Determination of Moment Invariants and Their Application to Visual Servoing

Omar Tahri, François Chaumette

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

Determination of Moment Invariants and Their Application to Visual Servoing

Omar Tahri — François Chaumette

N° 4845

Juin 2003

THÈMES 4 et 3

ISSN 0249-6399 ISRN INRIA/RR--4845--FR+ENG

Determination of Moment Invariants and Their Application to Visual Servoing

Omar Tahri *, François Chaumette †

Thèmes 4 et 3 — Simulation et optimisation
de systèmes complexes — Interaction homme-machine,
images, données, connaissances
Projet Vista

Rapport de recherche n° 4845 — Juin 2003 — 20 pages

Abstract: Moment invariants are important shape descriptors in computer vision. In this paper, we give a general and systematic method to derive moment invariants of any order and for k-dimensional space (invariants to scale, to orthogonal transformation and to translation). This result is applied to the selection of adequate features for visual servoing. To elaborate the control law, we also determine the analytical form of the interaction matrix of any image moment. This matrix relates the variation of the image moments to the 3D motions of the camera. We finally presents some visual servoing experimental results using very simple images. These results allow to validate the theoretical developments proposed in this paper.

Key-words: Moment invariants, interaction matrix, visual servoing, decoupled control.

* Vista/IRISA
† Vista/IRISA

Moments invariants : méthodes de calcul et application à l'asservissement visuel

Résumé : Dans cet article, on présente une nouvelle méthode de calcul des moments invariants. Nous utiliserons ces invariants pour minimiser les non linéarités de la matrice d'interaction des primitives choisies et pour obtenir un asservissement visuel 2D découplé. Des résultats expérimentaux utilisant une caméra embarquée sur un robot à 6 ddl pour se positionner parallèlement à un objet plan de forme complexe sont présentés pour démontrer l'efficacité de la méthode proposée.

Mots-clés : Moments invariants, commande découplée, non linéarité

1 Introduction

The moment invariants theory takes an important place in computer vision. It was first introduced in 1962 by Hu [5], when he established the fundamental theorem of moment invariants (FTMI). Hu employed his theorem to derive seven bi-dimensional moment invariants. Since 1962, the moment invariants theory has been applied in several applications: pose estimation [4], [12], character recognition [21], quality inspection in industrial areas [18], matching [20], etc. In fact, the FTMI contains some mistakes that have been emphasized by Mamistvalov in [10] in 1970 (in russian). However the FTMI has been quoted in several works [16], [20], [8], [4].... Subsequently, Reiss established in 1991 [15] the revised fundamental theorem of moment invariants (RFTMI). The FTMI was generalized for the k-dimensional case and applied in multisensor fusion in [11]. However, the generalized theorem contains the same mistake as the theorem given by Hu. Finally, Mamistvalov in [9] gave the correct generalization of the RFTMI to k-dimensional solids. However, this method is not systematic since it does not allow to determine the analytical form of all moment invariants for a given order.

In this paper, we give a new general and systematic method to derive moment invariants under polynomial form and for k-dimensional space (invariants to scale, to orthogonal transformation and to translations). The main advantage of our method is that it is systematic and it allows to obtain the analytical form of all invariant polynomials of a given order, which was not the case using other methods.

If image moments have been widely used in computer vision, it would also be interesting to use them in visual servoing since they provide a generic representation of any object, with simple or complex shapes, that can be segmented in an image. Attempts of using moments in 2D visual servoing have already been presented in the past. The problem was that the analytical form of the interaction matrix related to image moments was not available. This matrix is however essential to design a visual servoing control scheme [6]. That is why, in [1], if the idea of using image moments was expressed rigorously, very coarse approximations were performed in practice to control only 4 dof of a robot using the area, the centroid and the main orientation of an object in the image. Similarly, in [19], a neural network has been developed to numerically estimate this interaction matrix. In this paper, a method to determine the analytical form of the interaction matrix related to any image moment is given. Another important objective of using moments in image-based visual servoing is to try to determine visual features that avoid the potential problems that may appear when redundant image points coordinates are used: local minimum, coupled features that leads to inadequate robot trajectories, etc [2]. A nice step in that way has been recently presented in [3] for a simple rectangular object. More precisely, we search for six independent visual features such that the corresponding interaction matrix has a maximal decoupled structure, without any singularity, and such that its condition number is as low as possible (to improve the robustness and the numerical stability of the system [13, 17]). Finally, we also would like to minimize the non linearities in the interaction matrix in order to obtain an adequate behavior of the system, as well as in the image space than in 3D space. Indeed, the goal is to obtain a robot trajectory as near as possible of the optimal one (typically, a straight line as

for the translation and a geodesic as for the rotation) using only image moments as inputs of the visual servo. To reach these objectives, we apply in this paper the results obtained for moments invariants to design adequate visual features.

In the following, we first recall some definitions and important properties of moments. A general and systematic method to compute moment invariants is also proposed. In Section 3, we determine the analytical form of the interaction matrix related to any moment of the image of planar object. We then determine in Section 4 six visual features to control the six robot dof. The obtained control scheme is finally validated in Section 5 trough experimental results.

2 Moment invariants

We first recall some basic definitions of moment functions. More details can be found in Mukundan [12] and Prokop [14]. Denoting $\mathbf{X} = (x_1, \ldots, x_k)$ the coordinates of a point in a k-dimensional space, the moments of the density function $f(\mathbf{X})$ are defined by:

$$m_{p_1 \ldots p_k} \equiv \int_{-\infty}^{+\infty} \ldots \int_{-\infty}^{+\infty} x_1^{p_1} \ldots x_k^{p_k} f(\mathbf{X}) d\mathbf{X} \tag{1}$$

where $p = p_1 + \ldots + p_k$ is the order of moment $m_{p_1 \ldots p_k}$. The moments of the density function $f(\mathbf{X})$ exist if $f(\mathbf{X})$ is piecewise continuous and has nonzero values only in a finite region of the space. Similarly, the centred moments are defined by:

$$\mu_{p_1 \ldots p_k} \equiv \int_{-\infty}^{+\infty} \ldots \int_{-\infty}^{+\infty} (x_1 - x_{1g})^{p_1} \ldots (x_k - x_{kg})^{p_k} f(\mathbf{X}) d\mathbf{X} \tag{2}$$

where $x_{1g} = \frac{m_{10\ldots0}}{m_{0\ldots0}}$, ..., $x_{kg} = \frac{m_{0\ldots01}}{m_{0\ldots0}}$ are the coordinates of the center of gravity of the object. It is well known that the centred moments are invariant to translational motions in their respective k-dimensional space. In the following we treat only the invariance to orthogonal transformations and to scale variations. If we want to consider also invariance onto translational motions, we just have to consider the centred moments in the following derivation. The method we propose is based on a variational approach, it allows to obtain a basis of moments invariants.

