Path Planning in Image Space for Robust Visual Servoing

Youcef Mezouar
Youcef.Mezouar@irisa.fr

François Chaumette
Francois.Chaumette@irisa.fr

IRISA - INRIA Rennes
Campus de Beaulieu,
35042 Rennes Cedex, France

Abstract

Vision feedback control loop techniques are efficient for a great class of applications but they come up against difficulties when the initial and desired positions of the camera are distant. In this paper we propose a new approach to resolve these difficulties by planning trajectories in the image. Constraints such that the object remains in the camera field of view can thus be taken into account. Furthermore, using this process, current measurement always remain close to their desired value and a control by Image-based Servoing ensures the robustness with respect to modeling errors. We apply our method when object dimension are known or not and/or when the calibration parameters of the camera are well or badly estimated. Finally, real time experimental results using a camera mounted on the end effector of a six d-o-f robot are presented.

1 Introduction

Visual servoing is classified into two main approaches [15, 6, 8]. The first one is called Position-based Control (PbC) or 3D visual servoing. In PbC the control error function is computed in the Cartesian space. Image features are extracted from the image and a perfect model of the target is used to determine its position with respect to camera frame. The main advantage of this approach is that it controls the camera trajectory directly in Cartesian space. However there is no control in the image space and the object may get out of the camera field of view during servoing. Furthermore, it is impossible to analytically demonstrate the stability of the system in presence of modeling errors. Indeed, the sensitivity of pose estimation algorithm with respect to calibration errors and measurement perturbations is not available [2].

The second approach is called Image-based Control (IbC) or 2D visual servoing. In IbC the pose estimation is omitted and the control error function is computed in the image space. The IbC approach does not need a precise calibration and modeling since a closed loops scheme is performed. However, the stability is theoretically ensured only in the neighborhood of the desired position. Therefore, if initial and desired configurations are closed, IbC is robust with respect to measurement and modeling errors. Otherwise, that is if desired and initial position are distant, the stability is not ensured and the object can get out of the camera field of view [2]. Control laws taking into account this last constraint have been proposed for example in [13, 12]. We propose in this paper a more robust approach.

A third approach is described in [11] and is called *2 1/2 D* visual servoing. In this case the control error function is computed in part in the Cartesian space and in part in the 2D image space. An homography, computed at each iteration, is used to extract the Cartesian part of the error function. Hence, this method does not need a model of the target. Contrarily to the previous approaches, it is possible to obtain analytical results about stability with respect to modeling and calibration errors. However, the main drawback of *2 1/2 D* visual servoing is its relative sensitivity to measurement perturbations. Furthermore, keeping all the object in the camera field of view is not obvious.

In this paper, a new method, robust and stable even if initial and desired positions are distant, is described. The method consists in planning trajectories of a set of n points lying on the target in image space and then tracking these trajectories by 2D visual servoing (see Figure 1). Using this process, current measurements always remain close to their desired value . Thus the good behavior of IbC in such configuration can be exploited. Moreover, it is possible to ensure that the object will always remain in the camera field of view by enforcing such constraint on the trajectories.

There are few papers dealing with path planning in image space. In [7] a trajectory generator using a stereo system is proposed and applied to obstacle avoidance. In [14] an alignment task is realized using intermediate view of the target synthesized by image morphing. However, none of them were dealing with robustness issues. Our path planning strategy is based on the potential field method. This method was originally developed for an on-line collision avoidance [9, 10]. In this approach the robot motions are under the influence of an artificial potential field (V) defined as the sum of an attractive potential (V_a) pulling the robot toward the goal configuration ($\boldsymbol{\Upsilon}_g$) and a repulsive potential (V_r) pushing the robot away from the obstacles. Motion planning is performed in an iterative fashion. At each iteration an artificial force $\mathbf{F}(\boldsymbol{\Upsilon})$, where the 6×1 vector $\boldsymbol{\Upsilon}$ represents a parameterization of robot workspace, is induced by the potential function. This force is defined as $\mathbf{F}(\boldsymbol{\Upsilon}) = -\vec{\nabla} V$ where $\vec{\nabla} V$ denotes the gradient vector of V at $\boldsymbol{\Upsilon}$. Using these conventions, $\mathbf{F}(\boldsymbol{\Upsilon})$ can be decomposed as the sum of two vectors, $\mathbf{F}_a(\boldsymbol{\Upsilon}) = -\vec{\nabla} V_a$ and $\mathbf{F}_r(\boldsymbol{\Upsilon}) = -\vec{\nabla} V_r$, which are called the attractive and repulsive forces respectively. Path generation proceeds along the direction of $\mathbf{F}(\boldsymbol{\Upsilon})$ and the discrete-time trajectory is given by the transition equation :

