in canceling the task function:

$$\mathbf{e} = \mathbf{C}(\mathbf{s} - \mathbf{s}^*) \qquad (2)$$

where $\mathbf{s}$ is the current state and $\mathbf{s}^*$ the desired state. To control the 6 dof of the system, we usually choose $\mathbf{C} = \mathbf{I}_6$ if $k = 6$. If we specify an exponentially decoupled decrease of the task function:

$$\dot{\mathbf{e}} = -\lambda\, \mathbf{e} \qquad (3)$$

where λ is a proportional gain, the control law for an eye-in-hand system observing a static object is given by:

$$\mathbf{T_c} = -\lambda\, \widehat{\mathbf{L_s}}^{+} \mathbf{e} \qquad (4)$$

where $\mathbf{T_c} = (V_X, V_Y, V_Z, \Omega_X, \Omega_Y, \Omega_Z)^T$ is the camera velocity sent to the low-level robot controller, $\widehat{\mathbf{L_s}}$ is a model or an approximation of $\mathbf{L_s}$, and $\widehat{\mathbf{L_s}}^{+}$ the pseudo-inverse of $\widehat{\mathbf{L_s}}$. Several kind of image features $\mathbf{s}$ were proposed in the past. Most works were concerned with known and simple objects. They assume that the objects in the scene can be expressed with simple features such as points, straight lines, ellipses and more. The group of the objects that these methods can be applied to is thus limited. These methods have also the basic requirement of feature matching between initial and desired images, which is generally not easy to obtain. Other methods try to surmount the problems mentioned above, by using for example the Eigen space method [7] or the polar signature of an object contour [5]. Recently, a new method was proposed using image moments [4]. In this paper, we propose significant improvements to this method.

To date, an appropriate question in the visual servoing field is to determine the visual features to use in the control scheme in order to obtain an optimal behavior of the system. A first necessary condition of the convergence is that the interaction matrix must be not singular. Hence, a good choice of the features must allow to obtain a large domain where the matrix $\mathbf{L_s}$ has full rank 6. A good way to ensure this condition is to design a decoupled control scheme, i.e. to try to associate each camera dof with only one visual feature. Such control would make easy the determination of the potential singularities of the considered task, as well as the choice of $\widehat{\mathbf{L_s}}$. Unfortunately, a such totally decoupled control is ideal and seems impossible to reach. It is however possible to decouple the translational motion from the rotational one. In practice, it can be obtained using 3D visual servoing [19], but this approach requires the knowledge of a 3D CAD model of the object. It can also be obtained using $2\ 1/2D$ visual servoing [12], where the knowledge of a 3D model is not required. However, in this case, a homography matrix must be computed at each iteration of the control scheme. In 2D visual servoing, first attempts have been recently proposed in [6], [4]. In this paper, we present a more efficient method using moment invariants.

In 2D visual servoing, the behavior of the features in the image is generally satisfactory. On the other hand, the robot trajectory in 3D space is quite unpredictable and may be really unsatisfactory for large rotational displacements [3]. In fact, the difference of behaviors in image space and 3D space is due to the non linearities in the interaction matrix. To explain that, let us consider the basic interaction matrix related to the coordinates (x, y) of an image point:

$$\mathbf{L_x} = \begin{pmatrix} -1/Z & 0 & x/Z & xy & -1-x^2 & y \\ 0 & -1/Z & y/Z & 1+y^2 & -xy & -x \end{pmatrix} \quad (5)$$

We can see that the dynamic of $\dot{x}$ and $\dot{y}$ with respect to the camera velocity components are really not the same: some are inversely proportional to the depth Z of the point, some are linearly dependent to the image coordinates, while others depend on them at second order. Even if we are able to design a control scheme such that the error in the image has an exponential decoupled decrease, the robot dynamics will be unlikely very far from such an exponential decoupled decrease, because of the strong non linearities in $\mathbf{L_x}$. The robot trajectory will thus be very far from the optimal one (typically, a straight line as for the translation and a geodesic as for the rotation). An important problem is thus to determine visual features such that they minimize the non linearities in the related interaction matrix. In this paper, following the recent work described in [11], three new visual features are given to control the translational dof. They are such that the related interaction matrix is diagonal and constant with respect to these dof.

In the remainder of the paper, we first briefly recall some definitions and important properties of moments. We then determine in Section 3 six visual features to control the six robot dof. The obtained control scheme is finally validated in Section 4 trough experimental results.

