Model-free visual servoing on complex images based on 3D reconstruction

Christophe Collewet, A. Alhaj, François Chaumette

Proceedings of the 2004 IEEE
International Conference on Robotics & Automation
New Orleans, LA • April 2004

Model-Free Visual Servoing on Complex Images Based on 3D Reconstruction

Christophe Collewet
Cemagref
17 Avenue de Cucillé
35 044 Rennes Cedex, France
Email: christophe.collewet@cemagref.fr

Ali Alhaj
Cemagref
17 Avenue de Cucillé
35 044 Rennes Cedex, France
Email: ali.alhaj@cemagref.fr

François Chaumette
IRISA / INRIA Rennes
Campus de Beaulieu
35 042 Rennes Cedex, France
Email: francois.chaumette@irisa.fr

***Abstract*— We present a way to achieve positioning tasks by model-free visual servoing in the case of planar and motionless objects whose shape is unknown. Emphasize is made on the algorithm of 3d reconstruction which allows to synthetize easily the control law. More precisely, the reconstruction phase is based on the measurement of the 2D displacements in a region of interest and on the measurement of the camera velocity. However, we will show that the proposed algorithm is robust with respect to nonaccurate values of this velocity. 2D displacements rather than 2D motions are used to remove the assumption that the acquisition rate has to be high. In addition, a particular attention is paid to the complex case of large displacements to access high camera velocities. Once the parameters of the plane are sufficiently stable, a visual servoing scheme is used to control the orientation of the camera with respect to the object and to ensure that it remains in the camera field of view for any desired orientation. The 3D reconstruction phase is maintained active during the servoing to improve the accuracy of the parameters and, consequently, to obtain a small positioning error. Experimental results validate the proposed approach.**

I. Introduction

This paper addresses the problem of synthetizing robotic tasks by visual servoing when the observed object is not known. Indeed, except rigid manufactured goods for which a model often exists, we rarely have a precise description of the object or of the desired visual features, either because these objects can be subject to deformations or simply because of their natural variability. Such cases can appear for example in surgical domain, agrifood industry, agriculture or in unknown environments (underwater, space). Few authors relate such cases. In [1], the authors use a specific motion to perform an alignment task without a precise description of the desired visual features. Unfortunately, their study is restricted to planar motions. In [2], thanks to dynamic visual features, a positioning task consisting in moving the camera to a position parallel to a planar object of unknown shape is achieved. However, such an approach needs an estimation of the parameters of a model of the 2D motion [3] currently leading to a high computation duration and, consequently, to a low control scheme rate. In addition, this approach cannot be used for any specified orientation of the camera. This case has been taken into account in [4], where geometric features are used. However, three tasks have to be performed sequentially yielding, in some cases, to excessive durations of the task. An approach based on robust tracking is proposed in [5] to obtain a projective reconstruction required for a hybrid visual servoing, unfortunately no result concerning robotic tasks is provided.

The approach described in this paper proposes to treat the same problem, that is the realization of positioning tasks with respect to a planar and motionless object of unknown shape for any specified orientation of the camera. Since the shape of the object is considered as unknown, a 3D reconstruction phase by *dynamic vision* is first performed. This computation is based, contrary to a previous approach [6], on a *discrete approach*. We will see the benefit of using such an approach. In particular, it does not require the assumption of a high acquisition rate and provides consequently better results. In addition, higher camera velocities can be reached. On the other hand, as in [6], the use of a reconstruction phase allows more flexibility to synthetize the control law, in particular to ensure that the object remains in the camera field of view.

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 and describe how to obtain the 2D displacement in Section IV. Section V details the way we synthetize the control law. Experimental results concerning objects of unknown shape are presented in Section VI, next we show in Section VII that, under some conditions, a simpler version of the approach can be used. Section VIII is devoted to the study of the influence of the camera velocity to the final orientation error. Finally, Section IX presents a comparative study with our previous approach [6].

II. Previous works

Let us consider a point P of the object described by $\underline{P} = (X, Y, Z)^T$ 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 p, described by $\underline{p} = (x, y, 1)^T$, according to

$$\underline{p} = \frac{\underline{P}}{Z} \tag{1}$$

which yields to the well-known relation [7]

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} -1/Z & 0 & x/Z & xy & -1-x^2 & y \\ 0 & 1/Z & y/Z & 1+y^2 & -xy & -x \end{pmatrix} T_c \tag{2}$$

where $T_c = (\underline{V}^T, \underline{\Omega}^T)^T$ is the camera velocity and $\underline{V} = (V_x, V_y, V_z)^T$ and $\underline{\Omega} = (\Omega_x, \Omega_y, \Omega_z)^T$ its translational and rotational components respectively. In (2), only the depth Z is unknown if $\underline{p}$, $\underline{\dot{p}}$ and T_c can be measured.