$$\boldsymbol{\Upsilon}_{k+1} = \boldsymbol{\Upsilon}_k + \varepsilon_k \frac{\mathbf{F}(\boldsymbol{\Upsilon}_k)}{\|\mathbf{F}(\boldsymbol{\Upsilon}_k)\|} \tag{1}$$

where k is the increment index and ε_k a positive scaling factor denoting the length of the k^{th} increment.

The paper is organized as follows. We describe in Section 2 the method when a model of the target and the calibration of the camera are available. We present in Section 3 how we proceed if the object is planar but neither a model of the target and neither accurate calibration are available. In Section 4 we use the task function approach to track the trajectories. Experimental results are finally given in Section 5.

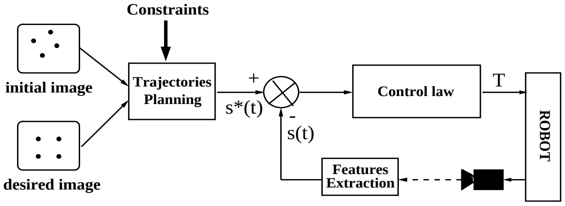

Figure 1: Block diagram of the method

2 Known target

Here, we assume that the calibration parameters and a target model are available. The technique consists in planning camera frame trajectory bringing it from initial camera frame $\mathcal{F}_i$ ($\boldsymbol{\Upsilon} = \boldsymbol{\Upsilon}_i$) to desired camera frame $\mathcal{F}_g$ ($\boldsymbol{\Upsilon} = \boldsymbol{\Upsilon}_g$) and then to project the target model in the image along the trajectory. Let ${}^g\mathbf{R}_c$, ${}^g\mathbf{t}_c$, $\mathbf{u}$ and θ be respectively the rotational matrix and the translational matrix between the current camera frame $\mathcal{F}_c$ and $\mathcal{F}_g$, the rotation axis and the rotation angle obtained from ${}^g\mathbf{R}_c$. We choose as parameterization of the workspace $\boldsymbol{\Upsilon}^T = [{}^g\mathbf{t}_c^T \ (\mathbf{u}\theta)^T]$. We thus have $\boldsymbol{\Upsilon}_i^T = [{}^g\mathbf{t}_i^T \ (\mathbf{u}\theta)_i^T]$ and $\boldsymbol{\Upsilon}_g^T = \mathbf{0}_{6\times 1}$. Using a pose estimation algorithm [3], we can determine ${}^i\mathbf{R}_o$, ${}^i\mathbf{t}_o$, ${}^g\mathbf{R}_o$ and ${}^g\mathbf{t}_o$ that represent respectively the rotation and the translation from object frame $\mathcal{F}_o$ to $\mathcal{F}_i$ and $\mathcal{F}_o$ to $\mathcal{F}_g$ (see Figure 3). The vector $\boldsymbol{\Upsilon}_i$ is then computed using the following relations :

$$\left\{ \begin{array}{lcl} {}^g\mathbf{R}_i & = & {}^g\mathbf{R}_o \, {}^i\mathbf{R}_o^T \\ {}^g\mathbf{t}_i & = & -\, {}^g\mathbf{R}_i \, {}^i\mathbf{t}_o + {}^g\mathbf{t}_o \end{array} \right.$$

According to (1) we construct a path as the sequence of successive path segments starting at the initial configuration $\boldsymbol{\Upsilon}_i$. We now present how the potentials functions and the induced forces are defined and calculated.