II. Moment invariants

A. Definitions

The 2D moments m_{pq} of order $p+q$ of the density function $f(x,y)$ are defined by:

$$m_{pq} \equiv \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} x^p y^q f(x,y)dxdy \quad (6)$$

The centred moments μ_{pq} are computed with respect to the object centroid (x_g, y_g). They are defined by:

$$\mu_{pq} = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} (x-x_g)^p (y-y_g)^q f(x,y)dxdy \quad (7)$$

where $x_g = \frac{m_{10}}{a}$ and $y_g = \frac{m_{01}}{a}$, $a = m_{00}$ being the object area.

The moments of a density function f exist if f is piecewise continuous and has nonzero values only in a finite region of the space. The centred moments are known to be invariant to 2D translational motion. The moment invariants to rotations are generally given in a polynomial form. In the literature, several works propose various methods to derive moment invariants. Reddi [16] obtained moment invariants to rotation using radial and angular moments, Teague [17], Belkassim [2], Walin and Kübler [18] derived Zernike moments invariant to rotation, Abu-Mustapha and Psaltis [1], Flusser [9] obtained invariants to rotation from complex moments. Finally, several formula have been proposed for invariants to scale, such as for example [13]:

$$I = \frac{m_{pq}}{m_{00}^{(p+q+2)/2}} \quad (8)$$

This normalization will be used in Section 3.1 to decouple the features involved in the control of the translational dof. More details about moment invariants can be found in Mukundan [14] and Prokop [15]. We just present some invariants to 2D rotations, to scale, and to 2D translations. They will be used in the design of the features involved to control the rotational velocities Ω_X and Ω_Y.

$$R_1 = \frac{I_{n_1}}{I_{n_2}},\ R_2 = \frac{I_{n_3}}{I_{n_2}},\ R_3 = \frac{I_{n_4}}{I_{n_6}},\ R_4 = \frac{I_{n_5}}{I_{n_6}} \quad (9)$$

where

$$\begin{aligned} I_{n_1} = {} & \mu_{30}^4 + 6\mu_{30}^3\mu_{12} + 9\mu_{30}^2\mu_{12}^2 + 6\mu_{30}^2\mu_{21}\mu_{03} \\ & + 2\mu_{30}^2\mu_{03}^2 + 18\mu_{30}\mu_{21}\mu_{12}\mu_{03} \\ & + 6\mu_{30}\mu_{12}\mu_{03}^2 + 9\mu_{21}^2\mu_{03}^2 + 6\mu_{21}\mu_{03}^3 + \mu_{03}^4 \end{aligned} \quad (10)$$

$$\begin{aligned} I_{n_2} = {} & 3\mu_{30}^2\mu_{12}^2 + 2\mu_{30}^2\mu_{03}^2 - 6\mu_{30}\mu_{21}^2\mu_{12} \\ & - 6\mu_{30}\mu_{21}\mu_{12}\mu_{03} + 2\mu_{30}\mu_{12}^3 + 3\mu_{21}^4 \\ & + 2\mu_{21}^3\mu_{03} + 3\mu_{21}^2\mu_{03}^2 - 6\mu_{21}\mu_{12}^2\mu_{03} + 3\mu_{12}^4 \end{aligned} \quad (11)$$

$$\begin{aligned} I_{n_3} = {} & -\mu_{30}^2\mu_{03}^2 + 6\mu_{30}\mu_{21}\mu_{12}\mu_{03} - 4\mu_{30}\mu_{12}^3 \\ & - 4\mu_{21}^3\mu_{03} + 3\mu_{21}^2\mu_{12}^2 \end{aligned} \quad (12)$$

$$I_{n_4} = (\mu_{50} + 2\mu_{32} + \mu_{14})^2 + (\mu_{05} + 2\mu_{23} + \mu_{41})^2 \quad (13)$$

$$I_{n_5} = (\mu_{50} - 2\mu_{32} - 3\mu_{14})^2 + (\mu_{05} - 2\mu_{23} - 3\mu_{41})^2 \quad (14)$$

$$I_{n_6} = (\mu_{50} - 10\mu_{32} + 5\mu_{14})^2 + (\mu_{05} - 10\mu_{23} + 5\mu_{41})^2 \quad (15)$$