Various ways to estimate Z exist, they are based on different approaches to cope with $\underline{\dot{p}}$. The most immediate way is to approximate the velocities $\dot{x}$ ($\dot{y}$) by $\frac{\Delta x}{\Delta t}$ ($\frac{\Delta y}{\Delta t}$) [8]. However, this method does not provide accurate results because of errors introduced by the discretization. Another approach is based on the assumption that the brightness of p remains constant during the motion yielding the well-known additional constraint [7]

$$\dot{x}I_x + \dot{y}I_y + I_t = 0 \tag{3}$$

where I_x, I_y and I_t represent the spatio-temporal derivatives of the intensity of the considered point in the image. By substituting $\dot{x}$ and $\dot{y}$ given by (2) in (3), an expression of Z can be obtained [9], [10] (note that these works treat the more general case where T_c 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 way is to locally model the surface of the object in the neighborhood of P. 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. On the other hand, these parameters can be obtained by a method of computation of the 2D motion. Finally, an expression of the structure of the object can be extracted [11] (here too, by considering a second point, the case where T_c is unknown is treated). 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, including our previous one [6], implicitly assume that the acquisition rate is high (or the camera velocity low) enough so that the parameters of the model of the motion can be considered as constant between two frames.

The main benefit of the present approach is to remove this assumption as well as for the computation of the structure of the object as for the computation of the 2D displacement. Therefore, higher camera velocities can be reached. The approach is now relevant to *discrete approaches*. Moreover, using explicitly parameters obtained by 3D reconstruction allows us to synthetize easily the control law, in particular to take into account any desired orientation of the camera.

III. Structure of the object

Let us assume that the observed object is planar, unless in a neighborhood of P

$$Z_k = A_k X_k + B_k Y_k + C_k \tag{4}$$

which, according to (1), can be rewritten in function of $\underline{p}_k$ as follows

$$\frac{1}{Z_k} = \underline{\Theta}_k^T . \underline{p}_k \tag{5}$$

with $\underline{\Theta}_k = (\alpha_k, \beta_k, \gamma_k)^T$ where $\alpha_k = -A_k/C_k$, $\beta_k = -B_k/C_k$ and $\gamma_k = 1/C_k$.

On the other hand, let us assume that the camera is subjected to the velocity T_c, therefore $\underline{\dot{P}}$ can be expressed as

$$\underline{\dot{P}} + \underline{\Omega} \times \underline{P} = -\underline{V}. \tag{6}$$

Thereafter, by integrating this relation with respect to the time we obtain

$$\underline{P}_{k+1} = \mathcal{R}_k \underline{P}_k + \underline{\mathcal{T}}_k \tag{7}$$

where $\mathcal{R}_k$ is a rotation matrix depending on $\underline{\Omega}$ and $\underline{\mathcal{T}}_k$ a translation vector depending on T_c. Finally, by using (5) and (7) in (1) we recover the well-known result that an homographic model describes exactly the frame-to-frame displacement of a point P belonging to a planar surface [12]:

$$\begin{cases} x_{k+1} &= \dfrac{M_{11}x_k + M_{12}y_k + M_{13}}{M_{31}x_k + M_{32}y_k + 1} \\ y_{k+1} &= \dfrac{M_{21}x_k + M_{22}y_k + M_{23}}{M_{31}x_k + M_{32}y_k + 1} \end{cases} \tag{8}$$

that we will write under a more compact form as a parametric model

$$\underline{p}_{k+1} = \delta(\underline{p}_k, \underline{\mu}_k) \tag{9}$$

with $\underline{\mu}_k = (M_{13}, M_{23}, M_{11}, M_{21}, M_{12}, M_{22}, M_{31}, M_{32})$. We will see how to compute this vector in the next Section.

Furthermore, by using explicitely $\underline{\mu}_k$ in function of $\mathcal{R}_k$ and $\underline{\mathcal{T}}_k$, one can show that $\underline{\Theta}_k$ satisfies the following linear system