Attractive potential and force. The attractive potential field V_a is simply defined as a parabolic function in order to minimize the distance between the current position and the desired one :

$$V_a = \frac{1}{2} a \|\boldsymbol{\Upsilon} - \boldsymbol{\Upsilon}_g\|^2 = \frac{1}{2} a \|\boldsymbol{\Upsilon}\|^2$$

where a is a positive scaling factor. The attractive force deriving from V_a is :

$$\mathbf{F}_a(\boldsymbol{\Upsilon}) = -\vec{\nabla} V_a = -a\boldsymbol{\Upsilon} \tag{2}$$

Repulsive potential and force. A point $\mathcal{P}_j$, which projects onto the camera's image plane at a point with image coordinates $\mathbf{p}_j = [u_j \, v_j \, 1]^T$, is observable by the camera if $u_j \in [u_m \, u_M]$ and $v_j \in [v_m \, v_M]$, where u_m, u_M, v_m, v_M are the limits of the image (see Figure 2). One way to create a potential barrier around the camera field of view, assuring that all features are always observable and do not affect the camera motion when they are sufficiently far away from the image limits, is to define the repulsive potential V_r as follow (see Figure 2) :

$$V_r(\mathbf{s}) = \tfrac{1}{2} \log \left(\prod_{j=1}^{n} \frac{1}{(1-\frac{u_j}{u_M})(1-\frac{u_j}{u_m})} \frac{1}{(1-\frac{v_j}{v_M})(1-\frac{v_j}{v_m})} \right)$$

$$\text{if } \mathbf{s} \in C \text{ and } V_r(\mathbf{s}) = 0 \text{ else.} \tag{3}$$

where $\mathbf{s}$ is the vector made up of the coordinates $u_{j=1..n}$, $v_{j=1..n}$, C is the set $\{ \mathbf{s} / \exists j \;\; u_j \in [u_m \;\, \alpha] \cap [u_M - \alpha \;\, u_M]$ or $v_j \in [v_m \;\, \alpha] \cap [v_M - \alpha \;\, v_M] \}$, α being a positive constant denoting the distance of influence of the image edges.

The artificial repulsive force deriving from V_r is :

$$\mathbf{F}_r(\boldsymbol{\Upsilon}) = -\left(\frac{\partial V_r(\mathbf{s})}{\partial \boldsymbol{\Upsilon}} \right)^T = -\left(\frac{\partial V_r(\mathbf{s})}{\partial \mathbf{s}} \frac{\partial \mathbf{s}}{\partial \mathbf{r}} \frac{\partial \mathbf{r}}{\partial \boldsymbol{\Upsilon}} \right)^T$$

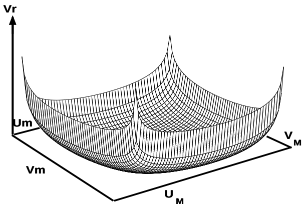

Figure 2: Repulsive potential

where $\mathbf{r}$ denote the situation of the camera with respect to a reference frame. The previous equation can be written :

$$\mathbf{F}_r(\mathbf{\Upsilon}) = -\mathbf{M}_{\Upsilon}^T \mathbf{L}^T \left(\frac{\partial V_r(\mathbf{s})}{\partial \mathbf{s}} \right)^T$$

where :

- $\mathbf{L} = \frac{\partial \mathbf{s}}{\partial \mathbf{r}}$ is the image Jacobian (or interaction matrix) [4]. It relates the variation of image feature $\mathbf{s}$ to the velocity screw of the camera $\mathbf{T}$: $\dot{\mathbf{s}} = \mathbf{LT}$. The well known interaction matrix for a point $\mathcal{P}$ with coordinate $(X\,Y\,Z)$ in camera frame and coordinates $\mathbf{p} = (x\,y)$ in the image expressed in meters, for a one meter focal length is :

$$\mathbf{L}(\mathbf{p}, Z) = \begin{pmatrix} -\frac{1}{Z} & 0 & \frac{x}{Z} & xy & -(1+x^2) & y \\ 0 & -\frac{1}{Z} & \frac{y}{Z} & (1+y^2) & -xy & -x \end{pmatrix}$$

When $\mathbf{s}$ is composed of the image coordinates of n points the corresponding interaction matrix is :

$$\mathbf{L}(\mathbf{s}, \mathbf{Z}) = \left[\mathbf{L}^T(p_1, Z_1) \;...\; \mathbf{L}^T(p_n, Z_n) \right]^T \qquad (4)$$