2.1 Invariants to orthogonal transformations

We consider that the object points undergoes the following orthogonal transformation:

$$\mathbf{X}' = \mathbf{R}\mathbf{X} \tag{3}$$

where $\mathbf{R}\mathbf{R}^T = \mathbf{I}_k$. After derivation of (3) we obtain:

$$\dot{\mathbf{X}} = \mathbf{A}\mathbf{X} \tag{4}$$

where $\mathbf{A} = \mathbf{R^T}\dot{\mathbf{R}}$ is an antisymmetric matrix. In a first step, we only consider the simple case where we seek invariant polynomials of order 1:

$$q(\mathbf{m}_p^1) = \boldsymbol{\alpha}^T \mathbf{m}_p^1 \tag{5}$$

where $\mathbf{m}_p^1 = (m_{p\ldots0}^1, \ldots, m_{0\ldots p}^1)$, $\boldsymbol{\alpha} = (\alpha_1, \ldots, \alpha_t)$ is a parameter vector to be determined, and $t = \dim(\mathbf{m}_p^1)$. The polynomial $q(\mathbf{m}_p^1)$ is invariant to $\mathbf{R}$, if and only if:

$$\dot{q}(\mathbf{m}_p^1) = \boldsymbol{\alpha}^T \dot{\mathbf{m}}_p^1 = 0 \ , \ \forall \mathbf{m}_p^1 \tag{6}$$

After derivation of (1) and using Green's theorem, we obtain:

$$\begin{aligned} \dot{m}_{p_1\ldots p_k} = & \sum_{i=1}^{k} \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} p_i x_i^{p_i-1} \dot{x}_i \prod_{j=1, j\neq i}^{k} x_j^{p_j} f(\mathbf{X}) d\mathbf{X} \\ & + \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} x_1^{p_1} \ldots x_k^{p_k} \dot{f}(\mathbf{X}) d\mathbf{X} \\ & + \int_{-\infty}^{+\infty} \cdots \int_{-\infty}^{+\infty} x_1^{p_1} \ldots x_k^{p_k} f(\mathbf{X}) (\sum_{i=1}^{k} \frac{\partial \dot{x}_i}{\partial x_i}) d\mathbf{X} \end{aligned} \tag{7}$$

Combining (7) with (4), we obtain:

$$\dot{m}_{p_1\ldots p_k} = \sum_{i=1}^{k} \sum_{j=1, j\neq i}^{k} p_i a_{ij} m_{p_1\ldots p_i-1\ldots p_j+1\ldots p_k} \tag{8}$$

where a_{ij} are the elements of matrix $\mathbf{A}$. To obtain this result, we assume that $\dot{f}(\mathbf{X}) = 0$. This hypothesis has also been made in [15] and in [9] to prove the RFTMI (i.e. the density function f does not change after transformation $\mathbf{R}$). Therefore the second term of (7) vanishes. Furthermore, since matrix $\mathbf{A}$ is antisymmetric, the third term of (7) also vanishes. From (8), we can see that the derivative of each element of $\mathbf{m}_p^1$ is a linear combination of the other elements of $\mathbf{m}_p^1$. Hence using (8), it is possible to compute some matrix $\mathbf{M}_1(\mathbf{A})$ such that equation (6) can be written:

$$\dot{q}(\mathbf{m}_p^1) = \boldsymbol{\alpha}^T \mathbf{M}_1(\mathbf{A}) \mathbf{m}_p^1 = 0 \tag{9}$$

Therfore, $\boldsymbol{\alpha}$ is the kernel of the matrix $\mathbf{M}_1(\mathbf{A})^T$. Now, we consider the general case where we seek an invariant polynomials of order n:

$$q(\mathbf{m}_p^n) = \boldsymbol{\alpha}^T \mathbf{m}_p^n \tag{10}$$

where $\mathbf{m}_p^n = (m_{p\ldots0}^n, m_{p\ldots0}^{n-1} m_{p-1,1\ldots0}^1, \ldots, m_{0\ldots p}^n)$. The polynomial $q(\mathbf{m}_p^n)$ is invariant to transformation (3) if and only if:

$$\dot{q}(\mathbf{m}_p^n) = \boldsymbol{\alpha}^T \dot{\mathbf{m}}_p^n = 0 \ , \ \forall \mathbf{m}_p^n \tag{11}$$

From (8) we note that the derivative of each element of $\mathbf{m}_p^n$ is a linear combination of the other elements of $\mathbf{m}_p^n$. Hence, it is possible to determine a matrix $\mathbf{M}_n(\mathbf{A})$ such that:

$$\dot{q}(\mathbf{m}_p^n) = \boldsymbol{\alpha}^T \mathbf{M}_n(\mathbf{A}) \mathbf{m}_p^n = 0 \tag{12}$$

Finally, if $\boldsymbol{\alpha}$ belongs to the kernel of matrix $\mathbf{M}_n^T(\mathbf{A})$ then $\boldsymbol{\alpha}^T \mathbf{m}_p^n$ is invariant to orthogonal transformation.

