Photometric moments: New promising candidates for visual servoing

Manikandan Bakthavatchalam, François Chaumette, Eric Marchand

Photometric moments: New promising candidates for visual servoing

Manikandan Bakthavatchalam, François Chaumette, Eric Marchand

***Abstract*— In this paper, we propose a new type of visual features for visual servoing : photometric moments. These global features do not require any segmentation, matching or tracking steps. The analytical form of the interaction matrix is developed in closed form for these features. Results from experiments carried out with photometric moments have been presented. The results validate our modelling and the control scheme. They perform well for large camera displacements and are endowed with a large convergence domain. From the properties exhibited, photometric moments hold promise as better candidates for IBVS over currently existing geometric and pure luminance features.**

I. INTRODUCTION

Visual servoing is the technique of controlling the degrees of freedom of a dynamic system by means of feedback information coming from visual sensors [1]. Visual features $\mathbf{s}(\mathbf{m}(t))$ are built from image measurements $\mathbf{m}(t)$. These measurements depend upon the pose of the robot at time instant t: $\mathbf{m}(t) = \mathbf{m}(\mathbf{r}(t))$. The goal of a visual servoing task consists in controlling the system in such a manner that the error in the visual features between their current and desired values $\mathbf{s}(t) - \mathbf{s}^*$ is regulated to 0. When geometrical features such as image points, straight lines and even 3D pose or homography are considered, a robust extraction, matching and spatio-temporal tracking of the visual measurements between $\mathbf{m}(\mathbf{r}(t_0))$ and $\mathbf{m}(\mathbf{r}(t))$ in the subsequent images is necessary [1].

In order to avoid this bottleneck, recent works [2]–[4] directly used the intensity as visual features. Servoing based on pure luminance feature (also known as *photometric visual servoing*) is an interesting step because image processing is almost completely avoided except for the image gradient calculations used in the interaction matrix [4]. However, one of the issues is that this approach suffers from a small convergence domain due to strong non-linearities in the system dynamics.

In this paper, we consider photometric moments as visual features. Moments are statistical scalar quantities that capture the essential characteristics of an unknown distribution. A set of invariants to translation, rotation and scale derived from moments were first introduced in pattern recognition in [5]. In computer vision, moments have been widely used as region-based shape descriptors for object recognition [6].

Manikandan Bakthavatchalam (Manikandan.Bakthavatchalam@inria.fr) and François Chaumette (François.Chaumette@inria.fr) are with Inria Rennes Bretagne Atlantique and IRISA, Campus de Beaulieu, 35042 Rennes cedex, France

Eric Marchand (Eric.Marchand@irisa.fr) is with Université de Rennes 1, IRISA, Inria Rennes Bretagne Atlantique, 35042 Rennes cedex, France

The use of image moments to serve as visual features in visual servoing has also been considered in [7]–[10]. However, all these works explicitly require a binary image or a spatial segmentation algorithm that produces a set of segmented homogeneous regions. In contrast, the method of photometric moments that we propose imposes no such restriction. In fact, using photometric moments, it is possible to avoid these steps thus liberating the visual servoing process from the crutches of image processing (image segmentation, etc.,) and feature tracking. In addition, we also obtain a large convergence domain. In [11], image moments computed over a set of SIFT keypoints are used for visual servoing. Naturally, this involves the robust extraction and matching of these keypoints at every frame. All the visual servoing approaches (but [2]–[4] and [12,13]) discard the pixel intensity information in the image. Homography-based visual servoing [14,15] is another existing visual servoing approach which uses textured objects but matching of the texture between the initial and desired images is required. Furthermore, a visual tracking method is necessary to estimate the homography parameters. The aim of our work is to use a different approach to capture the intensity information by means of photometric image moments and utilize them to perform servoing tasks with a robotic platform.

In the same spirit, Kernel-based visual servoing (KBVS) [12] is another recent approach where the authors proposed abstract visual features via spatial sampling functions called "kernels". While KBVS theory is conceptually elegant, how to design kernels and where to place them spatially remain open research issues. Further, our moment features remove the abstraction out of 'kernels' and instead give them an intuitive geometrical interpretation. In KBVS [12], Gaussian kernels and spatial Fourier transform (FT) have been used as visual features to control the four simplest degrees of freedom (dof) of the camera motion. Such kernels however might require a manual tuning of the kernel parameters. Photometric moments are free of such parameters and only moments upto second order are sufficient to control exactly the same subset of motions controlled via KBVS.

Our paper is organized as follows: Section II describes the formulation of photometric moments and mathematical developments for obtaining their interaction matrix. Section III explains the motivation behind the choice of our visual features and control aspects. Results of simulations and real experiments are presented and discussed in Section IV. Conclusions and future works are presented in Section V.