B. Interaction matrix of 2D moments

In this paragraph, we recall from [4] the interaction matrix of the 2D moments. In the following we assume that the object belongs to a plane whose equation is given by:

$$\frac{1}{Z} = Ax + By + C$$

We also assume that the image is binary or that the grey level does not change when the camera moves (i.e. $\frac{df(x,y)}{dt} = 0$). In that case, the interaction matrix $\mathbf{L}_{m_{ij}}$ related to m_{ij} can be determined:

$$\mathbf{L}_{m_{ij}} = \begin{pmatrix} m_{vx} & m_{vy} & m_{vz} & m_{wx} & m_{wy} & m_{wz} \end{pmatrix} \quad (16)$$

where:

$$\begin{cases} m_{vx} = -i(Am_{ij} + Bm_{i-1,j+1} + Cm_{i-1,j}) - Am_{ij} \\ m_{vy} = -j(Am_{i+1,j-1} + Bm_{ij} + Cm_{i,j-1}) - Bm_{ij} \\ m_{vz} = (i+j+3)(Am_{i+1,j} + Bm_{i,j+1} + Cm_{ij}) - Cm_{ij} \\ m_{wx} = (i+j+3)m_{i,j+1} + jm_{i,j-1} \\ m_{wy} = -(i+j+3)m_{i+1,j} - im_{i-1,j} \\ m_{wz} = im_{i-1,j+1} - jm_{i+1,j-1} \cdots \end{cases}$$

Similarly, the interaction matrix related to the centred moments μ_{ij} is given by:

$$\mathbf{L}_{\mu_{ij}} = \begin{pmatrix} \mu_{vx} & \mu_{vy} & \mu_{vz} & \mu_{wx} & \mu_{wy} & \mu_{wz} \end{pmatrix} \quad (17)$$

with:

$$\begin{array}{ll} \mu_{vx} & = -(i+1)A\mu_{ij} - iB\mu_{i-1,j+1} \\ \mu_{vy} & = -jA\mu_{i+1,j-1} - (j+1)B\mu_{ij} \\ \mu_{vz} & = -A\mu_{wy} + B\mu_{wx} + (i+j+2)C\mu_{ij} \\ \mu_{wx} & = (i+j+3)\mu_{i,j+1} + ix_g\mu_{i-1,j+1} \\ & +(i+2j+3)y_g\mu_{ij} - in_{11}\mu_{i-1,j} - jn_{02}\mu_{i,j-1} \\ \mu_{wy} & = -(i+j+3)\mu_{i+1,j} - (2i+j+3)x_g\mu_{ij} \\ & -jy_g\mu_{i+1,j-1} + in_{20}\mu_{i-1,j} + jn_{11}\mu_{i,j-1} \\ \mu_{wz} & = i\mu_{i-1,j+1} - j\mu_{i+1,j-1} \end{array}$$

where $n_{ij} = 4\mu_{ij}/m_{00}$. For the positions where the object is parallel to the image plane (i.e. $A = B = 0$), we can check from the first two components of $\mathbf{L}_{\mu_{ij}}$ that the variation of the centred moments with respect to V_X and V_Y vanishes, which proves that these moments are invariant to 2D translations parallel to the image plane when (and only when) the object is also parallel to the image plane. For the same positions, it is easy to prove that the variation of the scale moment invariants with respect to V_Z vanishes. Even if the invariance to translation is local (i.e. only valid when $A = B = 0$), these features depend mainly on the rotation. We will thus use them to control the rotational dof.

III. Choice of the feature vector

In this section, we select from the previous theoretical results six features to control the six dof of the robot. Our objective is to obtain a sparse interaction matrix that changes slowly around the desired position of the camera. We will see that the solution we present is such that the interaction matrix is triangular when the object is parallel to the image plane. Furthermore, we will see that, for the same positions, the elements corresponding to translational motions form a constant diagonal block, which is independent of depth. In [4], this last interesting property was not satisfied.

We now assume that the desired position of the image plane and the object is parallel (i.e. $A = B = 0$) and we denote $\mathbf{L}_{\mathbf{s}}^{\|}$ the interaction matrix for such positions. In the following, we will only be concerned with $\mathbf{L}_{\mathbf{s}}^{\|}$ since it will be used to build the model $\widehat{\mathbf{L}_{\mathbf{s}}}$ of $\mathbf{L}_{\mathbf{s}}$ in the control scheme (4).