2.2 Examples

In 2-dimensional space ($k = 2$), we have

$$\mathbf{A} = \begin{bmatrix} 0 & -w \\ w & 0 \end{bmatrix}$$

where w is the rotational velocity of the object in its plane.
$*$ For polynomial of order 1 and moments of order 2, we define $\mathbf{m}_2^1 = (m_{20},\ m_{11},\ m_{02})$. Using (7), we obtain:

$$\begin{array}{rcl} \dot{m_{20}} & = & 2\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} x_1 \dot{x}_1 f(x_1,x_2)dx_1dx_2 \\ & = & -2w\int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} x_1 x_2 f(x_1,x_2)dx_1dx_2 \\ & = & -2wm_{11} \\ \dot{m_{11}} & = & \int_{-\infty}^{+\infty}\int_{-\infty}^{+\infty} (\dot{x}_1 x_2 + x_1 \dot{x}_2) f(x_1,x_2)dx_1dx_2 \\ & = & wm_{20} - wm_{02} \\ \dot{m_{02}} & = & 2wm_{11} \end{array}$$

We thus obtain:

$$\mathbf{M}_1(\mathbf{A}) = \begin{bmatrix} 0 & -2w & 0 \\ w & 0 & -w \\ 0 & 2w & 0 \end{bmatrix}$$

from witch we deduce $\boldsymbol{\alpha} = (1,\ 0,\ 1)$. We thus easily find again the well known invariant to 2D rotation:

$$q(\mathbf{m}_2^1) = m_{20} + m_{02} \tag{13}$$

$*$ For polynomial of order 2 and moments of order 2, we have $\mathbf{m}_2^2 = (m_{20}^2, m_{20}m_{11}, m_{20}m_{02}, m_{11}^2, m_{11}m_{02}, m_{02}^2)$. Using the method proposed, we obtain after very simple derivations:

$$\text{Ker } \mathbf{M}_2^T(\mathbf{A}) = \begin{bmatrix} 0 & 0 & 1 & -1 & 0 & 0 \\ 1 & 0 & 2 & 0 & 0 & 1 \end{bmatrix}^T$$

from which we deduce two independent invariants to rotation:

$$q(\mathbf{m}_2^2) = m_{20}m_{02} - m_{11}^2 \tag{14}$$

$$q(\mathbf{m}_2^2) = m_{20}^2 + 2m_{20}m_{02} + m_{02}^2 \tag{15}$$

Note that (15) is nothing but the square of (13) and is thus useless. Many other invariants can be obtained using the same way, either by considering moments of higher order, polynomials of higher order, or space of higher dimension ($k = 3$ for instance).

Finally, as already denoted, if the centred moments are used in the previous results, the polynomials obtained are also invariant to translations.

2.3 Invariants to scale

In pattern recognition, several applications use a shape classification of object (square, spherical, cubic, etc.). This kind of applications requires a scale invariant classifiers. If we use the moments functions as descriptors and if the scale transformation is given by an uniform scale factor λ, it is very easy to make the polynomials invariant to translations and rotations also invariants to scale. Indeed, if each component of $\mathbf{X} = (x_1, \ldots, x_k)$ is multiplied by the scale factor λ, the polynomial $q(\mathbf{m}_p^n)$ is multiplied by $\lambda^{n(p+k)}$ and $m_{0\ldots0}$ is multiplied by λ^k. Hence, the ratio of $q(\mathbf{m}_p^n)$ by $m_{0\ldots0}^{n(\frac{p}{k}+1)}$ is invariant to scale. We now consider the general case. We remind that k is the space dimension, p is the moments order and n is the invariant polynomial order. For general scale transformation:

$$\mathbf{X}' = \mathbf{D}\mathbf{X} \tag{16}$$

where $\mathbf{D} = \mathrm{diag}(\lambda_1, \ldots, \lambda_k)$, Mamistvalov proposed in [9]:

$$I = \frac{m_{\frac{p}{k}\ldots\frac{p}{k}}}{m_{0\ldots0}^{\frac{p}{k}+1}}$$

as an invariant to scale. We propose a more general form of scale invariants:

$$I = \frac{\prod_{i=1}^{k} m_{p_1^i\ldots p_k^i}^{n_i}}{m_{0\ldots0}^{n(\frac{p}{k}+1)}} \tag{17}$$

where $\sum_{i=1}^{k}(p_j^i + 1)n_i = n(\frac{p}{k} + 1)\ \forall j$, $\sum_{j=1}^{k} p_j^i = p\ \forall i$, and $\sum_{i=1}^{k} n_i = n$. Indeed, using the moment definition, we obtain after transformation (16) $m_{p_1^i\ldots p_k^i}^{n_i}$ multiplied by $\prod_{j=1}^{k} \lambda_j^{(p_j^i+1)n_i}$, hence $\prod_{i=1}^{k} m_{p_1^i\ldots p_k^i}^{n_i}$ is multiplied by $\prod_{j=1}^{k} \lambda_j^{\sum_{i=1}^{k}(p_j^i+1)n_i}$. Since $m_{0\ldots0}^{n(\frac{p}{k}+1)}$ is multiplied by $\prod_{j=1}^{k} \lambda_j^{n(\frac{p}{k}+1)}$, (17) is thus invariant to (16).

We can note that the scale invariants given in [9] are a particular case of (17) (i.e. valid for $n = 1$ and $i = 1$). Using our formula, we obtain the following examples of scale invariants (for $k = 2$):

$$\begin{aligned} &I_1 = \frac{m_{11}}{m_{00}^2},\ I_2 = \frac{m_{02}m_{20}}{m_{00}^4},\ I_3 = \frac{m_{11}^2}{m_{00}^4},\ I_4 = \frac{m_{30}m_{03}}{m_{00}^5}, \\ &I_5 = \frac{m_{21}m_{12}}{m_{00}^5},\ I_6 = \frac{m_{22}}{m_{00}^3},\ I_7 = \frac{m_{40}m_{04}}{m_{00}^6} \end{aligned} \tag{18}$$

Other scale invariants can be derived by adding or multiplying the scale invariants of the above form. That property will be used to derive invariants to rotation, translation, and scale.

2.4 Invariants to rotation and scale

In Section 2.1, we have derived polynomials invariant to orthogonal transformation, whose form is given by:

$$q(\mathbf{m}_p^n) = \boldsymbol{\alpha}\mathbf{m}_p^n \text{ where } \alpha \in \text{Ker } \mathbf{M}_n^T(\mathbf{A})$$