II. MODELLING OF PHOTOMETRIC MOMENTS

The general expression for the photometric moments in the image plane π can be written as

$$m_{pq} = \iint_{\pi} x^p y^q \;\; I((x,y),t) \;\; \mathrm{d}x\,\mathrm{d}y \tag{1}$$

where $p+q$ denotes the order of the moment and $I(x,y)$ is the image intensity function.
A kernel is as a piece-wise continuous function $\mathbf{K} : \pi \rightarrow \mathbb{R}$ that produces a measurement $\xi \in \mathbb{R}$ when the spatial coordinates are projected onto it [12].

$$\xi(t) = \int_{\pi} K(x,y) \;\; I((x,y),t)) \;\; \mathrm{d}x\,\mathrm{d}y \tag{2}$$

So, photometric moments neatly fits into the above definition of kernel. Also, it has to be noted that in the case of binary moments that have been studied previously [7], [8], the range of function $I(x,y)$ was restricted to take on only two values : 0 for the background region and 1 for the image region corresponding to the object projection. So, the segmentation and tracking of that specific region was necessary. As expressed in (1), photometric moments, on the other hand, take into account the intensity information in all the image plane.

A. Development of the interaction matrix

In order to use any feature s for visual servoing, its interaction matrix $\mathbf{L}_s$ has to be determined. This matrix links the variation of the visual features to the spatial velocity of the vision sensor $\mathbf{v}_c$ (expressed in the camera frame $\mathcal{F}_c$) [1]:

$$\dot{s} = \mathbf{L}_s \; \mathbf{v}_c \tag{3}$$

The derivative of the photometric moments can be written as:

$$\dot{m}_{pq} = \iint_{\pi} x^p y^q \, \dot{I}(x,y) \, \mathrm{d}x\,\mathrm{d}y \tag{4}$$

If the derivative of the photometric moments could be expressed in terms of the camera velocity, we could obtain the interaction matrix of the image moments:

$$\dot{m}_{pq} = \mathbf{L}_{m_{pq}} \; \mathbf{v}_c \tag{5}$$

As in [2], we make use of the classical brightness constancy assumption [16] which considers that the intensity of a moving point $\mathbf{x} = (x,y)$ remains the same between successive images.

$$I(\mathbf{x} + \delta\mathbf{x}, t + \delta t) = I(\mathbf{x}, t) \tag{6}$$

A first order Taylor expansion of (6) around $\mathbf{x}$ leads to

$$\nabla\mathbf{I}^\top \dot{\mathbf{x}} + \dot{I} = 0 \tag{7}$$

where $\nabla\mathbf{I}^\top = \begin{bmatrix} \frac{\partial I}{\partial x} & \frac{\partial I}{\partial y} \end{bmatrix} = \begin{bmatrix} I_x & I_y \end{bmatrix}$ is the spatial gradient at the image point $\mathbf{x}$. We thus obtain

$$\dot{I}(x,y) = -\nabla\mathbf{I}^\top \dot{\mathbf{x}} \tag{8}$$

which is the well-known optic flow constraint equation [16]. The relation which links the image point velocity to the camera velocity is well known [1] and given by :

$$\dot{\mathbf{x}} = \mathbf{L}_{\mathbf{x}} \; \mathbf{v}_c \tag{9}$$

where

$$\mathbf{L}_{\mathbf{x}} = \begin{bmatrix} \mathbf{L}_x \\ \mathbf{L}_y \end{bmatrix} = \begin{bmatrix} \frac{-1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & \frac{-1}{Z} & \frac{y}{Z} & 1+y^2 & -xy & -x \end{bmatrix} \tag{10}$$

Let us consider that the scene is planar. The depth of the scene points is then related to the image point coordinates by the relation:

$$\frac{1}{Z} = Ax + By + C \tag{11}$$

where A, B and C are scalar parameters that describe the configuration of a plane. Typically, when this plane is parallel to the image plane, $A = B = 0$.
By plugging (11) into (10) and (9) into (8), we obtain

$$\dot{I}(x,y) = -\nabla\mathbf{I}^\top \mathbf{L}_{\mathbf{x}} \; \mathbf{v}_c = \mathbf{L}_I \; \mathbf{v}_c \tag{12}$$

where $\mathbf{L}_I$ is given by [2] :

$$\mathbf{L}_I^\top = \begin{bmatrix} I_x(Ax+By+C) \\ I_y(Ax+By+C) \\ (-xI_x - yI_y)(Ax+By+C) \\ -xyI_x - (1+y^2)I_y \\ (1+x^2)I_x + xyI_y \\ -yI_x + xI_y \end{bmatrix} \tag{13}$$

Substituting (12) into the equation for the moment derivatives (4), we see that

$$\dot{m}_{pq} = \iint_{\pi} x^p y^q \, \mathbf{L}_I \, \mathrm{d}x\,\mathrm{d}y\,\mathbf{v}_c \tag{14}$$

