



HAL
open science

Visual servoing on non-planar objects from active vision

Christophe Collewet, François Chaumette

► **To cite this version:**

Christophe Collewet, François Chaumette. Visual servoing on non-planar objects from active vision. IEEE Int. Conf. on Robotics and Automation, ICRA'07, 2007, Roma, Italy. pp.2446-2451. inria-00350641

HAL Id: inria-00350641

<https://inria.hal.science/inria-00350641>

Submitted on 7 Jan 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Visual Servoing on Non-Planar Objects From Active Vision

Christophe Collewet and François Chaumette

Abstract—This paper presents a method to achieve visual servoing tasks when the shape of the object being observed as well as the final image are unknown. Therefore, a reconstruction phase has to be performed during the camera motion. More precisely, the reconstruction phase is based on the measurement of the 2D motion in a region of interest and on the measurement of the camera velocity. Since the 2D motion depends on the shape of objects being observed, we introduce an unified motion model to cope as well with planar as with non-planar objects. This is an improvement of our previous works described in [1], [2] where only planar objects had been considered. However, since this model is only an approximation, we propose to use active vision to enlarge its domain of validity. Experimental results validate the proposed approach.

I. INTRODUCTION

This paper addresses the problem of achieving robotic tasks by visual servoing when the observed object and the desired image are unknown. Such cases can appear for example in surgical domain, agriculture, agrifood industry or in unknown environments like underwater or space (see for example [3]–[5]). They can also appear when considering specific tasks, like perception tasks, where the camera has, for example, to move with respect to the object of interest to perform automatically an optical character recognition task. In that case, the final image is also unknown since the goal of the task is precisely to move the camera to see clearly the characters to decode [6].

Classically, visual servoing approaches [7] cannot cope with this problem. Indeed, concerning image-based approach [8] the desired features are needed. In addition, to cope with non-planar objects, the knowledge of the object is required to compute the so-called interaction matrix. Desired features are also needed for the extended 2D visual servoing [9] as well as for the model-free visual servoing [10]. The position-based approach cannot also be used since the pose between the camera and the object is needed [11] which cannot be obtained without a precise model of the object. On the contrary, visual servoing based on dynamic visual features does not need the knowledge of the desired image [12], [13]. Indeed, the control law is based on the use of affine parameters of the 2D motion. This allows to achieve a positioning task consisting in moving the camera to a position parallel to an object of unknown shape. However, this approach is currently restricted to planar objects.

The aim of this article is to control the orientation of the tangent plane at a certain point on an unknown object corresponding to the center of a region of interest (ROI) and

to move this point to the principal point to fulfill a fixation task. Note that we use in this paper the same control law and the same image processing as in [1]. Emphasize is made here on an unified approach to cope with planar or non-planar objects, contrary to what has been proposed in our previous works [1], [2] where only planar objects could be considered. Since the shape of the object as well as the desired image are unknown, as in [1] or [2], a 3D reconstruction phase by *dynamic vision* is needed. However, we will see in this paper the benefit of using *active vision* instead of dynamic vision.

The paper is organized as follows: first, we present in Section II a brief review on previous works relevant to 3D reconstruction by dynamic vision. We show how to recover the structure of the object in Section III. Section IV shows how active vision can be used to help the reconstruction process while the control law is recalled in Section V. Finally, experimental results concerning planar and non-planar objects are presented in Section VI.

II. PREVIOUS WORKS

Let us consider a point M of the object described by $\mathbf{X} = (X, Y, Z)$ in the camera frame, with the Z axis the camera optical axis. Assuming without loss of generality a unit focal length, this point projects in m , described by $\mathbf{x} = (x, y, 1)$, according to

$$\mathbf{x} = \frac{\mathbf{X}}{Z} \quad (1)$$

which yields to the well-known relation [14]

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = \begin{bmatrix} -1/Z & 0 & x/Z & xy & -1-x^2 & y \\ 0 & 1/Z & y/Z & 1+y^2 & -xy & -x \end{bmatrix} \mathbf{v} \quad (2)$$

where $\mathbf{v} = (v, \omega)$ is the camera velocity and $\mathbf{v} = (v_x, v_y, v_z)$ and $\omega = (\omega_x, \omega_y, \omega_z)$ its translational and rotational components respectively. In (2), only the depth Z is unknown if \mathbf{x} , $\dot{\mathbf{x}}$ and \mathbf{v} can be measured.