- $\mathbf{M}_{\Upsilon} = \frac{\partial \mathbf{r}}{\partial \mathbf{\Upsilon}}$ is the 6×6 Jacobian matrix that relates the variation of $\mathbf{r}$ to the variation of $\mathbf{\Upsilon}$:

$$\mathbf{M}_{\Upsilon} = \begin{bmatrix} {}^g\mathbf{R}_c^T & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & L_w^{-1} \end{bmatrix}$$

where [11] :

$$L_w^{-1} = \mathbf{Id_{3\times3}} + \frac{\theta}{2}\text{sinc}^2\left(\frac{\theta}{2}\right)[\mathbf{u}]_\wedge + (1 - \text{sinc}(\theta))[\mathbf{u}]_\wedge^2$$

$[\mathbf{u}]_\wedge$ being the antisymmetric matrix of cross product associated to $\mathbf{u}$

- $\frac{\partial V_r(\mathbf{s})}{\partial \mathbf{s}}$ is easily obtained according to (3).

Let us note, using (2), (3) and (1), we obtain a camera trajectory in the workspace. A PbC could thus be used to follow it. However, it is more interesting to perform features trajectories in image in order to exploit the good behavior of IbC when the current and desired camera positions are close.

2D trajectories. Let ${}^g\mathbf{R}_k$, ${}^g\mathbf{t}_k$ and ${}^k\mathbf{R}_o$, ${}^k\mathbf{t}_o$ be the rotations and translations mapping $\mathcal{F}_k$ with $\mathcal{F}_g$ and $\mathcal{F}_o$ with $\mathcal{F}_k$, where $\mathcal{F}_k$ is the camera frame position at iteration k of the path planning. With these notations we have :

$$\begin{cases} {}^k\mathbf{R}_o &= {}^g\mathbf{R}_k^T \, {}^g\mathbf{R}_o \\ {}^k\mathbf{t}_o &= {}^g\mathbf{R}_k^T({}^g\mathbf{t}_o - {}^g\mathbf{t}_k) \end{cases}$$

In order to perform visual servo control, we construct the trajectory of the projection $\mathbf{p}_j$ of each point $\mathcal{P}_{j=1..n}$ onto the image using the known coordinates ${}^o\mathbf{P}_j$ of $\mathcal{P}_j$ in $\mathcal{F}_g$. The trajectory in image is obtained using the classical assumption that the camera performs a perfect perspective transformation with respect to the camera optic center (pinhole model) :

$$\mathbf{p}_{j,k} = \mathbf{A}[{}^k\mathbf{R}_o \;\; {}^k\mathbf{t}_o] \, {}^o\mathbf{P}_j$$

where $\mathbf{A}$ is the matrix of camera intrinsic parameters. In the next part, we extend this method to the case where the target model is unknown.

3 Unknown planar target

In this section, we assume that the target is planar but the target model is not available. After recalling the relations between two views of a planar target, we present the method with accurate calibration parameters and then we prove its robustness with respect to calibration error.

3.1 Euclidean reconstruction

Consider a reference plane Π given in desired camera frame ($\mathcal{F}_g$) by the vector $\pi^T = [\mathbf{n}^{*T} - d^*]$, where $\mathbf{n}^*$ is its unitary normal in $\mathcal{F}_g$ and d^* the distance from Π to the origin of $\mathcal{F}_g$ (see Figure 3). It is well known [5] that the projection of point $\mathcal{P}_j$ lying on Π in current view $\mathbf{p}_j = [u_j\; v_j\; 1]^T$ and in the desired view $\mathbf{p}_j^* = [u_j^*\; v_j^*\; 1]^T$ are linked by the projective relation :

$$\mu_j \mathbf{p}_j = \mathbf{G}_\pi \mathbf{p}_j^* \qquad (5)$$

where $\mathbf{G}_\pi$ is a projective homography, expressed in pixels, of plane Π between the current and desired images and μ a scaling factor. We can estimate it from a set of $N \geq 8$ points (three points defining Π) in general case or from a set of $N \geq 4$ points belonging to Π [11, 5]. Assuming that the camera calibration is known, the Euclidean homography $\mathbf{H}_\pi$ is computed as follows :