We can then write down the interaction matrix of the moments as

$$\mathbf{L}_{m_{pq}} = \iint_{\pi} x^p y^q \, \mathbf{L}_I \, \mathrm{d}x\,\mathrm{d}y \tag{15}$$

Direct substitution of $\mathbf{L}_I$ defined in (13) into the above equation gives us

$$\mathbf{L}_{m_{pq}}^\top = \begin{bmatrix} \iint_{\pi} x^p y^q \, I_x(Ax+By+C) \, \mathrm{d}x\,\mathrm{d}y \\ \iint_{\pi} x^p y^q \, I_y(Ax+By+C) \, \mathrm{d}x\,\mathrm{d}y \\ \iint_{\pi} x^p y^q \, (-xI_x - yI_y)(Ax+By+C) \, \mathrm{d}x\,\mathrm{d}y \\ \iint_{\pi} x^p y^q \, (-xyI_x - (1+y^2)I_y) \, \mathrm{d}x\,\mathrm{d}y \\ \iint_{\pi} x^p y^q \, ((1+x^2)I_x + xyI_y) \, \mathrm{d}x\,\mathrm{d}y \\ \iint_{\pi} x^p y^q \, (xI_y - yI_x) \, \mathrm{d}x\,\mathrm{d}y \end{bmatrix} \tag{16}$$

Direct computation in this form would be time-consuming. In order to simplify the above entries, let us introduce a compact notation as follows:

$$m_{pq}^{\nabla x} = \iint_{\pi} x^p y^q \, I_x \, \mathrm{d}x\,\mathrm{d}y \tag{17a}$$

$$m_{pq}^{\nabla y} = \iint_{\pi} x^p y^q \, I_y \, \mathrm{d}x \, \mathrm{d}y \tag{17b}$$

A component-wise representation of $L_{m_{pq}}$ can be written as:

$$\mathbf{L}_{m_{pq}} = \begin{bmatrix} L_{\mathtt{m_{pq}}}^{v_x} & L_{\mathtt{m_{pq}}}^{v_y} & L_{\mathtt{m_{pq}}}^{v_z} & L_{\mathtt{m_{pq}}}^{\omega_x} & L_{\mathtt{m_{pq}}}^{\omega_y} & L_{\mathtt{m_{pq}}}^{\omega_z} \end{bmatrix} \tag{18}$$

As a representative example, let us consider a single component of the interaction matrix; the one corresponding to the translational velocity in x.

$$\begin{aligned} L_{\mathtt{m_{pq}}}^{v_x} &= \iint_{\pi} x^p y^q \, (Ax + By + C) \, I_x \, \mathrm{d}x \, \mathrm{d}y \\ &= A \iint_{\pi} x^{p+1} y^q \, I_x \, \mathrm{d}x \, \mathrm{d}y + B \iint_{\pi} x^p y^{q+1} \, I_x \, \mathrm{d}x \, \mathrm{d}y \\ &\quad + C \iint_{\pi} x^p y^q \, I_x \, \mathrm{d}x \, \mathrm{d}y) \\ &= A \, m_{p+1,q}^{\nabla x} + B \, m_{p,q+1}^{\nabla x} + C \, m_{p,q}^{\nabla x} \end{aligned} \tag{19}$$

Similar developments for each entry leads to the following set of expressions:

$$\begin{cases} L_{\mathtt{m_{pq}}}^{v_x} = A \, m_{p+1,q}^{\nabla x} + B \, m_{p,q+1}^{\nabla x} + C \, m_{p,q}^{\nabla x} \\ L_{\mathtt{m_{pq}}}^{v_y} = A \, m_{p+1,q}^{\nabla y} + B \, m_{p,q+1}^{\nabla y} + C \, m_{p,q}^{\nabla y} \\ L_{\mathtt{m_{pq}}}^{v_z} = -A \, m_{p+2,q}^{\nabla x} - B \, m_{p+1,q+1}^{\nabla x} - C \, m_{p+1,q}^{\nabla x} \\ \qquad\quad - A \, m_{p+1,q+1}^{\nabla y} - B \, m_{p,q+2}^{\nabla y} - C \, m_{p,q+1}^{\nabla y} \\ L_{\mathtt{m_{pq}}}^{\omega_x} = - m_{p+1,q+1}^{\nabla x} - m_{p,q}^{\nabla y} - m_{p,q+2}^{\nabla y} \\ L_{\mathtt{m_{pq}}}^{\omega_y} = m_{p,q}^{\nabla x} + m_{p+2,q}^{\nabla x} + m_{p+1,q+1}^{\nabla y} \\ L_{\mathtt{m_{pq}}}^{\omega_z} = - m_{p,q+1}^{\nabla x} + m_{p+1,q}^{\nabla y} \end{cases} \tag{20}$$

We can observe that the expressions exhibit a complex form and involves image gradients. Image gradients are normally computed using derivative filters, which might introduce imprecision in the computed values. So, a further simplification step using Green's theorem was devised, partly to avoid this imprecision. The simplification step also led to an interesting result, which is presented next.