A. Features to control the translational dof

In [6], [4], the three visual features used to control the translational dof have been selected to be the coordinates x_g, y_g of the center of gravity and the area a of the object in the image. In that case, we obtain from (16):

$$\begin{array}{ll} \mathbf{L}_{x_g}^{\|} & = \begin{pmatrix} -C & 0 & Cx_g & \epsilon_1 & -(1+\epsilon_2) & y_g \end{pmatrix} \\ \mathbf{L}_{y_g}^{\|} & = \begin{pmatrix} 0 & -C & Cy_g & 1+\epsilon_3 & -\epsilon_1 & -x_g \end{pmatrix} \\ \mathbf{L}_{a}^{\|} & = \begin{pmatrix} 0 & 0 & 2aC & 3ay_g & -3ax_g & 0 \end{pmatrix} \end{array} \quad (18)$$

with $\epsilon_1 = n_{11} + x_g y_g$, $\epsilon_2 = n_{20} + x_g^2$ and $\epsilon_3 = n_{02} + y_g^2$. Even if the above matrix is triangular, we can note that its elements are strongly non linear. Moreover, the features do not have the same dynamic with respect to each translational dof.

Our choice is based on these intuitive features, but adding an adequate normalization. More precisely, we define:

$$a_n = Z^*\sqrt{\frac{a^*}{a}}, \quad x_n = a_n x_g, \quad y_n = a_n y_g \quad (19)$$

where a^* is the desired area of the object in the image, and Z^* the desired depth between the camera and the object. The interaction matrices related to these normalized features can be easily determined from (18). Noting that $Z^*\sqrt{a^*} = Z\sqrt{a} = \sqrt{S}$ where S is the area of the 3D object, we obtain:

$$\begin{array}{ll} \mathbf{L}_{x_n}^{\|} & = \begin{pmatrix} -1 & 0 & 0 & a_n\epsilon_{11} & -a_n(1+\epsilon_{12}) & y_n \end{pmatrix} \\ \mathbf{L}_{y_n}^{\|} & = \begin{pmatrix} 0 & -1 & 0 & a_n(1+\epsilon_{21}) & -a_n\epsilon_{11} & -x_n \end{pmatrix} \\ \mathbf{L}_{a_n}^{\|} & = \begin{pmatrix} 0 & 0 & -1 & -3y_n/2 & 3x_n/2 & 0 \end{pmatrix} \end{array} \quad (20)$$

with $\epsilon_{11} = n_{11} - x_g y_g/2$, $\epsilon_{12} = n_{20} - x_g^2/2$, and $\epsilon_{21} = n_{02} - y_g^2/2$. Since a_n is inversely proportional to $\sqrt{a}$, we find again the recent result given in [11] stating that the variation of such features depends linearly of the depth (note the constant term in the third element of $\mathbf{L}_{a_n}^{\|}$). The normalization by $Z^*\sqrt{a^*}$ has just be chosen so that this constant term is equal to -1. Furthermore, the design of x_n and y_n allows us to completely decouple the three selected features with respect to the translational dof. This property was expected from (8). We also obtain the same dynamics for the three features and the three translational dof (note the diagonal block equal to $-\mathbf{I}_3$ in (20)). This very nice property will allow us to obtain an adequate robot translational trajectory.

Finally, we can notice from the analytical and the numerical values of $\mathbf{L}_{x_n}^{\|}$ and $\mathbf{L}_{y_n}^{\|}$ (see (24)) the classical coupling between V_X and Ω_Y, and between V_Y and Ω_X. In fact, this natural coupling allows the object to remain as much as possible in the camera field of view.

B. Features to control the rotational dof

As in [6], [4], we use $\theta = \frac{1}{2}\arctan(\frac{2\mu_{11}}{\mu_{20}-\mu_{02}})$ defined as the orientation angle of the principal axis of inertia with the X-axis of the image frame. From Figure 1, we can notice that there are two solutions for θ: θ and $\theta + \pi$. However, the third order moments can be used to solve this ambiguity since a rotation of an object by π changes the sign of its third order moments.

We also use two moment invariants R_i and R_j chosen in (9). The related interaction matrices can be obtained from (17). We obtain (after tedious developments):

$$\begin{array}{ll} \mathbf{L}_{R_i}^{\|} & = \begin{pmatrix} 0 & 0 & 0 & R_{i_{wx}} & R_{i_{wy}} & 0 \end{pmatrix} \\ \mathbf{L}_{R_j}^{\|} & = \begin{pmatrix} 0 & 0 & 0 & R_{j_{wx}} & R_{j_{wy}} & 0 \end{pmatrix} \\ \mathbf{L}_{\theta}^{\|} & = \begin{pmatrix} 0 & 0 & 0 & \theta_{wx} & \theta_{wy} & -1 \end{pmatrix} \end{array} \quad (21)$$