Various approaches to estimate Z exist. They are based on different approaches to cope with $\dot{\mathbf{x}}$. Some works are based on the assumption that the brightness of m remains constant during the motion yielding the well-known additional constraint [15]

$$\dot{x}I_x + \dot{y}I_y + I_t = 0 \quad (3)$$

where I_x , I_y and I_t represent the spatio-temporal derivatives of the intensity in m . By substituting \dot{x} and \dot{y} given by (2) in (3), an expression of Z can be obtained [16] (note that this work treats the more general case where \mathbf{v} is also supposed to be unknown). Such approaches, known as *direct approaches*, require accurate estimations of I_x , I_y and I_t and therefore, are not very accurate in practice. Another

approach is to locally model the surface of the object in the neighborhood of M . That provides an expression of $1/Z$ in function of the chosen parameterization, which can be used in (2) to exhibit a parametric model of the 2D motion. These parameters can be then computed by a 2D motion estimation method. Finally, an expression of the structure of the object can be extracted [17]. Such an approach has also been used in our previous work [1] to deal with planar objects. These approaches are known as *indirect approaches* since they require an intermediate computation of the 2D motion. More precisely, they are relevant to *continuous approaches* since they use the 2D velocity. Such works implicitly assume that the acquisition rate is high (or the camera velocity low) enough so that the parameters of the motion model can be considered as constant between two frames. We will see that this assumption is valid here.

III. STRUCTURE OF THE OBJECT

Let us consider a point P described by $\mathbf{X}_P = (X_P, Y_P, Z_P)$ in the camera frame. This point is chosen so that its projection p described by $\mathbf{x}_p = (x_p, y_p, 1)$ lies in the center of the ROI. The position of this point being used in a control scheme, we will not assume as usually that it lies near the principal point as it is in lots of papers (see [1], [13], [17] for example), since the goal of the task is precisely to move it to this particular point.

Let us also consider a point M in a neighborhood of P . The tangent plane in P expresses as follows

$$Z = Z_P + A_{10}(X - X_P) + A_{01}(Y - Y_P) \quad (4)$$

where $A_{10} = \left. \frac{\partial Z}{\partial X} \right|_P$ and $A_{01} = \left. \frac{\partial Z}{\partial Y} \right|_P$ leading to the normal \mathbf{n} in P

$$\mathbf{n} = (A_{10}, A_{01}, -1) \quad (5)$$

which is required to compute the control law (see Section V).

On the other hand, we can rewrite (4) in a more compact form

$$Z = A_{00} + A_{10}X + A_{01}Y \quad (6)$$

where $A_{00} = -\mathbf{n}^\top \mathbf{X}_P$ such that (6) can be rewritten with respect to the normalized coordinates \mathbf{x} as follows

$$\frac{1}{Z} = \boldsymbol{\alpha}^\top \mathbf{x} \quad (7)$$

with $\boldsymbol{\alpha} = (\alpha_{10}, \alpha_{01}, \alpha_{00})$ where $\alpha_{10} = -A_{10}/A_{00}$, $\alpha_{01} = -A_{01}/A_{00}$ and $\alpha_{00} = 1/A_{00}$ or as follows by introducing $u = x - x_p$ and $v = y - y_p$

$$\frac{1}{Z} = \boldsymbol{\beta}^\top \mathbf{u} \quad (8)$$

with $\mathbf{u} = (u, v, 1)$ and $\boldsymbol{\beta} = (\beta_{10}, \beta_{01}, \beta_{00})$ where $\beta_{10} = \alpha_{10}$, $\beta_{01} = \alpha_{01}$ and $\beta_{00} = \alpha_{00} + \alpha_{10}x_p + \alpha_{01}y_p$.

Therefore, if we can measure $\boldsymbol{\alpha}$ or $\boldsymbol{\beta}$, the unit normal $\tilde{\mathbf{n}}$ in P can be deduced

$$\tilde{\mathbf{n}} = -\frac{\boldsymbol{\alpha}}{\|\boldsymbol{\alpha}\|}. \quad (9)$$

To compute $\boldsymbol{\alpha}$ we assume that the surface in the neighborhood of P is continuous so that we can write an expression of object depths in function of image coordinates

$$\frac{1}{Z} = \sum_{p \geq 0, q \geq 0, p+q \leq n} \alpha_{pq} x^p y^q \quad (10)$$

where n is the degree of this polynomial.

Thereafter, if we substitute this expression in (2) we obtain a model of the 2D motion. Unfortunately, this model depends on the surface being observed. For example, for a planar object ($n = 1$), a quadratic model with 8 parameters is obtained; for a surface described by a quadratic form ($n = 2$), the model contains 15 different parameters. Therefore, to not depend on the object shape, a way to proceed is to consider only a small neighborhood in p so that the depths given by (10) coincide with the ones given by the tangent plane (8). Thereafter, since (8) is only true locally, after having substituted it in (2) we have to perform a Taylor series expansion in p leading to the following unified affine motion model

$$\dot{\mathbf{x}} = \mathbf{M} \mathbf{u} \quad (11)$$

where

$$\begin{cases} M_{11} = y_p \omega_x - 2x_p \omega_y + \beta_{00} v_z + \beta_{10} a \\ M_{12} = x_p \omega_x + \omega_z + \beta_{01} a \\ M_{13} = x_p y_p \omega_x - (1 + x_p^2) \omega_y + y_p \omega_z + \beta_{00} a \\ M_{21} = \beta_{10} b - y_p \omega_y - \omega_z \\ M_{22} = 2y_p \omega_x - x_p \omega_y + \beta_{00} v_z + \beta_{01} b \\ M_{23} = (1 + y_p^2) \omega_x - x_p y_p \omega_y - x_p \omega_z + \beta_{00} b \end{cases} \quad (12)$$

with $a = x_p v_z - v_x$ and $b = y_p v_z - v_y$.