$$\mathbf{H}_\pi = \mathbf{A}^{-1}\mathbf{G}_\pi\mathbf{A} \qquad (6)$$

The matrix $\mathbf{H}_\pi$ can be decomposed using motion parameters between $\mathcal{F}_g$ and $\mathcal{F}_c$ [5]:

$$\mathbf{H}_\pi = {}^c\mathbf{R}_g + \frac{{}^c\mathbf{t}_g}{d^*}\mathbf{n}^{*T} = {}^g\mathbf{R}_c^T - {}^g\mathbf{R}_c^T\mathbf{t}_{d^*}\mathbf{n}^{*T} \qquad (7)$$

From $\mathbf{H}_\pi$ it is possible to compute ${}^g\mathbf{R}_c$, $\mathbf{t}_{d^*} = \frac{{}^g\mathbf{t}_c}{d^*}$, and $\mathbf{n}^*$ using for example the algorithm presented in [5]. The ratio ρ_j between the coordinate Z_j of a point lying on Π, with respect to camera frame, and d^*, that we will use in the continuation, can also be determined [11] :

$$\rho_j = \frac{Z_j}{d^*} = \frac{1 + \mathbf{n}^{*T}\,{}^c\mathbf{R}_g^T({}^c\mathbf{t}_g/d^*)}{\mathbf{n}^{*T}\,{}^c\mathbf{R}_g^T\mathbf{A}^{-1}\mathbf{p}_j} \tag{8}$$

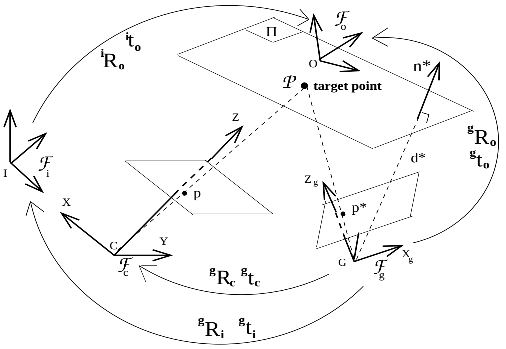

Figure 3: Euclidean reconstruction

3.2 Trajectory planning

We now choose the partial parameterization of the workspace as $\boldsymbol{\Upsilon}^T = [\mathbf{t}_{d^*}^T\ (\mathbf{u}\theta)^T]$. We thus have $\boldsymbol{\Upsilon}_i^T = [\mathbf{t}_{d^*i}^T\ (\mathbf{u}\theta)_i^T]$ and $\boldsymbol{\Upsilon}_g = \mathbf{0}_{6\times1}$. From initial and desired images, it is possible to compute the homography $\mathbf{H}_{\pi,i}$ and then to obtain ${}^g\mathbf{R}_i$, $\mathbf{t}_{d^*i} = {}^g\mathbf{t}_i/d^*$, $\mathbf{n}^*$ and thus $\boldsymbol{\Upsilon}_i$. As in the previous section, we construct a path starting at $\boldsymbol{\Upsilon}_i$ and oriented along the induced forces given by :

$$\left\{\begin{array}{lcl} \mathbf{F}_a(\boldsymbol{\Upsilon}) & = & -a\boldsymbol{\Upsilon} \\ \mathbf{F}_r(\boldsymbol{\Upsilon}) & = & -\mathbf{M}_\Upsilon^T(d^*)\mathbf{L}^T(\mathbf{s}, d^*)\left(\frac{\partial V_r(\mathbf{s})}{\partial\mathbf{s}}\right)^T \end{array}\right.$$

According to (4) and (8), $\mathbf{L}(\mathbf{s}, d^*)$ can be written :

$$\mathbf{L}(\mathbf{s}, d^*) = \left[\begin{array}{cc} \frac{1}{d^*}\mathbf{S} & \mathbf{Q} \end{array}\right] \tag{9}$$

where $\mathbf{Q} = [\mathbf{Q}_1^T .. \mathbf{Q}_j^T .. \mathbf{Q}_n^T]^T$ and $\mathbf{S} = [\mathbf{S}_1^T .. \mathbf{S}_j^T .. \mathbf{S}_n^T]^T$ are two $2n \times 3$ matrix independent of d^* :

$$\left\{\begin{array}{l} \mathbf{S}_j = \left[\begin{array}{ccc} -\frac{1}{\rho_j} & 0 & \frac{x_j}{\rho_j} \\ 0 & -\frac{1}{\rho_j} & \frac{y_j}{\rho_j} \end{array}\right] \\ \\ \mathbf{Q}_j = \left[\begin{array}{ccc} x_j y_j & -1 - x_j^2 & y_j \\ 1 + y_j^2 & -x_j y_j & -x_j \end{array}\right] \end{array}\right.$$