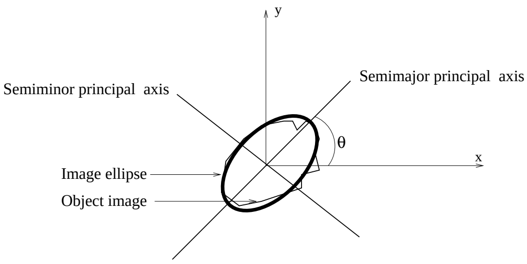


Fig. 1. Image ellipse

where the analytical form of the elements corresponding to Ω_X and Ω_Y can not be given here by lack of place (if interested, θ_{wx} and θ_{wy} can be found in [4]). As expected, we can notice the invariance of the selected features with respect to any 3D translational motion (remember that we consider here that $A = B = 0$), and the invariance of R_i and R_j with respect to Ω_Z. Finally, we can see from (24) that θ depends essentially on the rotation Ω_Z around the optical axis. As for R_i and R_j, they are chosen such that $\mathbf{L}^{\|}_{R_i}$ and $\mathbf{L}^{\|}_{R_j}$ are as orthogonal as possible. In the next section, we present experimental results where R_3 and R_4 have been selected. For the considered object, this choice has given the best results.

IV. Experimental results

This section presents some experimental results obtained with a six dof eye-in-hand system. The moments are computed at video rate after a simple binarisation of the aquired image, without any spatial segmentation. As already explained, we have used as visual features vector:

$$\mathbf{s} = \left(\begin{array}{cccccc} x_n & y_n & a_n & R_3 & R_4 & \theta \end{array} \right)^T \qquad (22)$$

In our experiments, the parameters of the object plane in the camera frame are given approximately for the desired position ($A = B = 0$, $C = 2$, which corresponds to $Z^* = 0.5$ m). They are not estimated at each step. For the two first experiments, a correct value of the camera intrinsic parameters has been used. The desired value $\mathbf{s}^*$ is given by:

$$\mathbf{s}^* = \left(\begin{array}{cccccc} \widehat{Z}^* x_g^* & \widehat{Z}^* y_g^* & \widehat{Z}^* & R_3^* & R_4^* & \theta^* \end{array} \right)^T \qquad (23)$$

where $x_g^*, y_g^*, R_3^*, R_4^*$ and θ^* are computed directly from the desired image (acquired during an off-line learning step), and where $\widehat{Z}^*$ has been set to 0.5 m. We can note from (23), (22) and (19) that using a wrong value $\widehat{Z}^*$ for Z^* has no influence on the convergence of the system ($\mathbf{s} = \mathbf{s}^*$ only for the desired position whatever the setting of value $\widehat{Z}^*$). It will just induce the same gain effect (with value $\widehat{Z}^*/Z^*$) for the decreasing of the three first features. An esperiment with a wrong setting of $\widehat{Z}^*$ is described in Section IV-C.

A. Pure translational motion

We first compare the results obtained with our features and those obtained using the centroid coordinates (x_g, y_g) and the area a for a pure translational motion between the initial and desired images (given on Figure 2.a and 2.b). For both schemes, we have used $\widehat{\mathbf{L}_\mathbf{s}} = \mathbf{L}_{\mathbf{s}|\mathbf{s}=\mathbf{s}^*} = \mathbf{L}^{\|}_{\mathbf{s}|\mathbf{s}=\mathbf{s}^*}$ in the control scheme (4) and gain λ has been set to 0.1.