Concretely, we obtain here an *unified motion model* since it can cope as well as with planar as with non-planar objects.

Thereafter, if we estimate the parameters of this motion model (see [1]) and if the 3D velocity is supposed to be known, an estimation $\hat{\boldsymbol{\beta}}$ can be obtained by solving a linear system. Indeed, (12) can be rewritten after simple manipulations as follows

$$\mathbf{C} \boldsymbol{\beta} = \boldsymbol{\Gamma} \quad (13)$$

with

$$\mathbf{C} = \begin{bmatrix} a & 0 & v_z \\ 0 & a & 0 \\ 0 & 0 & a \\ b & 0 & 0 \\ 0 & b & v_z \\ 0 & 0 & b \end{bmatrix} \quad (14)$$

and

$$\boldsymbol{\Gamma} = \begin{bmatrix} M_{11} - y_p \omega_x + 2x_p \omega_y \\ M_{12} - x_p \omega_x - \omega_z \\ M_{13} - x_p y_p \omega_x + (1 + x_p^2) \omega_y - y_p \omega_z \\ M_{21} + y_p \omega_y + \omega_z \\ M_{22} - 2y_p \omega_x + x_p \omega_y \\ M_{23} - (1 + y_p^2) \omega_x + x_p y_p \omega_y + x_p \omega_z \end{bmatrix} \quad (15)$$

leading to the following least-squares solution

$$\hat{\boldsymbol{\beta}} = (\mathbf{C}^\top \mathbf{C})^{-1} \mathbf{C}^\top \boldsymbol{\Gamma}. \quad (16)$$

However, this solution is only correct if the matrix $\mathbf{C}^\top \mathbf{C}$ is well conditioned, that is if the condition number ν of $\mathbf{C}^\top \mathbf{C}$ is low enough. Since $\mathbf{C}^\top \mathbf{C}$ is very simple, the analytical form of ν can be determined

$$\nu = \frac{a^2 + b^2 + v_z^2 + |v_z| \sqrt{a^2 + b^2 + v_z^2}}{a^2 + b^2 + v_z^2 - |v_z| \sqrt{a^2 + b^2 + v_z^2}}. \quad (17)$$

We will use this result in Section IV to ensure that ν does not become too high.

Remarks:

- Let us recall the well-known result that a 3D reconstruction is not possible if no translation occurs. Indeed, in that case $\mathbf{C}^\top \mathbf{C}$ becomes singular. Note that it becomes also singular when $a = b = 0$. We will return to this problem in Section IV.
- If we set p as the principal point in (12) we recover the classic affine motion model of a planar object when the terms of order 2 in x and y are neglected. That is what we used in [1].

IV. USE OF ACTIVE VISION

Since $\hat{\beta}$ depends on a measure of \mathbf{M} , the approximated motion model given in (11) has to fit as best as possible the true motion to provide an accurate value for $\hat{\beta}$. We focus in this section on the way to improve its domain of validity. On the other hand, note that (i) since a motion is needed to compute $\hat{\beta}$ and (ii) since the control law is based on the knowledge of $\hat{\beta}$ (see Section V), a preliminary step is required before the servoing step, i.e. when $\hat{\beta}$ is not known. Both these steps are studied in the next section.

We consider that $\hat{\beta}$ involved in (12) is only known through the approximation $\hat{\beta}$ given by (16). Moreover, we assume that a function $\varphi(u, v)$ (with $\varphi(0, 0) = 0$) exists so that the true depths can be expressed by

$$\frac{1}{Z} = \beta^\top \mathbf{u} + \varphi(u, v) \quad (18)$$

for any point belonging to the object.

This modeling of Z makes us possible to express the true motion model by substituting (18) in (2). We denote $\mathbf{E} = (E_u, E_v)$ the difference between the true 2D motion model and the approximated one given by (11) when using $\hat{\beta}$. After simple manipulations we obtain

$$\begin{cases} E_u &= a\epsilon_{00} + (a\epsilon_{10} + v_z\epsilon_{00})u + a\epsilon_{01}v + cu^2 + duv + \\ & \quad (a + v_zu)\varphi(u, v) \\ E_v &= b\epsilon_{00} + b\epsilon_{10}u + (b\epsilon_{01} + v_z\epsilon_{00})v + cvw + dv^2 + \\ & \quad (b + v_zv)\varphi(u, v) \end{cases} \quad (19)$$

with $c = \beta_{10}v_z - \omega_y$, $d = \beta_{01}v_z + \omega_x$ and $\epsilon = \beta - \hat{\beta}$. Note that ϵ is a function of v_z since $\hat{\beta}$ given by (16) depends on v_z .

Since the 2D motion is computed on a window W around p , we are interested in the minimization of

$$E_W^2 = \sum_i \sum_j (E_u^2 + E_v^2) \quad (20)$$

where i and j are counted on pixels around p .