Each element of $\mathbf{m}_p^n$ can be written: $\prod_{i=1}^{k} m_{p_1^i \ldots p_k^i}^{n_i}$ where $\sum_{j=1}^{k} p_j^i = p$ and $\sum_{i=1}^{k} n_i = n$. We note α_i the i^{th} element of $\boldsymbol{\alpha}$ and m_i the i^{th} element of $\mathbf{m}_p^n$. If $\forall i$ such that $\alpha_i \neq 0$, $\frac{m_p^n(s)}{m_{0\ldots0}^{l(\frac{p}{k}+1)}}$ is invariant to scal, then the ratio:

$$I_t = \frac{\boldsymbol{\alpha}\mathbf{m}_p^n}{m_{0\ldots0}^{l(\frac{p}{k}+1)}} \tag{19}$$

is invariant to rotation and to scale. Indeed I_t is the ratio of two invariants to rotation and is also a linear combination of scale invariants given by (17). For example:

$$I_{t_1} = \frac{m_{20}m_{02} - m_{11}^2}{m_{00}^4} \tag{20}$$

is invariant to scale and to rotations. Finally, using the centered moments, we can obtain forms that are also invariant to translation. As an example, the two following ratios that will be used in our visual servoing scheme are invariant to rotation, translation and scale:

$$r_1 = \frac{I_{n_1}}{I_{n_3}},\ r_2 = \frac{I_{n_2}}{I_{n_3}} \tag{21}$$

$$\text{where } \begin{cases} I_{n_1} = (\mu_{50} + 2\mu_{32} + \mu_{14})^2 + (\mu_{05} + 2\mu_{23} + \mu_{41})^2 \\ I_{n_2} = (\mu_{50} - 2\mu_{32} - 3\mu_{14})^2 + (\mu_{05} - 2\mu_{23} - 3\mu_{41})^2 \\ I_{n_3} = (\mu_{50} - 10\mu_{32} + 5\mu_{14})^2 + (\mu_{05} - 10\mu_{23} + 5\mu_{41})^2 \end{cases}$$

We now apply the previous result to image-based visual servoing. For that, we first have to derive the interaction matrix related to the moments. This matrix links the variation of the moments observed in the image (after a perspective projection) to the kinematic screw between the camera and the considered object. As will be seen in Section 4, it plays an essential role in the design and in the behavior of the control law.

3 Interaction matrix of 2D moments

We recall that the image moments m_{pq} of order $p+q$ are defined by the formula:

$$m_{pq} \equiv \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} h(x,y)dxdy \tag{22}$$

where $h(x,y) = x^p y^q f(x,y)$, $f(x,y)$ being the intensity level of image point with coordinates $\mathbf{x} = (x,y)$. We are interested in determining an analytical form describing the time variation $\dot{m}_{pq}$ of moment m_{pq} in function of the relative kinematic screw $\mathbf{v} = (\boldsymbol{\upsilon}, \boldsymbol{\omega})$ between the camera and the object, where $\boldsymbol{\upsilon} = (\upsilon_x, \upsilon_y, \upsilon_z)$ and $\boldsymbol{\omega} = (\omega_x, \omega_y, \omega_z)$ represent the translational and rotational velocity components respectively. As for classical geometrical features [6], we will obtain a linear link that can be expressed under the form:

$$\dot{m}_{pq} = \mathbf{L}_{m_{pq}}\ \mathbf{v} \tag{23}$$

where $\mathbf{L}_{m_{pq}}$ is called the interaction matrix related to m_{pq}. By developing (7) for 2D moments (where $k=2$), we obtain:

$$\dot{m}_{pq} = \iint_{\mathcal{D}} [\frac{\partial h}{\partial x}\dot{x} + \frac{\partial h}{\partial y}\dot{y} + h(x,y)(\frac{\partial \dot{x}}{\partial x} + \frac{\partial \dot{y}}{\partial y})]dxdy \tag{24}$$

In this expression, the terms $\dot{x}$, $\dot{y}$, $\frac{\partial \dot{x}}{\partial x}$ and $\frac{\partial \dot{y}}{\partial y}$ can be linearly expressed to the kinematic screw $\mathbf{v}$. Indeed, for any point $\mathbf{x}$ in the image whose corresponding 3D point has depth Z, we have the well known relation [6]:

$$\dot{\mathbf{x}} = \mathbf{L}_{\mathbf{x}}\ \mathbf{v} \tag{25}$$

where:

$$\mathbf{L}_{\mathbf{x}} = \begin{bmatrix} -1/Z & 0 & x/Z & xy & -1-x^2 & y \\ 0 & -1/Z & y/Z & 1+y^2 & -xy & -x \end{bmatrix} \tag{26}$$

If we consider a planar object, whose equation expressed in the camera frame is given by $Z = \alpha X + \beta Y + Z_0$, we have (since $x = X/Z$ and $y = Y/Z$):

$$\frac{1}{Z} = Ax + By + C \tag{27}$$

where $A = -\alpha/Z_0, B = -\beta/Z_0$ and $C = 1/Z_0$. Using (27) in (26), (25) can finally be written:

$$\begin{cases} \dot{x} = -(Ax+By+C)\upsilon_x + x(Ax+By+C)\upsilon_z \\ \quad + xy\omega_x - (1+x^2)\omega_y + y\omega_z \\ \dot{y} = -(Ax+By+C)\upsilon_y + y(Ax+By+C)\upsilon_z \\ \quad + (1+y^2)\omega_x - xy\omega_y - x\omega_z \end{cases} \tag{28}$$

from which we deduce:

$$\begin{cases} \frac{\partial \dot{x}}{\partial x} = -Av_x + (2Ax + By + C)v_z + y\omega_x - 2x\omega_y \\ \frac{\partial \dot{y}}{\partial y} = -Bv_y + (Ax + 2By + C)v_z + 2y\omega_x - x\omega_y \end{cases} \tag{29}$$

Substituting (28) and (29) in (24), and knowing that $\frac{\partial h}{\partial x} = px^{p-1}y^q f(x,y)$ and $\frac{\partial h}{\partial y} = qx^p y^{q-1} f(x,y)$, we can express (24) under the expected form (23). We obtain after simple developments:

$$\mathbf{L}_{m_{pq}} = \begin{bmatrix} m_{vx} & m_{vy} & m_{vz} & m_{wx} & m_{wy} & m_{wz} \end{bmatrix} \tag{30}$$

where:

$$\begin{cases} m_{vx} = -p(Am_{pq} + Bm_{p-1,q+1} + Cm_{p-1,q}) - Am_{pq} \\ m_{vy} = -q(Am_{p+1,q-1} + Bm_{pq} + Cm_{p,q-1}) - Bm_{pq} \\ m_{vz} = (p+q+3)(Am_{p+1,q} + Bm_{p,q+1} + Cm_{pq}) - Cm_{pq} \\ m_{wx} = (p+q+3)m_{p,q+1} + qm_{p,q-1} \\ m_{wy} = -(p+q+3)m_{p+1,q} - pm_{p-1,q} \\ m_{wz} = pm_{p-1,q+1} - qm_{p+1,q-1} \end{cases}$$