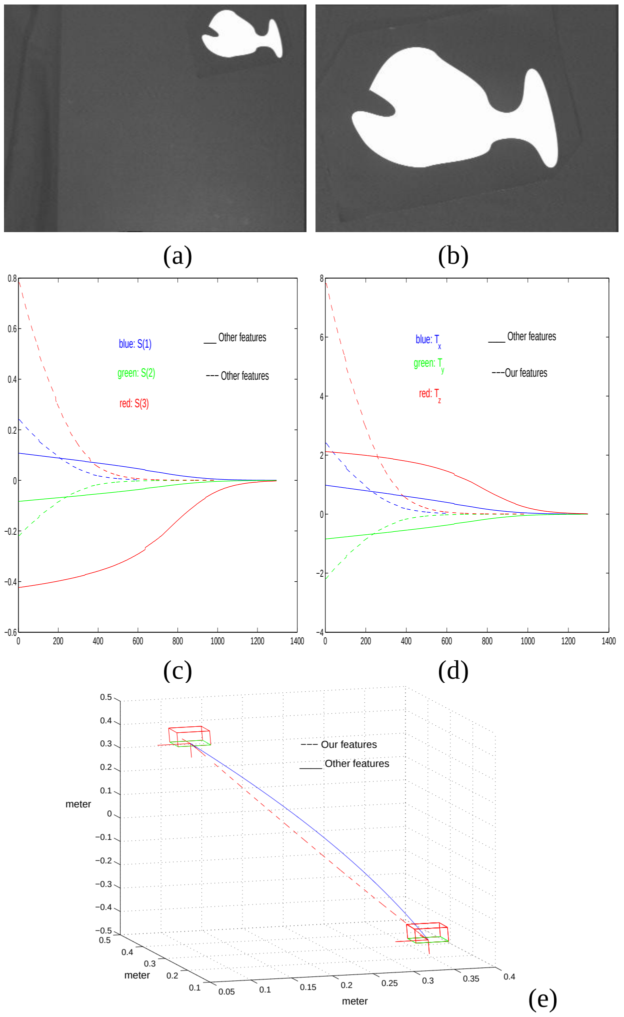

(a) (b) (c) (d) (e)

Fig. 2. Results for a pure translational motion: (a) initial image, (b) desired image, (c) visual features $(\mathbf{s} - \mathbf{s}^*)$, (d) camera velocity $\mathbf{T_c}$, (e) camera 3D trajectory

We can see on Figure 2 the improvements brought by the proposed features (in dashed lines) since they allow to obtain the same exponential decoupled decrease for the visual features and for the components of the camera velocity. As expected, the camera 3D trajectory is thus a pure straight line using the proposed features, while it is not using the other ones.

B. Complex motion

We now test our scheme for a displacement involving very large translation and rotation to realize between the initial and desired images (see Figures 3.a and 3.b). The interaction matrix computed at the desired position has the following form:

$$\mathbf{L}^{\|}_{\mathbf{s}|\mathbf{s}=\mathbf{s}^*} = \begin{pmatrix} -1 & 0 & 0 & 0.01 & -0.52 & 0.01 \\ 0 & -1 & 0 & 0.51 & -0.01 & 0.01 \\ 0 & 0 & -1 & -0.02 & -0.01 & 0 \\ 0 & 0 & 0 & -0.33 & -0.62 & 0 \\ 0 & 0 & 0 & -0.61 & 0.09 & 0 \\ 0 & 0 & 0 & -0.04 & -0.08 & -1 \end{pmatrix} \qquad (24)$$

We can note that this matrix is block triangular with main terms around the diagonal. The value of its condition number (equal to 2.60) is also very satisfactory. Finally, we have used the following model of $\mathbf{L_s}$ in the control scheme (4):

$$\widehat{\mathbf{L}}_{\mathbf{s}} = \frac{1}{2}(\mathbf{L}^{\|}_{\mathbf{s}} + \mathbf{L}^{\|}_{\mathbf{s}|\mathbf{s}=\mathbf{s}^*}) \qquad (25)$$

This choice has given the best experimental results. The obtained results are given on Figure 3. They show the good behavior of the control law. First, we can note the fast convergence towards the desired position (while the system does not converge for the six visual features proposed in [4]). Then, there is no oscillation in the decrease of the visual features (see Figure 3.c), and there is only one small oscillation for only two components of the camera velocity (see Figure 3.d). Finally, even if the rotation to realize between the initial and the desired positions is very large, the obtained camera 3D trajectory is satisfactory (see Figure 3.e), while it was an important drawback for classical 2D visual servoing.

C. Results with a bad camera calibration and object occultation

We now test the robustness of our approach with respect to a bad calibration of the system. In this experiment, errors have been added to camera intrinsic parameters (25% on the focal length and 20 pixels on the coordinates of the principal point) and to the object plane parameters ($\widehat{Z}^* = 0.8m$ instead of $Z^* = 0.5m$). We can also notice that the lighting conditions for the desired and the initial positions given on Figure 4.a and 4.b are different. Furthermore, an occultation has been generated since the object is not completely in the camera field of view at the begining of the servo. The obtained results are given in Figure 4. We can notice that the system converges despite the worse conditions of experimentations and, as soon as the occultation ends (after iteration 30), the behavior of the system is similar to those of the previous experiment, which validates the robustness of our scheme with respect to modeling errors.