Remark that v_z appears in each term of coefficients involved in (19), thus it is possible to minimize (20) with respect to v_z . Let us consider both steps mentioned at the top this section.

Recall that the goal of the first step is to provide an initial value for $\hat{\beta}$ which will be used in the second step to perform the positioning task while improving this value. Since $\hat{\beta}$ is initially not known, only constant translations are considered (as seen in Section V, a rotation motion needs an estimation of β). They are chosen so that p will move towards the principal point.

Whatever the shape of the object is, planar or not, one can show that E_W^2 takes the following form

$$E_W^2(v_z) = v_z f(v_z) + r \quad (21)$$

where f is a polynomial of degree 9 in v_z , which is not useful to detail, and r a positive constant such that $r = r_1$ with

$$r_1 = (v_x^2 + v_y^2) (\epsilon_{00}^2 K_1 + (\epsilon_{10}^2 + \epsilon_{01}^2) K_2) \quad (22)$$

if the object is planar, i.e. when $\varphi(u, v) = 0$. K_1 and K_2 are positive constants depending on the size of W .

To cope with non-planar objects, we consider for simplicity the class of objects where $\varphi(u, v)$ expresses as follows

$$\varphi(u, v) = \beta_{20}u^2 + \beta_{11}uv + \beta_{02}v^2 \quad (23)$$

where the parameters β_{ij} depend on the pose of the camera w.r.t. the object and on the object curvature. In that case, $r = r_1 + r_2$ with

$$r_2 = (v_x^2 + v_y^2) (\epsilon_{00}\bar{\beta}_2 K_2 + \tilde{\beta}_2 K_3 + (\beta_{20}^2 + \beta_{02}^2) K_4), \quad (24)$$

K_3 and K_4 being positive constants depending on W , $\bar{\beta}_2 = (\beta_{20} + \beta_{02})/2$ and $\tilde{\beta}_2 = \beta_{11}^2 + 2\beta_{20}\beta_{02}$.

Let us focus now on the second step. During this step both the reconstruction and the servoing are performed in order to achieve the positioning task and to improve the value of $\hat{\beta}$ provided by the first step of the algorithm.

We also obtain a similar form for E_W^2 than in the first step (when assuming that $\varphi(u, v)$ is given by (23) if the object is not planar). The function f is also a polynomial of degree 9 in v_z .

For a planar object we have $r = r_1 + r_3$ with

$$r_3 = 2\epsilon_{00}K_2 (v_x\omega_y - v_y\omega_x) + (K_3 + K_4) (\omega_x^2 + \omega_y^2) \quad (25)$$

while in the case of a non-planar object $r = \sum_{i=1}^4 r_i$ with

$$\begin{aligned} r_4 &= 2K_3 ((v_x\beta_{02} + v_y\beta_{11})\omega_y - (v_x\beta_{11} + v_y\beta_{20})\omega_x) + \\ & \quad 2K_4 (v_x\omega_y\beta_{20} - v_y\omega_x\beta_{02}). \end{aligned} \quad (26)$$

Whatever the step we consider, remark that r does not depend on v_z since $\hat{\beta}$ involved in ϵ is expressed here at $v_z = 0$. Therefore, minimizing $E_W^2(v_z)$ is equivalent to the minimization of $v_z f(v_z)$. We have thus to solve

$$f(v_z) + v_z \frac{\partial f(v_z)}{\partial v_z} = 0. \quad (27)$$

However, this equation is very difficult to solve since it is a polynomial of degree 9 in v_z . Moreover, β is involved in the

coefficients of this polynomial which is unknown. Rather to solve (27), we assume that r is small. It occurs when ϵ and the parameters β_{ij} are small, if so $v_z f(v_z) = E_W^2(v_z) - r \geq 0$. Consequently, we can state that $v^* = 0$ minimizes $v_z f(v_z)$ and thus (20).

In that case, the residual $E_W^2(v^*)$ is simply the constant r . Note that it is possible to minimize this residual. To do that, we have to choose a low value for $\|v\|$ during the first step (see r_1 and r_2) while in the second step we have to minimize all the components of v since v is involved in r_k ($k = 1, \dots, 4$) while ω is involved in r_3 and r_4 . Nevertheless, a compromise between low values for the 3D velocities and a sufficient 2D motion (to perform an accurate computation of \mathbf{M}) has to be taken into account.

Note that $v^* = 0$ is also the best choice for the condition number since it leads to an optimal value $\nu = 1$ (see (17)). In addition, since the other translations are not null, we are sure that the matrix $\mathbf{C}^\top \mathbf{C}$ will be never singular (see the remark at the end of Section III).

On the other hand, recall that the optimal choice $v^* = 0$ was already proposed in [18] but the problem to solve was very different than ours.