$$\mathcal{M}_k \underline{\Theta}_k = \underline{\Gamma}_k \tag{10}$$

with

$$\mathcal{M}_k = \begin{pmatrix} \mathcal{T}_1 & 0 & -M_{11}\mathcal{T}_3 \\ 0 & \mathcal{T}_1 & -M_{12}\mathcal{T}_3 \\ 0 & 0 & \mathcal{T}_1 - M_{13}\mathcal{T}_3 \\ \mathcal{T}_2 & 0 & -M_{21}\mathcal{T}_3 \\ 0 & \mathcal{T}_2 & -M_{22}\mathcal{T}_3 \\ 0 & 0 & \mathcal{T}_2 - M_{23}\mathcal{T}_3 \\ \mathcal{T}_3 & 0 & -M_{31}\mathcal{T}_3 \\ 0 & \mathcal{T}_3 & -M_{32}\mathcal{T}_3 \end{pmatrix}, \underline{\Gamma}_k = \begin{pmatrix} M_{11}\mathcal{R}_{33} - \mathcal{R}_{11} \\ M_{12}\mathcal{R}_{33} - \mathcal{R}_{12} \\ M_{13}\mathcal{R}_{33} - \mathcal{R}_{13} \\ M_{21}\mathcal{R}_{33} - \mathcal{R}_{21} \\ M_{22}\mathcal{R}_{33} - \mathcal{R}_{22} \\ M_{23}\mathcal{R}_{33} - \mathcal{R}_{23} \\ M_{31}\mathcal{R}_{33} - \mathcal{R}_{31} \\ M_{32}\mathcal{R}_{33} - \mathcal{R}_{32} \end{pmatrix}. \tag{11}$$

Finally, using a measure of the displacement between the frames k and $k+1$ modeled as an homographic deformation and the measure of the camera velocity, one can easily obtain $\underline{\Theta}_k$ by solving (10)

$$\underline{\Theta}_k = ({\mathcal{M}_k}^T \mathcal{M}_k)^{-1} {\mathcal{M}_k}^T \underline{\Gamma}_k. \tag{12}$$

Let us note that it is usually possible to obtain from (8) both the structure $\underline{\Theta}$ and the motion $(\mathcal{R}, \mathcal{T})$ [13], but in our case of small displacements (because of frame-to-frame displacements) the results will be not accurate enough to ensure correctly the realization of the task.

IV. Estimation of the frame-to-frame displacement

Let us consider two consecutive frames f and g and assume that the brightness of p_k remains unchanged during the motion, so we can write

$$f(\underline{p}) = g(\delta(\underline{p}, \underline{\mu})). \tag{13}$$

Because of the noise, (13) is generally not satisfied. Therefore, the solution is to move the problem to an optimization one to find the parameters which have to minimize the following criterion

$$\chi^2(\underline{\mu}) = \frac{1}{2} \sum_{\underline{x} \in W} \left(f(\underline{x}) - g(\delta(\underline{x}, \underline{\mu}) \right)^2 \tag{14}$$

where W is a window of interest centered in p_k.

To carry out the optimization, the classical approach [14], [15] is to assume that the acquisition rate and the displacements are sufficiently small. If so, a Taylor expansion of g can be performed. However, if we want to access high camera velocities this way to proceed cannot be used. To cope with this problem multi-scale [5] or multi-resolution approaches [16], [17] can be used. Nevertheless, these solutions remain time consuming.

Here, since T_c can be approximately known it can be used to provide an estimation $\widehat{\underline{\mu}}$ of $\underline{\mu}$ according to (9) (note that once $\widehat{\underline{\Theta}}$ is known, it can also be introduced in (9) to improve $\widehat{\underline{\mu}}$, otherwise a coarse approximation of $\underline{\Theta}$ is used). This way to proceed can be used in applications involving controlled motion, i.e. visual servoing or active vision.

Thereafter, it is now possible to perform a first order Taylor expansion of $g(\delta(\underline{x}, \underline{\mu}))$ in a neighborhood of $\widehat{\underline{\mu}}$ such that $\underline{\mu} = \widehat{\underline{\mu}} + \underline{\varsigma}$:

$$g(\delta(\underline{x}, \underline{\mu})) = g(\delta(\underline{x}, \widehat{\underline{\mu}})) + \nabla g^T (\delta(\underline{x}, \widehat{\underline{\mu}})) . {J_\delta}^{\mu} . \underline{\varsigma} \tag{15}$$

where ${J_\delta}^{\mu}$ represents the Jacobian matrix of δ with respect to $\underline{\mu}$.

Therefore, using (15) in (14) and derivating with respect to $\underline{\varsigma}$ leads to a linear system in $\underline{\varsigma}$. As usually, this system is inverted by using an iterative Newton-Raphson style algorithm to take into account the error introduced by the Taylor expansion. After some manipulations we obtain

$$\underline{\varsigma}_{k+1} = \underline{\varsigma}_k + \eta \left(\sum_W \underline{\Psi}_k \underline{\Psi}_k^T \right)^{-1} \left(f(\underline{p}_k) - g(\delta(\underline{x}_k), \underline{\mu}_k) \right) . \underline{\Psi}_k \tag{16}$$

where η is a positive scalar and $\underline{\Psi}$ the vector given by

$$\underline{\Psi} = (J_\delta^{\mu})^T . \nabla g(\delta(\underline{x}, \underline{\mu})). \tag{17}$$