V. Conclusion

In this paper, we have proposed a new visual servoing scheme based on the moments of an object. Our approach presents several advantages: there is no constraint on the object shape and the model of the object is also not required. Moment invariants have been used to decouple the camera dof, which allows the system to have a large convergence domain and a good behavior for the features in the image and for the camera trajectory. The experimental results show the validity of the approach and its robustness with respect to calibration errors. To improve further the obtained results, future works will be devoted to the development of an estimation scheme of the pose between the object and the camera.

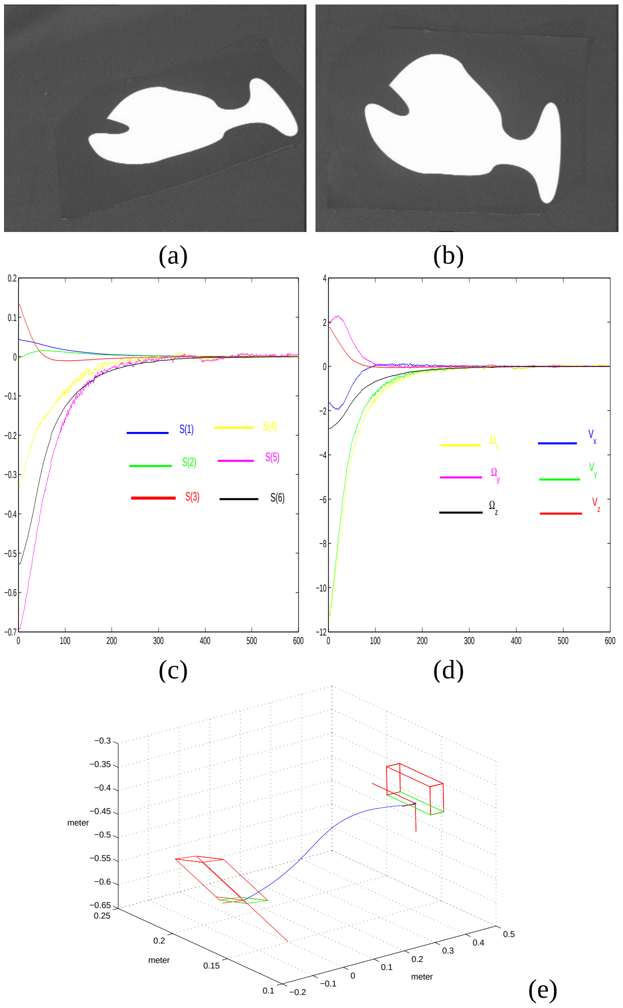


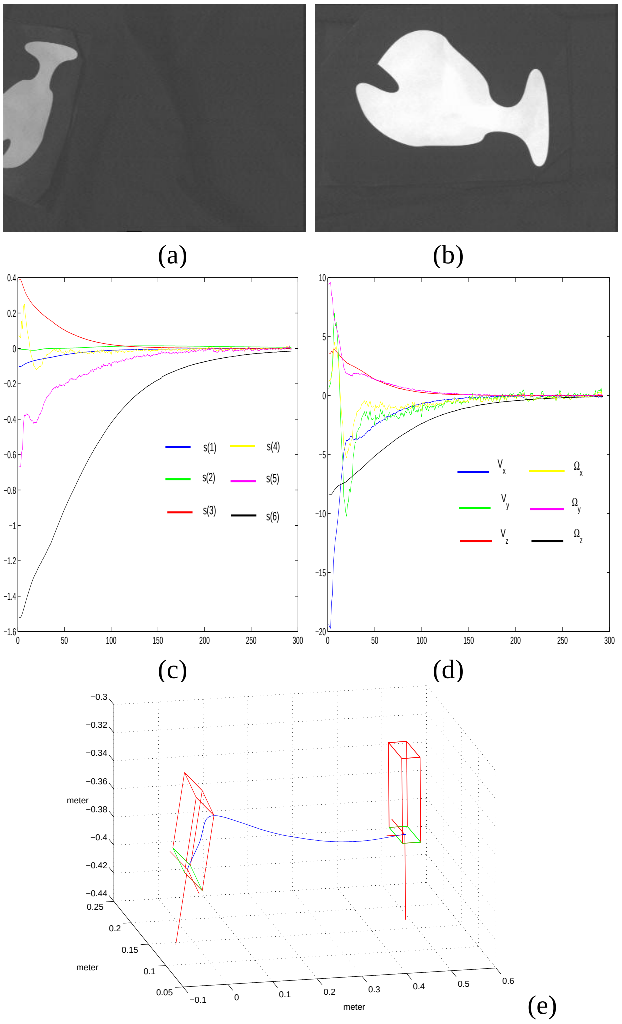
Fig. 3. Results for a complex motion: (a) initial image, (b) desired image, (c) visual features $(\mathbf{s} - \mathbf{s}^*)$, (d) camera velocity $\mathbf{T_c}$, (e) camera 3D trajectory