The time variation of a moment of order $p+q$ can thus be expressed from the moments of order less than $p+q+2$ and from the 3D parameters A, B and C. Similarly, the interaction matrix related to the centred moments μ_{pq} can be obtained (after tedious developments):

$$\mathbf{L}_{\mu_{pq}} = \begin{bmatrix} \mu_{vx} & \mu_{vy} & \mu_{vz} & \mu_{wx} & \mu_{wy} & \mu_{wz} \end{bmatrix} \tag{31}$$

with:

$$\begin{aligned} \mu_{vx} &= -(p+1)A\mu_{pq} - pB\mu_{p-1,q+1} \\ \mu_{vy} &= -qA\mu_{p+1,q-1} - (q+1)B\mu_{pq} \\ \mu_{vz} &= -A\mu_{wy} + B\mu_{wx} + (p+q+2)C\mu_{pq} \\ \mu_{wx} &= (p+q+3)\mu_{p,q+1} + px_g\mu_{p-1,q+1} \\ &\quad +(p+2q+3)y_g\mu_{pq} - pn_{11}\mu_{p-1,q} - qn_{02}\mu_{p,q-1} \\ \mu_{wy} &= -(p+q+3)\mu_{p+1,q} - (2p+q+3)x_g\mu_{pq} \\ &\quad -qy_g\mu_{p+1,q-1} + pn_{20}\mu_{p-1,q} + qn_{11}\mu_{p,q-1} \\ \mu_{wz} &= p\mu_{p-1,q+1} - q\mu_{p+1,q-1} \end{aligned}$$

where $n_{pq} = 4\mu_{pq}/m_{00}$. 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 $\mathbf{L}_{\mu_{pq}}$ that the variation of the centred moments with respect to v_x and v_y vanishes, which proves that the centred 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 check that the variation of the invariants to 2D rotations given in Section 2.2 vanishes for any motion ω_z, and that the variation of the invariants to scale given in Section 2.3 vanishes for

any translational motion v_z. Because of the perspective transformation, we can note that these invariances are local (i.e. only valid when $A = B = 0$), but we will see that a good behavior of the control scheme is obtained using such invariants even if the object plane is not parallel to the image plane.

4 Application to visual servoing

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. This property had never been obtained yet in image-based visual servoing.

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

$$\mathbf{T} = -\lambda \, \widehat{\mathbf{L}_\mathbf{s}}^{+} (\mathbf{s} - \mathbf{s}^*) \tag{32}$$

where λ is a proportional gain, $\mathbf{s}$ is the current value of the selected visual features, and $\mathbf{s}^*$ their desired value.

4.1 Features to control the translational dof

In [3] where a simple rectangular object was considered, 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 of the object in the image ($x_g = m_{10}/m_{00}, y_g = m_{01}/m_{00}$), and its area $a(= m_{00})$. In that case, we obtain from (30):

$$\begin{array}{rl}
\mathbf{L}_{x_g}^{\parallel} & = [\, -C \quad 0 \quad Cx_g \quad \epsilon_1 \quad -(1+\epsilon_2) \quad y_g \,] \\
\mathbf{L}_{y_g}^{\parallel} & = [\, 0 \quad -C \quad Cy_g \quad 1+\epsilon_3 \quad -\epsilon_1 \quad -x_g \,] \\
\mathbf{L}_{a}^{\parallel} & = [\, 0 \quad 0 \quad 2aC \quad 3ay_g \quad -3ax_g \quad 0 \,]
\end{array} \tag{33}$$

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 and the results for invariance to scale. More precisely, we define:

$$a_n = Z^* \sqrt{\frac{a^*}{a}}, \quad x_n = a_n x_g, \quad y_n = a_n y_g \tag{34}$$

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 (33). Noting that $Z^*\sqrt{a^*} = Z\sqrt{a} = \sqrt{S}$ where S is the area of the 3D object, we obtain:

$$\begin{array}{l} \mathbf{L}_{x_n}^{\parallel} = [-1 \;\; 0 \;\; 0 \;\;\; a_n\epsilon_{11} \;\;\; -a_n(1+\epsilon_{12}) \;\; y_n \;] \\ \mathbf{L}_{y_n}^{\parallel} = [\; 0 \;\; -1 \;\; 0 \;\; a_n(1+\epsilon_{21}) \;\;\; -a_n\epsilon_{11} \;\;\; -x_n] \\ \mathbf{L}_{a_n}^{\parallel} = [\; 0 \;\;\; 0 \;\; -1 \;\; -3y_n/2 \;\;\; 3x_n/2 \;\;\; 0 \;\;] \end{array} \tag{35}$$

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 [7] stating that the variation of such features depends linearly of the depth (note the constant term in the third element of $\mathbf{L}_{a_n}^{\parallel}$). 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 (so that they are invariant to scale) allows us to completely decouple the three selected features with respect to the translational dof. 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 (35)). 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 $\mathbf{L}_{x_n}^{\parallel}$ and $\mathbf{L}_{y_n}^{\parallel}$ (see (39)) 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.

4.2 Features to control the rotational dof

As in [3], we use the classical object orientation $\alpha = \frac{1}{2}\arctan(\frac{2\mu_{11}}{\mu_{20}-\mu_{02}})$ to control the rotation around the optical axis ω_z. For the two remaining dof ω_x and ω_y (that are really more difficult to control), we propose to use the two moment invariants r_1 and r_2 given in (21). The related interaction matrices can be obtained from (31). We obtain (after tedious developments):

$$\begin{array}{lll} \mathbf{L}_{r_1}^{\parallel} & = [\;\; 0 \;\; 0 \;\; 0 \;\; r_{1_{wx}} \;\;\; r_{1_{wy}} \;\;\; 0 \;\;] \\ \mathbf{L}_{r_2}^{\parallel} & = [\;\; 0 \;\; 0 \;\; 0 \;\; r_{2_{wx}} \;\;\; r_{2_{wy}} \;\;\; 0 \;\;] \\ \mathbf{L}_{\alpha}^{\parallel} & = [\;\; 0 \;\; 0 \;\; 0 \;\; \alpha_{wx} \;\;\; \alpha_{wy} \;\;\; -1 \;\;] \end{array} \tag{36}$$

where the analytical form of the elements corresponding to ω_x and ω_y can not be given here by lack of place 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_1 and r_2 with respect to ω_z. As expected, we can also note the linear link between the variation of α and ω_z. As for r_1 and r_2, they have been chosen from all the invariants to translation, rotation, and scale, such that $\mathbf{L}_{r_1}^{\parallel}$ and $\mathbf{L}_{r_2}^{\parallel}$ are as orthogonal as possible in order to decouple as much as possible ω_x and ω_y.