V. CONTROL LAW

First, let us remember the task to achieve. The goal is to ensure a given final orientation of the camera with respect to tangent plane π described by (4) and, also to ensure that P will still remain in the camera field of view. Once β is estimated and therefore α , the unit normal $\tilde{\mathbf{n}}$ of plane π in P in the camera frame can be derived from (9). To cope with a desired orientation of the camera w.r.t. the plane (π) different than parallel, we introduce \mathbf{n}^* such that $\mathbf{n}^* = \mathbf{R}^* \tilde{\mathbf{n}}$ where the matrix \mathbf{R}^* described the rotation between $\tilde{\mathbf{n}}$ and \mathbf{n}^* (see Fig. 1). Therefore, we have to move the camera so that $\mathbf{Z} = \mathbf{n}_c$ with $\mathbf{n}_c = -\mathbf{n}^*$ and \mathbf{Z} the unit vector carried by the optical axis. This rotation to perform can be expressed under the form $\mathbf{k}\theta$ where \mathbf{k} represents the unit rotation axis vector and θ the rotation angle around this axis

$$\mathbf{k} = \frac{\mathbf{n}_c \wedge \mathbf{Z}}{\|\mathbf{n}_c \wedge \mathbf{Z}\|} \quad (28)$$

and

$$\theta = \arccos(\mathbf{n}_c^\top \mathbf{Z}). \quad (29)$$

The camera orientation being known, it is possible to compute the control law. We used the one described in [10]. Indeed, it ensures that P remains in the camera field of view since the trajectory of p is a straight line between the current position p and the desired position p^* (which has been chosen as the principal point of the image). We describe here briefly this approach known as *hybrid visual servoing*.

First, \mathbf{x}_r is defined as follows

$$\mathbf{x}_r = \frac{1}{Z^*} \mathbf{X}_P = \frac{Z_P}{Z^*} \mathbf{x}_P \quad (30)$$

with Z^* the desired depth for P in final position or the depth in initial position if active vision is used (thus $Z_P/Z^* = 1$).

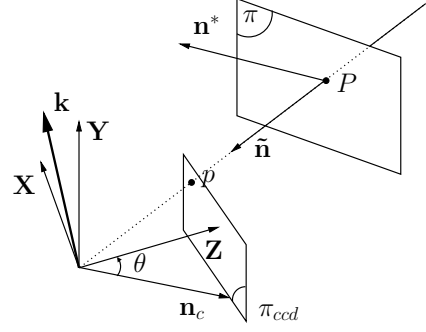


Fig. 1. Rotation to perform by the camera.

In few words, this approach is based on the regulation to zero of the following task function

$$\mathbf{e} = (\mathbf{x}_r - \mathbf{x}_r^*, \mathbf{k}\theta) \quad (31)$$

yielding to the camera velocity

$$\mathbf{v} = -\lambda \hat{\mathbf{L}}^{-1} \mathbf{e} \quad (32)$$

λ being a positive gain and $\hat{\mathbf{L}}^{-1}$ the inverse of an approximation of the interaction matrix given by [10]

$$\hat{\mathbf{L}}^{-1} = \begin{bmatrix} -Z^* \mathbf{I}_3 & Z^* [\mathbf{x}_r]_\times \\ \mathbf{0}_{3 \times 3} & \mathbf{I}_3 \end{bmatrix} \quad (33)$$

with $[\mathbf{x}_r]_\times$ the antisymmetric matrix associated to \mathbf{x}_r .

If active vision is used, we do not control v_z , we simply set $v_z = 0$ and since $Z_P/Z^* = 1$ we remove the 3rd component of \mathbf{e} leading finally to the following approximation of the inverse of the interaction matrix

$$\hat{\mathbf{L}}^{-1} = \begin{bmatrix} -Z^* \mathbf{I}_2 & Z^* [\mathbf{x}_P]_\times \\ \mathbf{0}_{3 \times 2} & \mathbf{I}_3 \end{bmatrix}. \quad (34)$$

Let us note that the value of Z_P required for the computation of \mathbf{x}_r is obtained by using (7) while \mathbf{x}_P is obtained by the tracking algorithm described in [19].

VI. EXPERIMENTAL RESULTS

Before presenting experimental results, we detail the way we measure the orientation error with respect to the tangent plane in P . We assume that a model of the object being observed by the camera exists in a certain frame \mathcal{F}_f . Knowing this model and the pose between \mathcal{F}_f and the camera frame \mathcal{F}_c (using dots, as can be seen on the next figures), it is possible to obtain from p the coordinates of P by intersecting the line of view with the object. Consequently, it becomes easy to compute the orientation of the tangent plane in P with respect to \mathcal{F}_c .

The experimental system consists of a 6 dof robot with an eye-in-hand CCD camera. The transformation matrix between the end-effector and the camera has been calibrated. In contrast, the intrinsic parameters of the camera are roughly known. The point p has been chosen from the initial image by the Harris detector. The orientation of the tangent plane in P w.r.t. the camera frame is described by the Cardan's angles Φ (respectively pitch, yaw, roll). We introduce the following notations concerning the superscript of Φ : i for the initial rotation, d for the desired one and m for the measured one.