B. Simplifications using Green's Theorem

We proceed with simplifying the terms with ∇ superscripts in the above expressions. We begin with

$$m_{pq}^{\nabla x} = \iint_{\pi} \frac{\partial I(x,y)}{\partial x} x^p y^q \, \mathrm{d}x \, \mathrm{d}y \tag{21}$$

Green's theorem is an elegant mathematical tool which lets us compute the integral of a function defined over a subdomain π of $\mathbb{R}^2$ by transforming it into a line (curve/contour) integral over the boundary of π, denoted here as $\partial\pi$:

$$\iint_{\pi} (\frac{\partial Q}{\partial x} - \frac{\partial P}{\partial y}) \mathrm{d}x \, \mathrm{d}y = \oint_{\partial\pi} P \mathrm{d}x + \oint_{\partial\pi} Q \mathrm{d}y \tag{22}$$

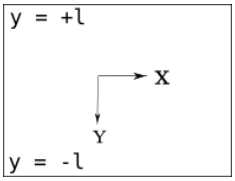

Fig. 1. Limits for evaluation represented over image

If we let $Q = I(x,y) \, x^p \, y^q$ and $P = 0$, then

$$\begin{cases} \dfrac{\partial Q}{\partial x} = \dfrac{\partial I}{\partial x} \, x^p \, y^q \; + \; p \, x^{p-1} \, y^q \, I(x,y) \\ \dfrac{\partial P}{\partial y} = 0 \end{cases} \tag{23}$$

Replacing these terms in Green's theorem, we obtain

$$\begin{aligned} \iint_{\pi} \Big[\frac{\partial I}{\partial x} \, x^p \, y^q \; + \; p \, x^{p-1} \, y^q \, I(x,y) \Big] \mathrm{d}x \, \mathrm{d}y \\ = \oint_{\partial\pi} I(x,y) \, x^p \, y^q \mathrm{d}y \end{aligned} \tag{24}$$

from which we deduce

$$\begin{aligned} \iint_{\pi} \frac{\partial I}{\partial x} \, x^p \, y^q \, \mathrm{d}x \, \mathrm{d}y = - \iint_{\pi} p \, x^{p-1} \, y^q \, I(x,y) \mathrm{d}x \, \mathrm{d}y \\ + \oint_{\partial\pi} x^p \, y^q \, I(x,y) \mathrm{d}y \end{aligned} \tag{25}$$

Therefore, we get

$$m_{pq}^{\nabla x} = -p \, m_{p-1,q} + \oint_{-l}^{+l} x^p \, y^q \, I(x,y) \mathrm{d}y \tag{26}$$

The integral term has to be evaluated for the limits $-l$ and l of the image plane (see Fig.1). Under the assumption that the border of the image is uniform, this term would evaluate to 0. This "constant border assumption" seems reasonable in practice (as evidenced by our results). In that case, we obtain:

$$m_{pq}^{\nabla x} = -p \, m_{p-1,q} \tag{27}$$

Similar developments in an identical manner yield

$$m_{pq}^{\nabla y} = -q \, m_{p,q-1} \tag{28}$$

Substituting Equations (27) and (28) into the set of equations given by (20), we get the final closed form expressions for the interaction matrix:

$$\begin{aligned} L_{\mathtt{m_{pq}}}^{v_x} &= -A(p+1)m_{pq} - Bpm_{p-1,q+1} - Cpm_{p-1,q} \\ L_{\mathtt{m_{pq}}}^{v_y} &= -Aqm_{p+1,q-1} - B(q+1)m_{p,q} - Cqm_{p,q-1} \\ L_{\mathtt{m_{pq}}}^{v_z} &= A \, (p+q+3) \, m_{p+1,q} + B(p+q+3) \, m_{p,q+1} \\ &\quad + C(p+q+2) \, m_{pq} \\ L_{\mathtt{m_{pq}}}^{\omega_x} &= q \, m_{p,q-1} + (p+q+3) \, m_{p,q+1} \\ L_{\mathtt{m_{pq}}}^{\omega_y} &= -p \, m_{p-1,q} + (p+q+3) \, m_{p+1,q} \\ L_{\mathtt{m_{pq}}}^{\omega_z} &= p \, m_{p-1,q+1} - q \, m_{p+1,q-1} \end{aligned} \tag{29}$$

From the terms above, we observe that in order to calculate $L_{m_{pq}}$, only moments of order upto $p+q+1$ are required. Also, we see that the interaction matrix components corresponding to the rotational degrees of freedom are free from 3D parameters A, B and C. Another important remark is that the image gradients do not appear anymore, which is beneficial in terms of computation time and robustness to noise. It is interesting to note that exactly the same analytical form as in [7] has been obtained, although in our case, a completely different method has been used for the derivations. So, all the useful results of [7] and [8] as regards to the visual feature selection are applicable as they are for photometric moments.