5 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 binarization of the acquired images, without any spatial segmentation. As already explained, we have used as visual features:

$$\mathbf{s} = (x_n, y_n, a_n, r_1, r_2, \alpha) \tag{37}$$

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:

$$\mathbf{s}^* = \left(\widehat{Z}^* x_g^*, \widehat{Z}^* y_g^*, \widehat{Z}^*, r_1^*, r_2^*, \alpha^*\right) \tag{38}$$

where $x_g^*, y_g^*, r_1^*, r_2^*$ and α^* 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 (38), (37) and (34) 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 experiment with a wrong setting of $\widehat{Z^*}$ is described at the end of this section.

5.1 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 1.a and 1.b). For both schemes, we have used $\widehat{\mathbf{L}_\mathbf{s}} = \mathbf{L}_{\mathbf{s}|\mathbf{s}=\mathbf{s}^*} = \mathbf{L}^{\parallel}_{\mathbf{s}|\mathbf{s}=\mathbf{s}^*}$ in the control scheme (32) and gain λ has been set to 0.1. We can see on Figure 1 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.

5.2 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 2.a and 2.b). The interaction

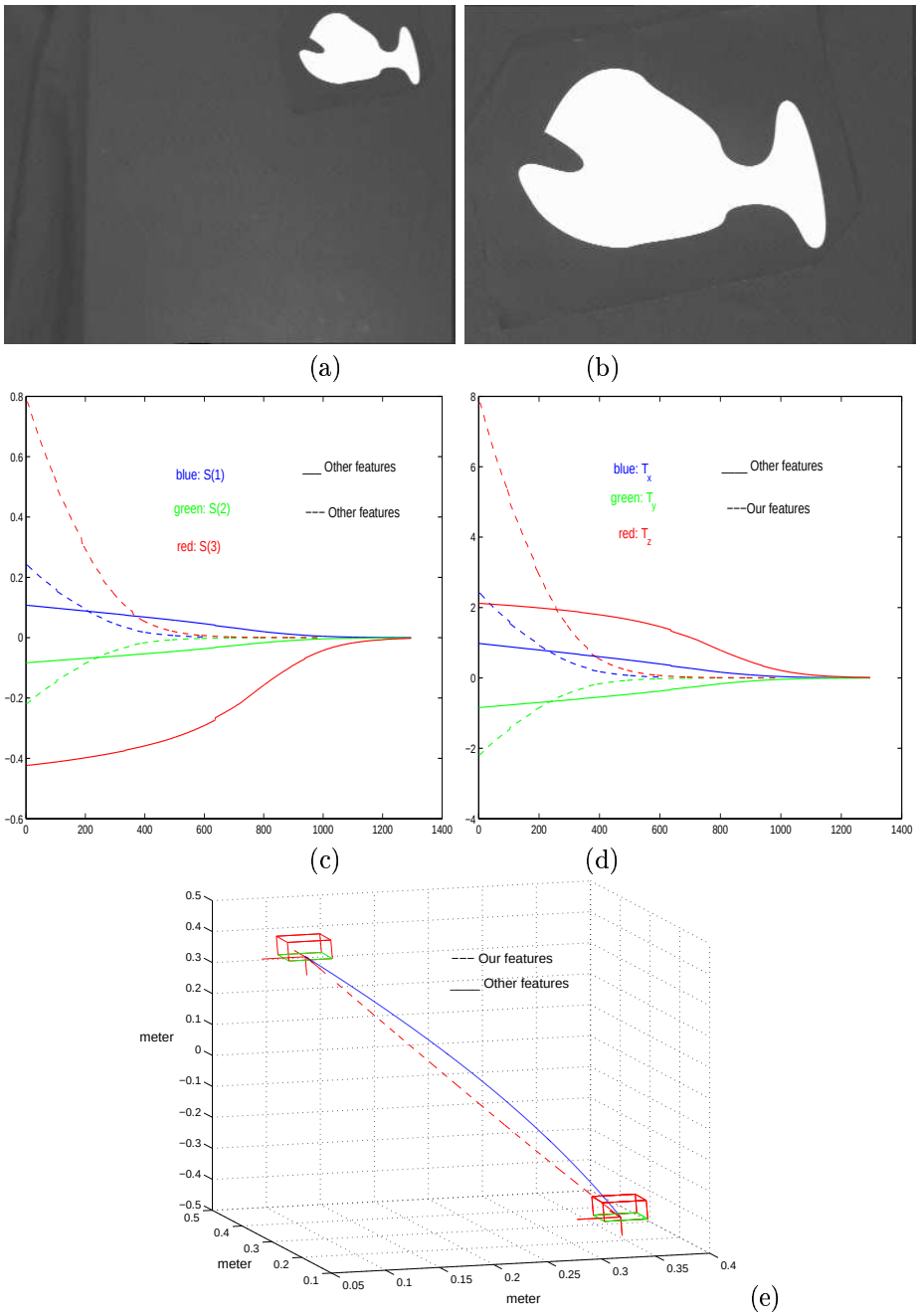

Figure 1: Results for a pure translational motion: (a) initial image, (b) desired image, (c) visual features $(\mathbf{s} - \mathbf{s}^*)$, (d) camera velocity $\mathbf{v}$, (e) camera 3D trajectory

matrix computed at the desired position has the following form:

$$\mathbf{L}_{\mathbf{s}|\mathbf{s}=\mathbf{s}^*}^{\parallel}=\begin{bmatrix} -1 & 0 & 0 & 0.01 & -0.52 & 0.01 \\ 0 & -1 & 0 & 0.51 & -0.01 & 0.01 \\ 0 & 0 & -1 & -0.02 & -0.01 & 0 \\ 0 & 0 & 0 & -0.33 & -0.62 & 0 \\ 0 & 0 & 0 & -0.61 & 0.09 & 0 \\ 0 & 0 & 0 & -0.04 & -0.08 & -1 \end{bmatrix} \tag{39}$$

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 $\mathbf{L}_\mathbf{s}$ in the control scheme (32):

$$\widehat{\mathbf{L}}_\mathbf{s} = \frac{1}{2}(\mathbf{L}_\mathbf{s}^{\parallel} + \mathbf{L}_{\mathbf{s}|\mathbf{s}=\mathbf{s}^*}^{\parallel}) \tag{40}$$

This choice has given the best experimental results. The obtained results are given on Figure 2. They show the good behavior of the control law. First, we can note the fast convergence towards the desired position. Then, there is no oscillation in the decrease of the visual features (see Figure 2.c), and there is only one small oscillation for only two components of the camera velocity (see Figure 2.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 2.e), while it was an important drawback for classical 2D visual servoing.