In the case of a homographic model, one can show that $\underline{\Psi}$ can be expressed as follows

$$\underline{\Psi} = (G_x, G_y, xG_x, xG_y, yG_x, yG_y, xS_x, yS_y)^T \tag{18}$$

where $G_x = \frac{g_x}{M_{31}x + M_{32}y + 1}$ and $G_y = \frac{g_y}{M_{31}x + M_{32}y + 1}$ with g_x and g_y the coordinates of $\nabla g(\delta(\underline{x}, \underline{\mu}))$ and $S_x = -(\widehat{x}_{k+1} G_x + \widehat{y}_{k+1} G_y)$.

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 plane π described by (4) and, also to ensure that P will still remain in the camera field of view.

Once $\underline{\Theta}$ is estimated, the unit normal $\underline{n}$ of plane π in P in the camera frame can be derived. However, in the case of any orientation we rather have to consider $\underline{n}^* = \mathcal{R}\underline{n}$ where $\mathcal{R}$ is the rotation matrix computed from the desired orientation (see Figure 1). Therefore, we have to move the camera so that $\underline{Z} = \underline{n}_c$ with $\underline{Z}$ the unit vector carried by the optical axis and $\underline{n}_c = -\underline{n}^*$. This rotation to perform can be expressed under the form $\underline{u}\theta$ where $\underline{u}$ represents the rotation axis and θ the rotation angle around this axis

$$\underline{u} = \frac{\underline{Z} \wedge \underline{n}_c}{\| \underline{Z} \wedge \underline{n}_c \|} \tag{19}$$

and

$$\theta = \arccos(\underline{Z} . \underline{n}_c) \tag{20}$$

The camera orientation being known, it is possible to compute the control law. We used the one described in [6]. 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* [18].

First, $\underline{p}_r$ is defined as follows

$$\underline{p}_r = \frac{1}{Z^*} \underline{P} = \frac{Z}{Z^*} \underline{p} \tag{21}$$

with Z^* the desired depth for P in final position.

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

$$\underline{e} = \begin{pmatrix} \underline{p}_r - \underline{p}_r^* \\ \underline{u}\theta \end{pmatrix} \tag{22}$$

yielding to the camera velocity

$$T_c = -\lambda \widehat{L}^{-1} \underline{e} \tag{23}$$

λ being a positive gain and $\widehat{L}$ an approximation of the interaction matrix given by [18]

$$L = \begin{pmatrix} -\frac{1}{Z^*} \mathbb{I}_3 & [\underline{p}_r]_\times \\ 0 & L_\Omega \end{pmatrix} \tag{24}$$

where the notation $[\underline{v}]_\times$ denotes the antisymmetric matrix associated to $\underline{v}$ and

$$L_\Omega(\underline{u}, \theta) = \mathbb{I}_3 - \frac{\theta}{2} [\underline{u}]_\times + \left(1 - \frac{\mathrm{sinc}(\theta)}{\mathrm{sinc}^2(\theta/2)} \right) [\underline{u}]_\times^2 \tag{25}$$

with $\mathrm{sinc}(\theta) = \sin(\theta)/\theta$.

Let us note that the value of Z required for the computation of $\underline{p}_r$ is obtained thanks to (5).

VI. Experimental results

In order to validate the proposed algorithm, we present here experimental results for two different desired orientations. The experimental system is described in [4], except the PC is now a Pentium at 2 Ghz.

The object consists of a photograph of a raw ham fixed on a planar support. To evaluate the positioning accuracy of our method, this support makes possible to express precisely

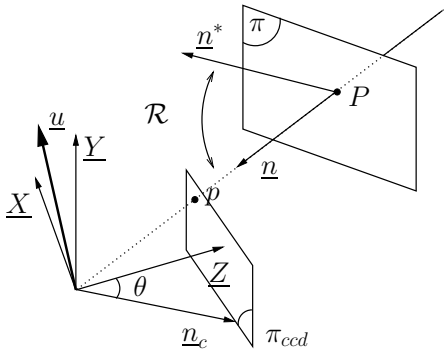

Fig. 1. Rotation to perform by the camera.

the transformation matrix between the camera frame and the object one with the method used in [4]. This matrix is characterized by the Euler's angles denoted ϕ_X, ϕ_Y, ϕ_Z which respectively represent the angles of the X, Y and Z rotations.

Furthermore, since the object is motionless, one can improve the accuracy on $\underline{\Theta}$. Indeed, in a fixed frame, one can express a value $\underline{\Theta}^f$ 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. Moreover, proceeding this way allows to know when $\underline{\Theta}^f$ is stable enough to be used in the control law (typically five acquisitions are sufficient). Thus, a preliminary phase is required.