VI. References

[1] Y. S. Abu-Mustapha and D. Psaltis. Image normalisation by complex moments. *IEEE Trans. on PAMI*, 7(1):46–55, 1985.

[2] S. O. Belkassim, M. Shridhar, and M. Ahmadi. Shape-contour recognition using moment invariants. In *10th Int. Conf. on Pattern Recognition*, pages 649–651, Atlantic City, NJ, USA, June 1990.

[3] F. Chaumette. Potential problems of stability and convergence in image-based and position-based visual servoing. In A.S. Morse D. Kriegman, G. Hager, editor, *The Confluence of Vision and Control*, number 237, pages 66–78. Springer-Verlag, 1998. LNCIS.



(a) (b) (c) (d) (e)

Fig. 4. Results using a bad camera calibration: (a) initial image, (b)desired image (c) visual features $(\mathbf{s}-\mathbf{s}^*)$, (d) camera velocity $\mathbf{T_c}$, (e) camera 3D trajectory

[4] F. Chaumette. A first step toward visual servoing using image moments. In *IEEE/RSJ IROS'02*, pages 378–383, Lausanne, Switzerland, Oct. 2002.

[5] C. Collewet and F. Chaumette. A contour approach for image-based control of objects with complex shape. In *IEEE/RSJ IROS'00*, pages 751–756, Takamatsu, Japan, Nov. 2000.

[6] P. I. Corke and S. A. Hutchinson. A new partitioned approach to image-based visual servo control. *IEEE Trans. on Robotics and Automation*, 17(4):507–515, Aug. 2001.

[7] K. Deguchi. A direct interpretation of dynamic images with camera and object motions for vision guided robot control. *Int. Journal of Computer Vision*, 37(1):7–20, 2000.

[8] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. In *IEEE Trans. on Robotics and Automation*, volume 8, pages 313–326, June 1992.

[9] J. Flusser. On the independance of rotation moment invariants. *Pattern Recognition*, 33:1405–1410, 2000.

[10] S. Hutchinson, G. Hager, and P. Corke. A tutorial on visual servo control. *IEEE Trans. on Robotics and Automation*, 12(5):651–670, Oct. 1996.

[11] R. Mahony, P. Corke, and F. Chaumette. Choice of image features for depth-axis control in image-based visual servo control. In *IEEE/RSJ IROS'02*, pages 390–395, Lausanne, Switzerland, Oct. 2002.

[12] E. Malis, F. Chaumette, and S. Boudet. 2 1/2 d visual servoing. *IEEE Trans. on Robotics and Automation*, 15(2):238–250, Apr. 1999.

[13] A. G. Mamistvalov. n-dimensial moment invariants and conceptual theory of recognition n-dimensional solids. *IEEE Trans. on PAMI*, 20(8):819–831, Aug. 1998.

[14] R. Mukundan and K. R. Ramakrishnan. *Moment Functions in Image Analysis Theory and Application*. World Scientific Publishing Co.Pte.Ltd, 1998.

[15] R.J. Prokop and A. P. Reeves. A survey of moments based techniques for unoccluded object representation. *Graphical models and Image Processing*, 54(5):438–460, Sep. 1992.

[16] S. S. Reddi. Radial and angular moment invariants for image identification. *IEEE Trans. on PAMI*, 3(2):240–242, 1981.

[17] M.R. Teague. Image analysis via the general theory of moments. *Journal of Opt. Soc. of America*, 70:920–930, Aug. 1980.

[18] A. Walin and O. K*ü*bler. Complete sets of complex zernike moments invariants and the role of the pseudo-invariants. *IEEE Trans. on PAMI*, 17(11):1106–1110, Nov. 1995.

[19] W. Wilson, C. Hulls, and G. Bell. Relative end-effector control using cartesian position-based visual servoing. *IEEE Trans. on Robotics and Automation*, 12(5):684–696, Oct. 1996.