5.3 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 $Z^* = 0.5m$). Furthermore, an occultation has been generated since the object is not completely in the camera field of view at the beginning of the servo. The obtained results are given in Figure 3. 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.

6 Conclusion

In this paper, a new general and systematic method to compute moments invariants to scale and rotation has been proposed. These results have been applied to design a new visual servoing scheme able to handle planar objects with complex and unknown shapes. Moment invariants have been used to decouple the camera dof, which allows the system to have a

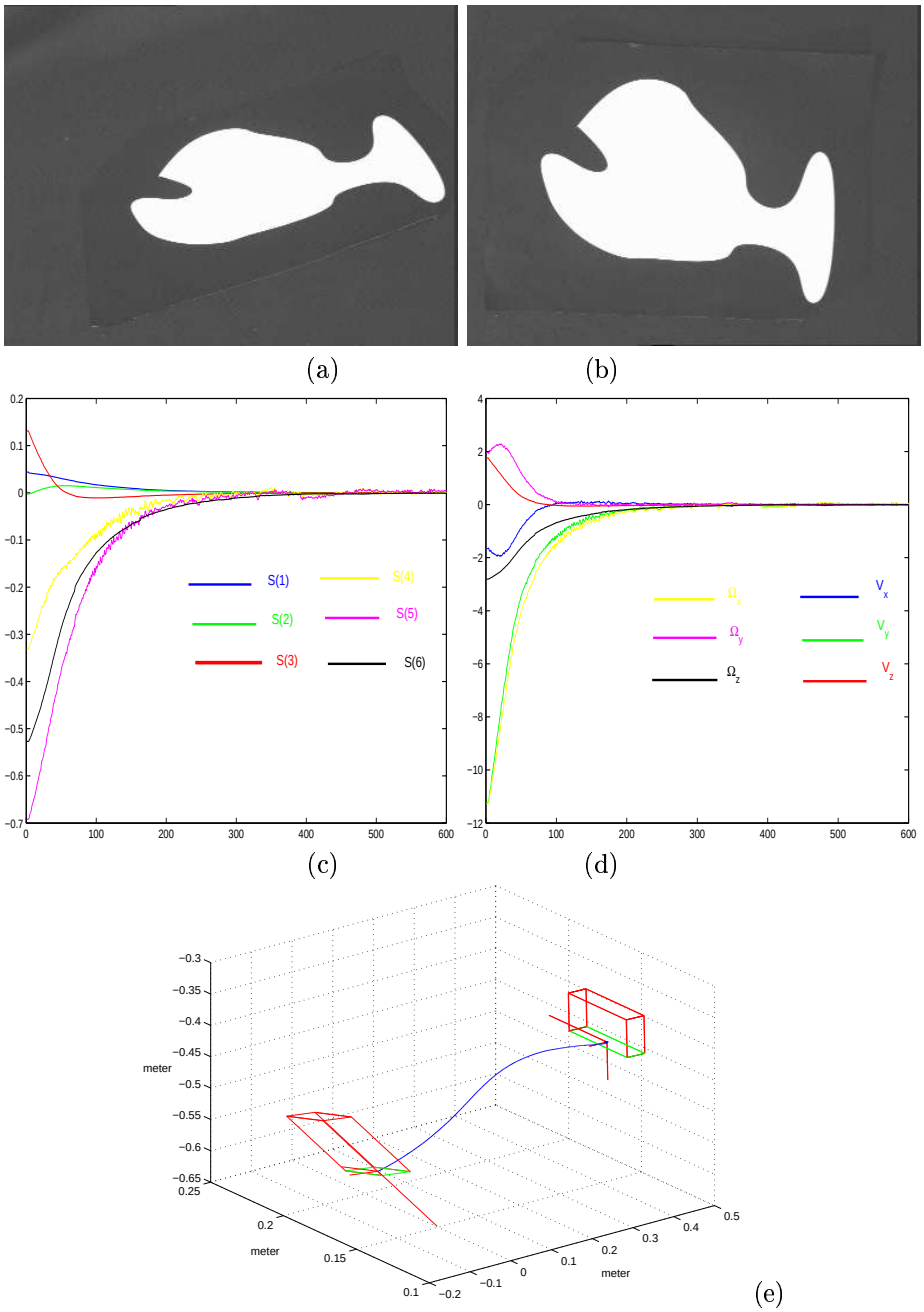

Figure 2: Results for a complex motion: (a) initial image, (b) desired image, (c) visual features ($\mathbf{s} - \mathbf{s}^*$), (d) camera velocity $\mathbf{v}$, (e) camera 3D trajectory