Finally, the algorithm consists of three phases, a first phase at constant velocity, a second phase when both reconstruction and servoing are performed, and a last phase where only the servoing operates. This last phase occurs when the camera translation velocities are too small, in practice when $\| \underline{V} \| <$ 1 cm/s. Note that during this phase the camera velocity can be increased since W can be fixed to a lower value, p is therefore tracked by taking into account only the rigid terms of the model of displacement. This is done by considering $\underline{\Psi} = (g_x, g_y)^T$ in (16).

The following values have been used for all the experiments: Z^* = 65 cm, for the first and second phases $\lambda = 0.3$ (0.1 for the second experiment), $W = 171 \times 171$ pixels. The acquisition rate Δt is not constant, it varies from 200 ms to 720 ms. For the last phase $W = 11 \times 11$ pixels, Δt = 40 ms and $\lambda = 1$.

The first experiment consists in positioning the camera parallel to π. Figures 2.a depicts respectively the behavior of parameters A, B and C (filtered and nonfiltered) in a fixed frame; Figures 2.b the components of $\underline{n}_c$; Figures 2.c the components of the camera velocity; Figure 2.d the norm of $\underline{e}$; Figure 2.e the parameters M_{13} and M_{23} (in pixel); Figure 2.f the estimated depth. Finally, the initial and final images are reported respectively on Figures 2.g-h. Note that the unit of x axes is second. First, Figure 2.d confirms that the control law converges since $\| \underline{e} \|$ tends towards zero, as well as $\widehat{Z}$ towards Z^* (Figure 2.f). One can also remark on Figures 2.a and 2.c the three phases of the algorithm, the last phase begins near 8.5 s when the camera velocity is too low to provide accurate estimations of $\underline{\Theta}$ as shown clearly on Figure 2.a. One can also remark the benefit of using an estimation of $\underline{p}_{k+1}$ in the algorithm of computation of the displacement since high values are observed in Figure 2.e. For this experiment, the initial orientation of the camera was $\phi_X = 3.3°$, $\phi_Y = 12.6°$ and $\phi_Z = -2.5°$, the orientation after the servoing was $\phi_X = 2.94°$ and $\phi_Y = 0.0°$ (let us recall that ϕ_Z is not controlled). Consequently, the orientation error is around 3°.

The second experiment consists in positioning the camera so that $\phi_X = 10°$ and $\phi_Y = -20°$. Figures 3.a-h describe the behavior of the same variables as for the previous experiment. The same comments can be made, in particular concerning the convergence of the control law and the benefit of using an estimation of $\underline{p}_{k+1}$. The initial orientation of the camera was the same as for the first experiment, the final orientation was $\phi_X = 13.1°$ and $\phi_Y = -21.9°$. Here again the orientation error is around 3°.

VII. A SIMPLER VERSION OF THE ALGORITHM

In the case of a planar object and when the desired values for ϕ_X and ϕ_Y are small, the parameters M_{31} and M_{32} involved in (8) are very close to zero. In this case, we rather have to consider them direclty as zero in (8) leading to

$$\mathcal{M}_k = \begin{pmatrix} \mathcal{T}_1 & 0 & -M_{11}\mathcal{T}_3 \\ 0 & \mathcal{T}_1 & -M_{12}\mathcal{T}_3 \\ 0 & 0 & \mathcal{T}_1 - M_{13}\mathcal{T}_3 \\ \mathcal{T}_2 & 0 & -M_{21}\mathcal{T}_3 \\ 0 & \mathcal{T}_2 & -M_{22}\mathcal{T}_3 \\ 0 & 0 & \mathcal{T}_2 - M_{23}\mathcal{T}_3 \end{pmatrix}, \underline{\Gamma}_k = \begin{pmatrix} M_{11}\mathcal{R}_{33} - \mathcal{R}_{11} \\ M_{12}\mathcal{R}_{33} - \mathcal{R}_{12} \\ M_{13}\mathcal{R}_{33} - \mathcal{R}_{13} \\ M_{21}\mathcal{R}_{33} - \mathcal{R}_{21} \\ M_{22}\mathcal{R}_{33} - \mathcal{R}_{22} \\ M_{23}\mathcal{R}_{33} - \mathcal{R}_{23} \end{pmatrix} \quad (26)$$

in (10).

The computation of the frame-to-frame displacements becomes also simpler since $\delta(.)$ describes an affine deformation, consequently we have to consider in (16)

$$\underline{\Psi} = (g_x, g_y, xg_x, xg_y, yg_x, yg_y)^T \quad (27)$$

and $\underline{\mu} = (M_{13}, M_{23}, M_{11}, M_{21}, M_{12}, M_{22})$.