III. Control Scheme

Inspired by [7] and [8], the following set of visual features were chosen to be used in this paper:

$$\mathbf{s} = (x_n, y_n, a_n, s_x, s_y, \alpha) \tag{30}$$

where $x_n = x_g a_n$, $y_n = y_g a_n$, $a_n = Z^* \sqrt{a^*/a}$. Z^* is the distance between the object plane and the camera, $a = m_{00}$ is the photometric area. $x_g = m_{10}/m_{00}$ and $y_g = m_{01}/m_{00}$ are the centre of gravity coordinates along the x and y axes. The feature set (x_n, y_n, a_n) is responsible for controlling the three translation degrees of freedom. This feature set has been selected because it was already shown to produce straight line camera trajectories for pure 3D translation motions in the case of binary moments [8].

As in [7], the features s_x, s_y and α were chosen to control the rotational degrees of freedom. α represents the orientation of the image texture and is used to control the rotational motion around the optical axis. It is given by:

$$\alpha = \frac{1}{2} \arctan \left(\frac{2\mu_{11}}{\mu_{02} + \mu_{20}} \right) \tag{31}$$

where μ_{pq} are the centered moments. As for s_x and s_y, we have

$$s_x = (c_2 c_3 + s_2 s_3)/K \tag{32}$$

$$s_y = (s_2 c_3 + c_2 s_3)/K \tag{33}$$

where $c_3 = {c_1}^2 - {s_1}^2, s_3 = 2 s_1 c_1$ and $K = I_1 {I_3}^{\frac{3}{2}}/\sqrt{a}$ and

$$\begin{cases} I_1 = {c_1}^2 + {s_1}^2, I_2 = {c_2}^2 + {s_2}^2, I_3 = \mu_{02} + \mu_{20} \\ c_1 = \mu_{20} - \mu_{02}, s_1 = 2\mu_{11} \\ c_2 = \mu_{03} - 3\mu_{21}, s_2 = \mu_{30} - 3\mu_{12} \end{cases} \tag{34}$$

These features are built from Hu's set of invariants [5].

We recall that the distinction that has to be made clear is that our features and their interaction matrices depend on photometric moments and not the binary moments as was done in [7] and [8]. We used the classical control law

$$\mathbf{v}_c = -\lambda \, \mathbf{L}_{\mathbf{s}^*}^{||-1} (\mathbf{s} - \mathbf{s}^*) \tag{35}$$

without any modifications, where $\mathbf{L}_{\mathbf{s}^*}^{||}$ is the interaction matrix at the desired position. This choice allows avoiding the online estimation of any 3D parameter, while ensuring local asymptotic stability of the system in case the desired configuration is such that the object and image planes are parallel ($A^* = B^* = 0$) [1]. This matrix has the following decoupled form:

$$\hat{\mathbf{L}}_{\mathbf{s}}^{||-1} = \begin{bmatrix} 1 & 0 & 0 & L_{\mathtt{x_n}}^{\omega_x} & L_{\mathtt{x_n}}^{\omega_y} & y_n \\ 0 & 1 & 0 & L_{\mathtt{y_n}}^{\omega_x} & L_{\mathtt{y_n}}^{\omega_y} & -x_n \\ 0 & 0 & 1 & L_{\mathtt{a_n}}^{\omega_x} & L_{\mathtt{a_n}}^{\omega_y} & 0 \\ 0 & 0 & 0 & L_{\mathtt{s_x}}^{\omega_x} & L_{\mathtt{s_x}}^{\omega_y} & 0 \\ 0 & 0 & 0 & L_{\mathtt{s_y}}^{\omega_x} & L_{\mathtt{s_y}}^{\omega_y} & 0 \\ 0 & 0 & 0 & L_{\alpha}^{\omega_x} & L_{\alpha}^{\omega_y} & -1 \end{bmatrix} \tag{36}$$

This structure of the interaction matrix is advantageous since it decouples the rotational motions from the translation.

IV. Experimental Results

This section presents results from both simulation and real experiments that have been conducted on a 6dof Gantry robot. The code developed for this research was based on the ViSP software platform [17]. We conceived two sets of experiments to study photometric moments. In the first set, we used a reduced set of degrees of freedom of our experimental platform, specifically only the 3D translations and rotation around the optical axis. This set of experiments will allow us to see if there is any undesired behaviour arising due to i) assumptions made in the theoretical developments and ii) the inclusion of pixel intensities. Also, this is the same subset of dof utilised in KBVS [12]. In the second set of experiments, a full 6DOF visual servoing was performed with the full set of features presented in Section III.