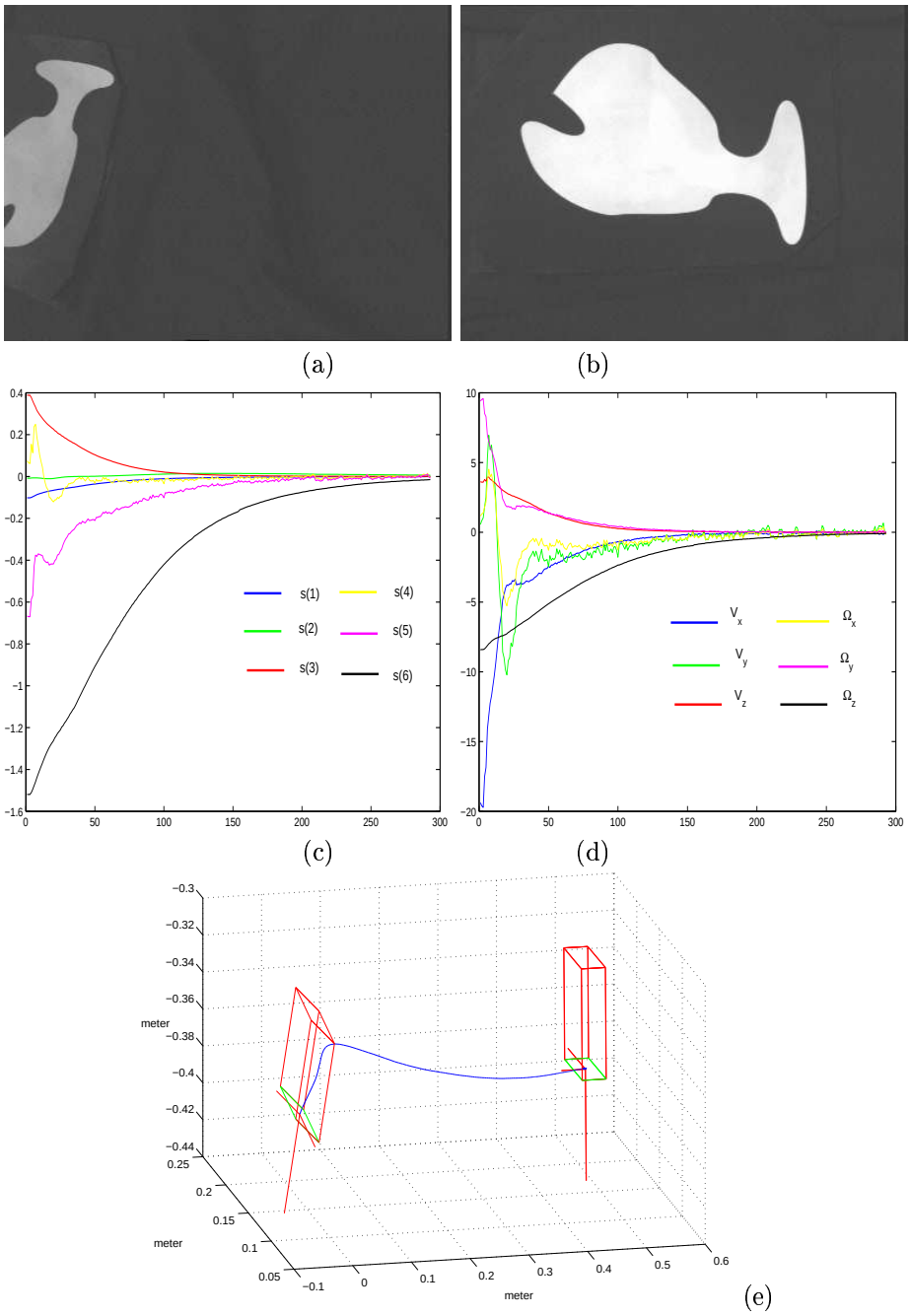

Figure 3: Results using a bad camera calibration: (a) initial image, (b) desired image, (c) visual features ($\mathbf{s} - \mathbf{s}^*$), (d) camera velocity $\mathbf{v}$, (e) camera 3D trajectory

large convergence domain and a good behavior for the features in the image and for the robot trajectory. The experimental results show the validity of the approach and its robustness with respect to calibration errors. Future works will be devoted to apply the results obtained on moments invariants to the classical pose problem.

References

[1] Z. Bien, W. Jang, and J. Park. Characterization and use of feature-jacobian matrix for visual servoing. In K. Hashimoto ed., editor, *Visual Servoing*, pages 317–363. World Scientific Publishing Co.Pte.Ltd, 1993.

[2] 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.

[3] 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.

[4] D. Cyganski and J. A. Orr. Application of tensor theory to object recognition and orientation determination. *IEEE Trans. on PAMI*, 7:662–673, Nov. 1985.

[5] M-K Hu. Visual pattern recognition by moments invariants. *IRE Trans. on Information Theory*, 8:179–187, Feb 1962.

[6] 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.

[7] 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.

[8] S. Maitra. Moment invariants. In *Proc IEEE*, volume 67, pages 697–699, Apr. 1979.

[9] 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.

[10] A.G. Mamistvalov. On the fundamental theorem of moment invariants. *Bull.Acade. Sciences Georgian SSR*, 59(2):297–300, Aug 1970. in Russian.

[11] V. Markandey and R. J. P. deFigueiredo. Robot sensing techniques based on high-dimensional moment invariants and tensors. *IEEE Trans. on Robotics and Automation*, 8(2):186–195, Apr. 1992.

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

[13] B. Nelson and P. Khosla. The resolvability ellipsoid for visual servoing. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition, CVPR'94*, pages 829–832, Seattle, Washington, June 1994.

[14] 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.

[15] T. H. Reiss. The revised fundamental theorem of moment invariants. *IEEE Trans. on PAMI*, 13(8):830–834, August 1991.

[16] S.A.Dudani, K.J. Breeding, and R.B. MCGhee. Aircraft identification by moments invariants. *IEEE Trans.Comput*, C26:39–46, Jan 1977.

[17] R. Sharma and S. Hutchinson. Optimizing hand/eye configuration for visual servo systems. In *IEEE Int. Conf. on Robotics and Automation, ICRA'95*, volume 1, pages 172–177, Nagoya, Japan, May 1995.

[18] A. Sluzek. Identification and inspection of 2-d objects using new moment-based shape descripters. *Pattern Recognition Letters*, 16:687–697, 1995.

[19] G. Wells, C. Venaille, and C. Torras. Vision-based robot positioning using neural networks. In *Image and Vision Computing*, volume 14, pages 715–732. World Scientific Publishing Co.Pte.Ltd, OPTnote = , OPTannote = , December 1996.

[20] R. Y. Wong and E. L. Hall. Scene matching with invariants moments. In *Computer Graphics Image Proceeding*, volume 8, pages 16–24, Sept 1978.

[21] W. H. Wong, W. C. Siu, and K. M. Lam. Generation of moment invariants and their uses for characters recognition. *Pattern Recognition*, 16:115–123, 1995.

Contents

Unité de recherche INRIA Rennes
IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)
Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)
Unité de recherche INRIA Rhône-Alpes : 655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)
Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)
Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
http://www.inria.fr
ISSN 0249-6399