Remark that the accuracy on α , required for the control law, can be improved in practice. Indeed, since the object is motionless, one can express a value α^f in a fixed frame that can be filtered, since a fixed value has to be obtained. Thereafter, this value is expressed in the camera frame to be used in the control law. Once α^f is sufficiently stable (typically after 7 iterations), the servoing step can begin. This step ends when the mean of the 2D motion is lower than 1/4 pixel and a last step begins. It consists in a servoing step where only the servoing operates without any 3D reconstruction. This step uses α^f , considered as constant, which is expressed in the camera frame.

The following constants have been used during all the experiments: $W = 111$ (that means 55 pixels each side of p), $\|v\| = 3$ cm/s during the first step, $Z^* = 65$ cm when active vision is not used, $\lambda = 0.3$ during the second step while $\lambda = 1$ during the last one.

Three objects have been used for these experiments, a planar object, a cylinder and a sphere of radius 7 cm. Of course, our algorithm does not know which object is being observed by the camera.

The first experiment concerns the planar object and consists in positioning the camera parallel to the object when using active vision. Fig. 2a depicts the components of the camera velocity; Fig. 2b the norm of the task function e ; Fig. 2c the magnitude of the rotation θ to reach the desired orientation; Fig. 2d the behavior of α (filtered and non-filtered) expressed in a fixed frame. Finally, the initial and final images are reported respectively on Fig. 2e-f. First, Fig. 2b confirms that the control law converges since $\|e\|$ tends towards zero. One can also remark clearly on Fig. 2a the three steps of the algorithm (the last step begins near 10 s). For this experiment, the initial orientation of the camera was ${}^i\Phi = (-1.1^\circ, -15.1^\circ, 4.7^\circ)$ with ${}^iZ_P = 93.3$ cm. The orientation after servoing was ${}^m\Phi = (0.3^\circ, -0.9^\circ)$ (recall that we are not interested in the last component of Φ). Consequently, we obtained an accurate positioning. In addition, we performed the same task without using active vision in the second step. We obtained ${}^m\Phi = (-1.4^\circ, -0.9^\circ)$. The benefit of using active vision is clear since the positioning error is higher.

The second experiment concerns a positioning task with respect to the cylinder when ${}^d\Phi = \mathbf{0}$ and when using active vision. The initial orientation of the tangent plane w.r.t. the camera was ${}^i\Phi = (13.4^\circ, 8.7^\circ, 4^\circ)$ with ${}^iZ_P = 82.7$ cm. After servoing we obtained ${}^m\Phi = (0.8^\circ, 1.1^\circ)$. Here again, this result is better than when active vision is not used since we obtained in that case ${}^m\Phi = (1.6^\circ, 1.5^\circ)$. We also performed another experiment by setting ${}^d\Phi = (15^\circ, 0^\circ)$. Fig. 3 depicts the same parameters as in Fig. 2 and confirms that the control law converges without any problem. For this experiment we had ${}^i\Phi = (-1.5^\circ, -4.8^\circ, 4.7^\circ)$ and ${}^iZ_P = 86.7$ cm and we obtained ${}^m\Phi = (14.2^\circ, 1.9^\circ)$. Without active vision we obtained a very bad result since we had ${}^m\Phi = (13.0^\circ, 3.0^\circ)$.

The last experiment has been carried out on the sphere when ${}^d\Phi = (20^\circ, 20^\circ)$. Note that this object is more complex

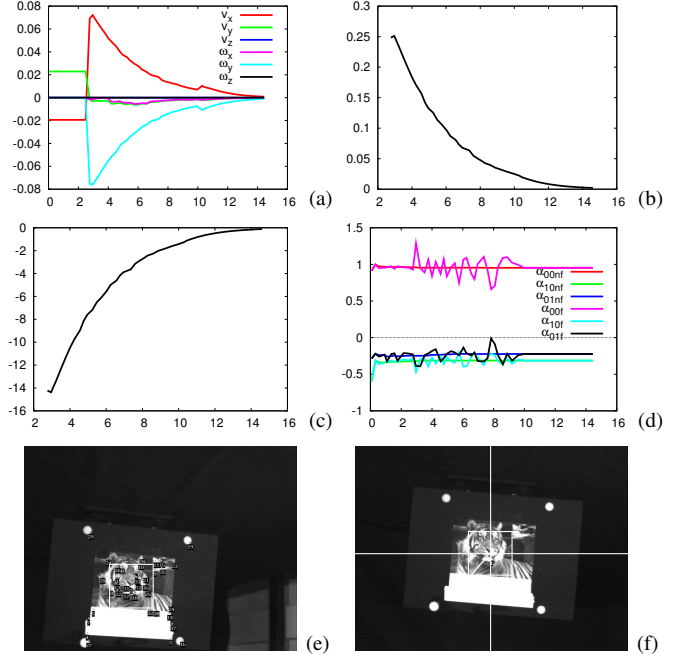


Fig. 2. 1st experiment (x axes in seconds). ${}^d\Phi = \mathbf{0}$. (a) v (m/s or rad/s). (b) Error defined as $\|e\|$. (c) Magnitude θ of the rotation (deg). (d) Vector α in a fixed frame (filtered and non-filtered). (e) Initial image. (f) Final image.