A. Simulation results

For the simulation, only results obtained for 6DOF have been reproduced here due to space constraints. The original parameters of the calibrated camera and the same textured images used in the actual experiments were employed in the simulation. For this experiment, translation displacements of $\Delta T = [10cm,\ 10cm,\ 70cm]$ and rotational displacements of $\Delta R = [20deg,\ 20deg,\ 30deg]$ are required in order to attain the desired pose. The desired pose is such that the object and the image planes are parallel and the camera is oriented at 10deg around the optical axis (see Fig.2b).

We can observe from the results in Fig.2 that the errors decrease exponentially and the pose parameters converge to their desired values. A good behaviour has been observed in spite of the very large displacements to realize. Let us note here that for the same task, pure luminance features [2,4] does not allow the system to converge since the displacement to realize is very large.

B. Experimental Results

Experiment B1: This experiment is a representative case from a series of experiments conducted with various textures (differing in shape and image content). A non-rectangular, irregularly cut graffiti texture was used. The robot was configured to use only four of its available dof and only the first four of the proposed visual features from (30). The camera displacement to realize was

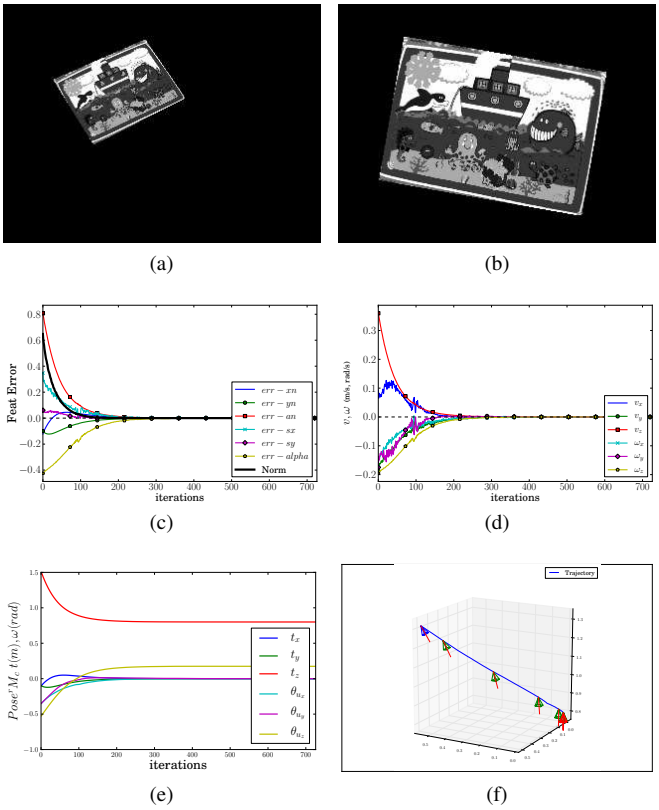

Fig. 2. Simulation results (a) Initial image, (b) Desired image, (c) Errors in visual features $\mathbf{s} - \mathbf{s}*$, (d) Robot velocities applied, (e) Robot pose, (f) Spatial camera trajectory

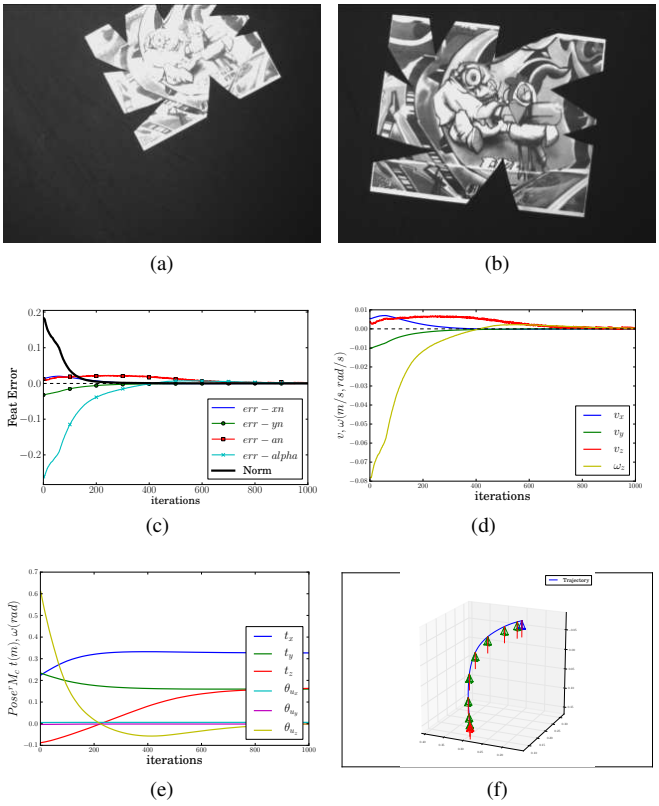

Fig. 3. Experimental results for Experiment B1 : Photometric moments VS with graffiti texture (a) Initial image, (b) Desired image, (c) Errors in visual features $\mathbf{s} - \mathbf{s}*$, (d) Robot velocities applied (e) Robot pose, (f) Spatial camera trajectory