We performed an experiment consisting in moving the camera in front of the object. The behavior of the algorithm is depicted on figure 4 where the same variables as previously are used.

For this experiment we used the same parameters as for the previous experiments except W is now 101 $\times$101 pixels. The acquisition rate varies now from 120 ms to 200 ms. The initial orientation was $\phi_X = 8.8°$ and $\phi_Y = -14.6°$. Here again, the final orientation error is small, since we have $\phi_X = 1.5°$ and $\phi_Y = -1.3°$.

To conclude, similar results can be obtained with this approach by comparison with the complete model defined by (8) when an affine model is valid[1]. However, the image processing is less time consuming since W and $\sum \underline{\Psi}_k \underline{\Psi}_k^T$ is smaller, that leads globally to a better dynamic behavior of

[1]That is essentially when the desired orientation is close to zero and the desired depth is high.

the robot. Note that in the case of an high initial orientation error the control law converges even if $\underline{\widehat{\Theta}}_{k=0}$ is not accurate since $\underline{\widehat{\Theta}}_k$ refines during the motion.

VIII. Influence of the estimation of the camera velocity

Since $\mathcal{M}$ and Γ given by (11) depend on $\mathcal{R}$ and $\mathcal{T}$ which both depend on the camera velocity T_c, $\underline{\Theta}$ also depends on the camera velocity. To measure the influence of T_c on the estimation of the structure of the object we have to compute $\partial\underline{\Theta}/\partial T_c$. However, it leads to very complicated expressions in the case of any values for the components of T_c. Nevertheless, since in practice $\underline{\Theta}$ is filtered (see VI and VII), the perturbations due to the noise introduced by an error of T_c will be also filtered. Figure 5 show simulations results of the behavior of the parameters A, B and C involved in (4) when T_c is given by (23). More precisely, the figure 5.a compares the filtered values of A, B and C with the measured one; the figure 5.b compares the filtered values of A, B and C with the real one. Figures 5.c-d represent the same variables but when a random noise of 20 % has been added to T_c. As we can see, filtering dicreases highly the noise introduced by T_c.

IX. Comparison between the continuous and the discrete approach

First, let us recall the continuous approach [6], [11]. By substituting (5) in (2), one can show that the 2D motion is exactly modeled by a parametric model with 8 parameters [19]. By neglecting the second order terms, we obtain

$$\begin{pmatrix} \dot{x} \\ \dot{y} \end{pmatrix} = \begin{pmatrix} a_0 + a_1 x + a_2 y \\ a_3 + a_4 x + a_5 y \end{pmatrix} \tag{28}$$

where

$$\begin{cases} a_0 = -\Omega_y - \gamma V_x \\ a_1 = -\alpha V_x + \gamma V_z \\ a_2 = \Omega_z - \beta V_x \\ a_3 = \Omega_x - \gamma V_y \\ a_4 = -\Omega_z - \alpha V_y \\ a_5 = -\beta V_y + \gamma V_z \end{cases} \tag{29}$$

Consequently, if the parameters involved in (28) and the camera velocity can be measured, $\underline{\Theta}$ can be obtained.

In continuous approaches the problem is to obtain precisely the a_i's. They can be obtained by measuring the displacement between two frames by considering the a_i's constant during Δt. In fact it is not true since $\underline{\Theta}$ is not constant during the motion. Thus, with such approaches, we have to consider high acquisition rates or small displacements.

On the other hand, once (29) has been integrated, we obtain an affine relation between p_{k+1} and p_k that can be used to express the a_i's by substituting $\dot{x} = (x_{k+1} - x_k)/\Delta t$ and $\dot{y} = (y_{k+1} - y_k)/\Delta t$ in (29). Here again, these relations hold only for small values of Δt.

Note that the discrete approaches presented in Sections III and VII do not depend at all of Δt regardless of the reconstruction of the structure or of the estimation of the frame-to-frame displacement since we have an estimation of $\underline{\mu}$. Moreover, recall that we can use with these discrete approaches high camera velocities in order to decrease the duration of the task. The drawback of using such velocities is that we have less values to filter leading to similar positioning errors than with our previous approach [6].

X. Conclusion and future works

We have presented a way to achieve positioning tasks by visual servoing when the desired image of the object cannot be precisely described and for any desired orientation of the camera assuming the object to be planar and motionless. The approach is based on a 3D reconstruction allowing the estimation of the current orientation of the object with respect to the camera, and thereafter on the elaboration of the control law. The special case of high camera velocities has been studied. Experimental results validated our algorithm, low orientation errors were observed ($\approx 3^o$). We also showed the robustness of the approach with respect to a coarse estimation of the camera velocity.