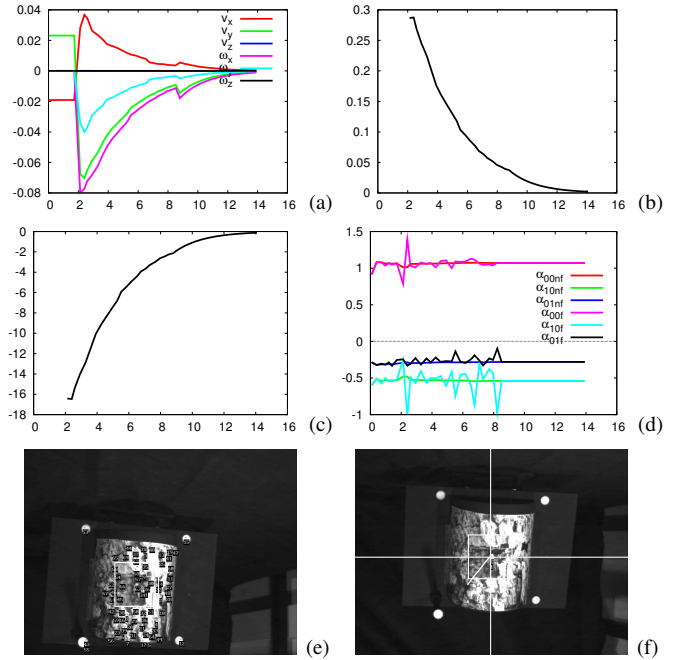


Fig. 3. 2nd experiment. Cylinder ($R = 7$ cm). ${}^d\Phi = (15^\circ, 0^\circ)$.

than the previous one since the curvature is higher. The initial orientation was ${}^i\Phi = (4.2^\circ, -5.2^\circ, 3.2^\circ)$ and we had ${}^iZ_P = 70.9$ cm. Whatever the approaches we used, with or without active vision, we obtained bad results (the orientation error was around 4° when using active vision). This can be caused by a too high value of constants r_2 or r_4 (since they depend on the curvature and on the pose, see (24) and (26)) leading to $v_z f(v_z) < 0$ and consequently to a bad value of v^* .

Finally, we performed another task with ${}^d\Phi = \mathbf{0}$. In that

case, a good orientation error has been obtained when using active vision ${}^m\Phi = (-0.1^\circ, 0.5^\circ)$, if not, we obtained a bad one: ${}^m\Phi = (-4.8^\circ, 2.8^\circ)$. For this experiment we had ${}^i\Phi = (-17^\circ, -17.2^\circ, -1.9^\circ)$ and ${}^iZ_P = 71.5$ cm. Fig. 4 describes the same parameters as for the previous experiment. One can see that similar curves has been obtained than in the case of the cylinder and that the convergence of the control law is obtained.

VII. CONCLUSION

We have presented in this paper a way to achieve visual servoing tasks when the desired visual features and the shape of the object being observed are unknown. To do that, we recover the parameters of the tangent plane at a certain point of an unknown object that are introduced in a control law to perform a positioning task. The approach we used is based on an indirect 3D reconstruction. Therefore, we have introduced an unified motion model which allows to deal as well with planar as with non-planar objects. However, since this model is an approximation of the true motion, we have proposed to use active vision during the servoing to enlarge its validity domain. Indeed, theoretical issues have shown that $v_z = 0$ minimizes the modeling error between both those models. Besides, experiments have validated that better results are obtained, whatever the shape of the object is, when active vision is used (i.e. $v_z = 0$). They have also shown that, very often, similar positioning errors are obtained for a planar object than for a non-planar object. Nevertheless, when the curvature of the object and the desired orientation are high, we have seen that a bad positioning could be obtained. To conclude, except this case, the only drawback of using active vision is that the final depth is imposed by the initial relative position between the camera and the object. However, remark that a desired depth could be reached during the last step. In addition, the computation cost of our approach is low (around 280 ms with a Pentium 4 at 2 Ghz) allowing the use of a 3D reconstruction based on a continuous approach.

REFERENCES

- [1] A. Alhaj, C. Collewet, and F. Chaumette, "Visual servoing based on dynamic vision," in *IEEE Int. Conf. on Robotics and Automation, ICRA'2003*, Taipei, Taiwan, September 14-19, 2003.
- [2] C. Collewet, A. Alhaj, and F. Chaumette, "Model-free visual servoing on complex images based on 3d reconstruction," in *IEEE Int. Conf. on Robotics and Automation, ICRA'2004*, New Orleans, USA, April 26-May 1, 2004, pp. 751-756.
- [3] F.-X. Espiau, E. Malis, and P. Rives, "Robust features tracking for robotic applications: towards 2d/2 visual servoing with natural images," in *IEEE Int. Conf. on Robotics and Automation, ICRA'2002*, Washington, USA, May 11-15, 2002.
- [4] A. Krupa, J. Gangloff, C. Doignon, M. de Mathelin, G. Morel, J. Leroy, L. Soler, and J. Marescaux, "Autonomous 3-d positioning of surgical instruments in robotized laparoscopic surgery using visual servoing," *IEEE Trans. on Robotics and Automation*, vol. 19, no. 5, pp. 842-853, October 2003.
- [5] C. Collewet and F. Chaumette, "Positioning a camera with respect to planar objects of unknown shape by coupling 2-d visual servoing and 3-d estimations," *IEEE Trans. on Robotics and Automation*, vol. 18, no. 3, pp. 322-333, June 2002.
- [6] M. Gouiffès, C. Fernandez-Maloigne, A. Trémeau, and C. Collewet, "Color segmentation of inked characters: application to meat traceability control," in *IEEE Int. Conf. on Image Processing, ICIP'2004*, Singapore, October 24-27, 2004.