$\Delta M = [-10cm,\ 8cm,\ -25cm,\ 35deg]$. A gain of $\lambda = 0.2$ and depth of $Z^* = 0.8$ was used in the control law (35). From Figs. 3a and 3b, we can clearly perceive a difference in lighting between the initial and final images. Then, a portion of the texture is occluded in the initial image. In spite of such disturbances, we can note from Fig. 3 that the visual features converged to the desired value, driving the robot to the desired pose. The final accuracy of the positioning from the robot odometry was $-1.17mm, -4.52mm, -1.31mm, 0.09deg]$. The errors in translations are of the order of few mm while the rotation error is less than 1 degree. We can also observe that the camera trajectory is not a straight-line at the beginning of the servo (see Fig. 3f). First, a portion of the texture is occluded from the initial camera view. There are a subset of intensities present in the desired image which are missing from the initial image and during the initial stages of the servo. Such phenomena are not accounted for in our model and this explains the non-optimal camera spatial trajectory during the initial iterations. However, such phenomena of missing intensities occur especially for large displacements (such as the one we have chosen) and reduce as the system is driven to the desired configuration.

Experiment B2: In this experiment, we tested the photometric moments using all the degrees of freedom of our 6DOF robot. All the six visual features proposed in Section III were used in this experiment. A gain of $\lambda = 0.7$ and desired depth of $Z^* = 0.8$ were considered. The camera displacement realized in this experiment was very large as well. $\Delta T = [10.23cm,\ 42.76cm,\ 10cm]$ and rotational displacements of $\Delta R = [25deg,\ 15deg,\ 12deg]$ around the coordinate axes.

The initial pose is such that the image plane is not parallel to the object plane as shown in Fig.4a. This requires a control law with a full rank interaction matrix to drive the task error to zero. In this case involving all the six degrees of freedom, photometric moment features converged to the desired values (see Fig. 4e). The positioning accuracy for this experiment is $[0.66mm, -1.42mm, 0.16mm]$ for the translation and $[-0.13deg, -0.06deg, 0.002deg]$ for the rotation axes. However, the camera trajectory is not as direct like in the previous cases. This is due to the use of the approximated interaction matrix at the parallel position $\mathbf{L}_{\mathbf{s}}^{||}$ and the non-optimal choice of visual features s_x and s_y. This was not too surprising, since earlier research has demonstrated that the choice of the last two features is difficult [8].

The same experiments as described above were carried out with pure luminance features [2] and we observed that the system did not converge to the desired pose. As said earlier, this is due to the small convergence domain of that

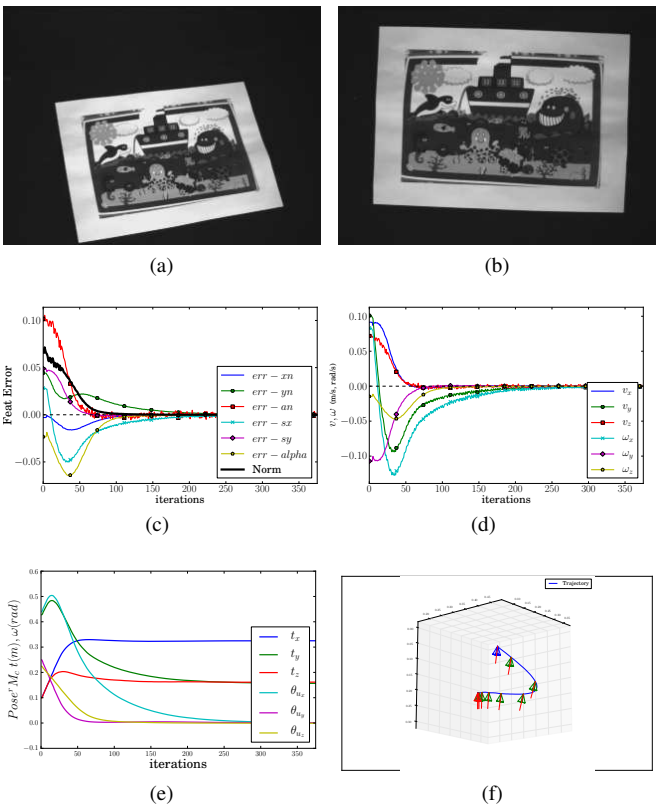

Fig. 4. Experimental results for Experiment B2 : Photometric moments VS in 6DOF (a) Initial image, (b) Desired image, (c) Errors in visual features $\mathbf{s} - \mathbf{s}*$, (d) Robot velocities applied, (e) Robot pose, (f) Spatial camera trajectory

approach and presence of strong non-linearities in the system. Photometric moments do not suffer from this restriction, thanks to the almost linear form of the interaction matrix for the chosen features. On the other hand, when starting from near the desired pose, pure luminance features converged with an excellent accuracy with positioning errors less than 1mm in translation and less than 1 degree for each of the rotations. This is because of the high redundancy in that approach which makes it extremely sensitive to even small errors. Therefore, in applications where an excellent final accuracy is needed, the recommended approach would be to start with the photometric moments and at near-convergence, switch to the pure luminance features.