Future works will concern the realization of positioning tasks in the case of unknown and nonplanar objects.

References

[1] B. Yoshimi and P. K. Allen, "Active uncalibrated visual servoing," in *IEEE Int. Conf. on Robotics and Automation, ICRA'94*, San Diego, May 1994, pp. 156–161.

[2] A. Crétual and F. Chaumette, "Visual servoing based on image motion," *Int. Journal of Robotics Research*, vol. 20, no. 11, pp. 857–877, November 2001.

[3] J. Odobez and P. Bouthemy, "Robust multiresolution estimation of parametric motion models," *Journal of Visual Communication and Image Representation*, vol. 6, no. 4, pp. 348–365, December 1995.

[4] 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.

[5] F. Espiau, E. Malis, and P. Rives, "Robust features tracking for robotic applications: towards 2d1/2 visual servoing with natural," in *IEEE Int. Conf. on Robotics and Automation, ICRA'2002*, Washington, USA, May 11-15, 2002.

[6] 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.

[7] B. Horn and B. Schunck, "Determining optical flow," *Artificial Intelligence*, vol. 16, no. 1–3, pp. 185–203, August 1981.

[8] P. Rives and M. Xie, "Towards dynamic vision," in *Proc. IEEE Workshop on Interpretation of 3D scenes*, Austin, Texas, November 1989.

[9] 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.

[10] S. Negahdaripour and B. K. P. Horn, "Direct passive navigation," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 9, no. 1, pp. 168–176, January 1987.

[11] 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.

[12] O. Faugeras, *Three-dimensional computer vision: a geometric viewpoint*. Boston: MIT Press, 1993.

[13] T. Huang and A. Netravali, "Motion and structure from feature correspondences: A review," *Proceedings of IEEE*, vol. 82, no. 2, pp. 252–268, February 1994.

[14] J. Shi and C. Tomasi, "Good features to track," in *IEEE Int. Conf. on Computer Vision and Pattern Recognition, CVPR'94*, Seattle, USA, June 1994, pp. 593–600.

[15] G. D. Hager and P. N. Belhumeur, "Efficient region tracking with parametric models of geometry and illumination," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 20, no. 10, pp. 1025–1039, 1998.

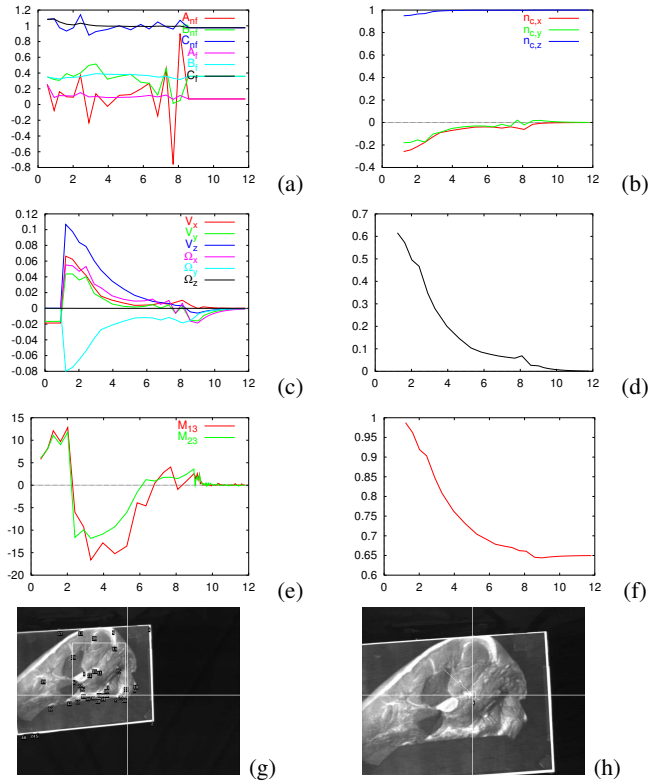

Fig. 2. 1st experiment (x axes in seconds) : (a) Parameters of the plane in a fixed frame (filtered and nonfiltered). (b) Components of $\underline{n}_c$. (c) Kinematic screw (m/s or rad./s). (d) Error defined as $\| \underline{e} \|$. (e) Rigid terms of the 2D displacement model (pixel). (f) Desired depth (meter). (g) Initial image. (h) Final image.