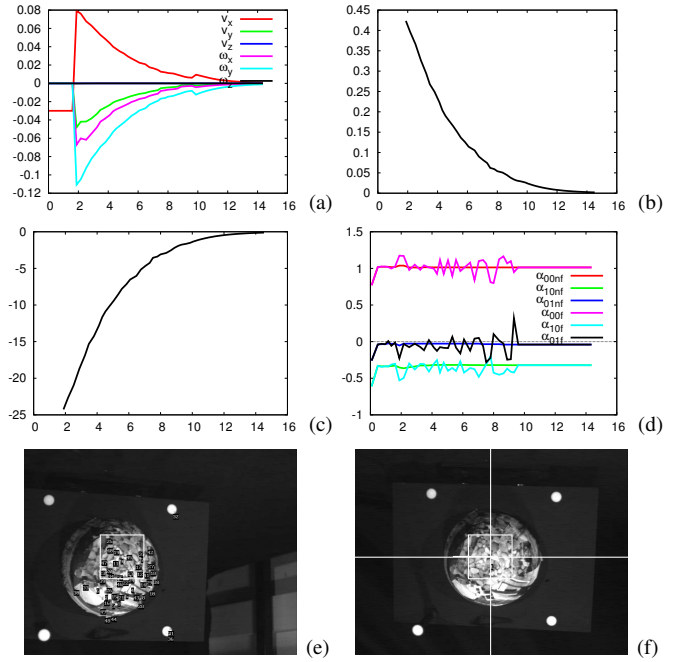


Fig. 4. 3rd experiment. Sphere ($R = 7$ cm). ${}^d\Phi = 0$.

- [7] S. Hutchinson, G. D. Hager, and P. I. Corke, "A tutorial on visual servo control," *IEEE Trans. on Robotics and Automation*, vol. 12, no. 5, pp. 651-670, October 1996.
- [8] B. Espiau, F. Chaumette, and P. Rives, "A new approach to visual servoing in robotics," *IEEE Trans. on Robotics and Automation*, vol. 8, no. 3, pp. 313-326, June 1992.
- [9] F. Schramm, G. Morel, A. Micaelli, and A. Lottin, "Extended-2d visual servoing," in *IEEE Int. Conf. on Robotics and Automation, ICRA'2004*, New Orleans, USA, April 26-May 1, 2004, pp. 267-273.
- [10] E. Malis and F. Chaumette, "Theoretical improvements in the stability analysis of a new class of model-free visual servoing methods," *IEEE Trans. on Robotics and Automation*, vol. 18, no. 2, pp. 176-186, April 2002.
- [11] W. J. Wilson, C. C. W. Hulls, and G. S. Bell, "Relative end-effector control using cartesian position based visual servoing," *IEEE Trans. on Robotics and Automation*, vol. 12, no. 5, pp. 684-696, October 1996.
- [12] J. Santos-Victor and G. Sandini, "Visual behaviors for docking," *Computer Vision and Image Understanding*, vol. 67, no. 3, pp. 223-238, September 1997.
- [13] A. Crétul and F. Chaumette, "Visual servoing based on image motion," *Int. Journal of Robotics Research*, vol. 20, no. 11, pp. 857-877, November 2001.
- [14] H. C. Longuet-Higgins and K. Prazdny, "The interpretation of a moving retinal image," in *Proceedings of the Royal Society of London*, vol. B208, 1980, pp. 385-397.
- [15] B. K. P. Horn and B. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 16, no. 1-3, pp. 185-203, August 1981.
- [16] B. K. P. Horn and E. J. Weldon, "Direct methods for recovering motion," *Int. Journal of Computer Vision*, vol. 2, no. 1, pp. 51-76, June 1988.
- [17] S. Negahdaripour and S. Lee, "Motion recovery from image sequences using only first order optical flow information," *Int. Journal of Computer Vision*, vol. 9, no. 3, pp. 163-184, 1992.
- [18] F. Chaumette, S. Boukir, P. Bouthemy, and D. Juvin, "Structure from controlled motion," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 18, no. 5, pp. 492-504, May 1996.
- [19] M. Gouiffès, C. Collewet, C. Fernandez-Maloigne, and A. Trémeau, "Feature points tracking: robustness to specular highlights and lighting changes," in *European Conf. on Computer Vision, ECCV'2006*, Graz, Austria, May 7-13, 2006.