V. Conclusions & Future Work

In this paper, photometric image moments have been introduced as new visual features for image-based visual servoing. The interaction matrix has been developed in closed-form for the proposed photometric moments. Photometric moments allow us to avoid any spatial segmentation steps and reduce the image processing to a simple and systematic moments computation on all the image plane, while simultaneously leveraging the excellent decoupling properties of binary image moments. Photometric moments, although based on intensity, can be used without any modification of the classical control laws. Experimental results have been presented that confirm the validity of our approach. In comparison with existing methods (based on geometric features or binary moments), photometric moments can be used to servo planar complex textured objects without any image processing. They perform well even when large displacements are involved since they possess a larger convergence domain than the pure luminance features. However, the method is not free of problems like local minima and susceptible to failure when the target object has a degenerate photometric profile. In future works, an analysis of conditions under which the method is likely to fail has to be made. More work is required to remove the restrictive constant border hypothesis made in the interaction matrix developments. Our immediate future work will be oriented toward finding a good method to derive robust visual features to control the two out-of-plane rotations. The goal is to be able to devise visual servoing schemes for full 6DOF with optimal characteristics (exponential decrease of the feature errors, decoupled control law and an optimal camera spatial trajectory).

References

[1] F. Chaumette and S. Hutchinson, "Visual servo control, part i: Basic approaches," *IEEE Robot. Autom. Mag.*, vol. 13, no. 4, pp. 82–90, Dec. 2006.

[2] C. Collewet, E. Marchand, and F. Chaumette, "Visual servoing set free from image processing," in *IEEE Int. Conf. on Robotics and Automation, ICRA'08*, Pasadena, California, May 2008, pp. 81–86.

[3] S. Han, A. Censi, A. Straw, and R. Murray, "A bio-plausible design for visual pose stabilization," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'10*, Taipei, Taiwan, Oct. 2010.

[4] C. Collewet and E. Marchand, "Photometric visual servoing," *IEEE Trans. on Robotics*, vol. 27, no. 4, pp. 828–834, Aug. 2011.

[5] M. Hu, "Visual pattern recognition by moment invariants," *IEEE Trans. on Information Theory*, vol. 8, no. 2, pp. 179–187, 1962.

[6] M. Sonka, V. Hlavac, R. Boyle *et al.*, *Image processing, analysis, and machine vision*. PWS publishing Pacific Grove, CA, 1999, vol. 1.

[7] F. Chaumette, "Image moments: a general and useful set of features for visual servoing," *IEEE Trans. on Robotics*, vol. 20, no. 4, pp. 713–723, Aug. 2004.

[8] O. Tahri and F. Chaumette, "Point-based and region-based image moments for visual servoing of planar objects," *IEEE Trans. on Robotics*, vol. 21, no. 6, pp. 1116–1127, Dec. 2005.

[9] G. Wells, C. Venaille, and C. Torras, "Vision-based robot positioning using neural networks," *Image and Vision Computing*, vol. 14(10), pp. 715–732, 1996.

[10] J. Wang and H. Cho, "Micropeg and hole alignment using image moments based visual servoing method," *IEEE Trans. on Industrial Electronics*, vol. 55(3), pp. 1286–1294, 2008.

[11] F. Hoffmann, T. Nierobisch, T. Seyffarth, and G. Rudolph, "Visual servoing with moments of SIFT features," in *IEEE Int. Conf. on Systems, Man and Cybernetics, SMC'06*, vol. 5, Oct., pp. 4262–4267.

[12] V. Kallem, M. Dewan, J. Swensen, G. Hager, and N. Cowan, "Kernel-based visual servoing," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'07*, San Diego, California, pp. 1975–1980.

[13] A. Dame and E. Marchand, "Mutual information based visual servoing," *IEEE Trans. on Robotics*, vol. 27, no. 5, pp. 958–969, Oct. 2011.

[14] S. Benhimane and E. Malis, "Homography-based 2d visual tracking and servoing," *The International Journal of Robotics Research, IJRR'07*, vol. 26, no. 7, pp. 661–676, 2007.

[15] G. Silveira and E. Malis, "Direct visual servoing: Vision-based estimation and control using only nonmetric information," *IEEE Transactions on Robotics*, vol. 28, no. 4, pp. 974–980, 2012.

[16] B. K. P. Horn and B. G. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 17, pp. 185–203, 1981.

[17] E. Marchand, F. Spindler, and F. Chaumette, "ViSP for visual servoing: a generic software platform with a wide class of robot control skills," *IEEE Robot. Autom. Mag.*, vol. 12, no. 4, pp. 40–52, Dec. 2005.