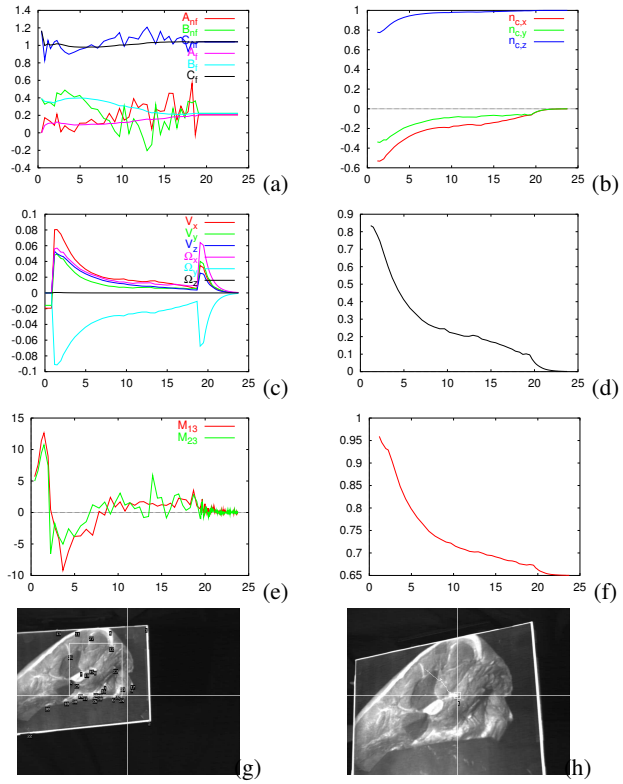

Fig. 3. 2nd experiment.

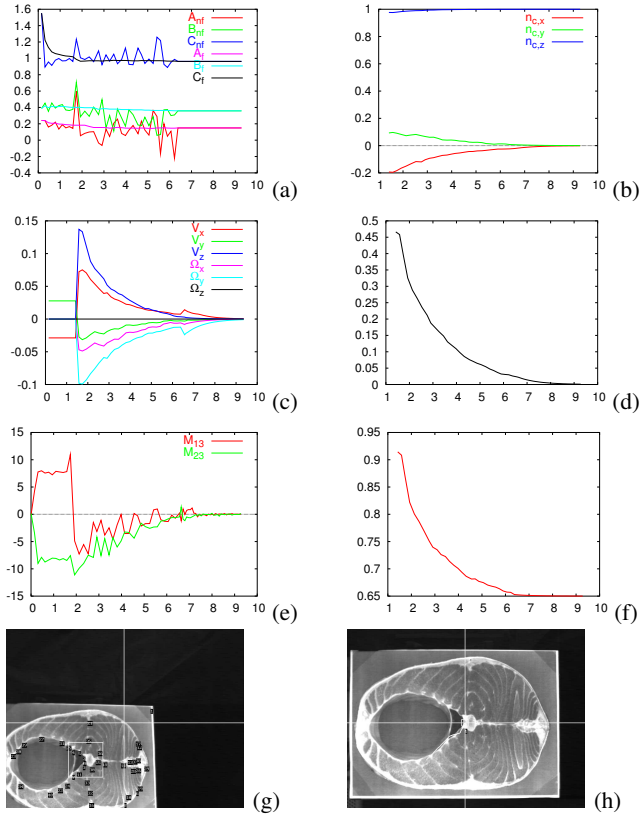

Fig. 4. Case of an affine deformation.

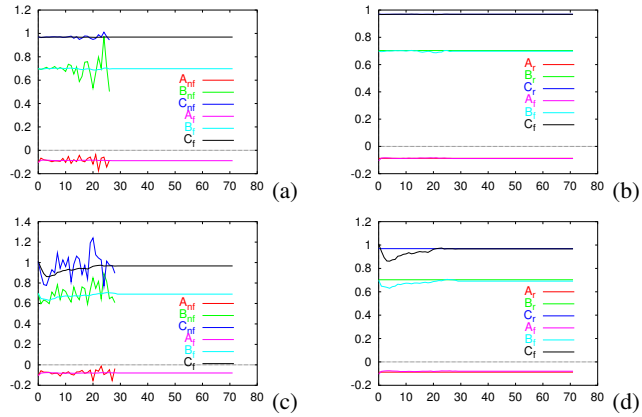

Fig. 5. Influence of the accuracy of the estimation of the camera velocity.

[16] M. Irani, B. Rousso, and S. Peleg, "Recovery of ego-motion using image stabilization," *IEEE Int. Conf. on Computer Vision and Pattern Recognition, CVPR'94*, vol. 19, pp. 454–460, June 1994.

[17] G. Hager and K. Toyama, "Incremental focus of attention for robust visual tracking," *International Journal of Computer Vision*, vol. 35, no. 1, pp. 45–63, November 1999.

[18] 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.

[19] G. Adiv, "Determining 3d motion and structure from optical flow generated by several moving objects," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 7, no. 4, pp. 384–401, July 1985.