The Jacobian matrix $\mathbf{M}_\Upsilon(d^*)$ is given by :

$$\mathbf{M}_\Upsilon(d^*) = \left[\begin{array}{cc} d^*\,{}^g\mathbf{R}_c^T & \mathbf{0_{3\times3}} \\ \mathbf{0_{3\times3}} & L_w^{-1} \end{array}\right] \tag{10}$$

Using the above equation, the vector $\boldsymbol{\Upsilon}_k$ can be computed at each iteration and from $\boldsymbol{\Upsilon}_k$, the rotation matrix ${}^g\mathbf{R}_k$ and the vector $\mathbf{t}_{d^*,k} = {}^g\mathbf{t}_k/d^*$ are obtained.

2D trajectories The homography matrix $\mathbf{H}_{\pi,k}$ of plane Π relating the current and desired images can be computed from $\boldsymbol{\Upsilon}_k$ using (7) :

$$\mathbf{H}_{\pi,k} = {}^g\mathbf{R}_k^T - {}^g\mathbf{R}_k^T\mathbf{t}_{d^*,k}\mathbf{n}^{*T}$$

According to (5) the image coordinates of the points $\mathcal{P}_j$ belonging to Π at time k are given by :

$$\mu_j\mathbf{p}_{j,k} = \left[\begin{array}{ccc} \mu_j u_{j,k} & \mu_j v_{j,k} & \mu_j \end{array}\right]^T = \mathbf{G}_{\pi,k}\mathbf{p}_j^* \tag{11}$$

$\mathbf{p}_{j,k}$ is easily obtained by dividing $\mu_j\mathbf{p}_{j,k}$ by its last component, thus the equation (11) allows us to obtain the trajectories in the image.

Influence of d^*. The parameter d^* appears only in repulsive force through the matrix $\mathbf{J}$ composed of the product of $\mathbf{M}_\Upsilon^T(d^*)$ and $\mathbf{L}^T(\mathbf{s}, d^*)$. According to (9) and (10) we have :

$$\mathbf{J} = \mathbf{M}_\Upsilon^T(d^*)\mathbf{L}^T(\mathbf{s}, d^*) = \left[\begin{array}{c} {}^g\mathbf{R}_c\mathbf{S}^T \\ \mathbf{L}_w^{-T}\mathbf{Q}^T \end{array}\right]$$

That proves that $\mathbf{J}$ and thus the trajectories in the image are independent of parameter d^*.

Influence of intrinsic parameters. If the camera is not perfectly calibred and $\widehat{\mathbf{A}}$ is used instead of $\mathbf{A}$, the estimated homography matrix is :

$$\widehat{\mathbf{H}}_{\pi,i} = \widehat{\mathbf{A}}^{-1}\mathbf{A}\mathbf{H}_{\pi,i}\mathbf{A}^{-1}\widehat{\mathbf{A}} \tag{12}$$

Let us assume the following hypothesis (H1):

$$\widehat{\mathbf{H}}_{\pi,i} = \widehat{\mathbf{A}}^{-1}\mathbf{A}\mathbf{H}_{\pi,i}\mathbf{A}^{-1}\widehat{\mathbf{A}} \Longrightarrow \widehat{\mathbf{H}}_{\pi,k} = \widehat{\mathbf{A}}^{-1}\mathbf{A}\mathbf{H}_{\pi,k}\mathbf{A}^{-1}\widehat{\mathbf{A}}$$

This assumption means that the initial error in the estimated homography is propagated along the trajectory. According to (11) and (6) we obtain :

$$\widehat{\mu}_j\widehat{\mathbf{p}}_{j,k} = \widehat{\mathbf{A}}\widehat{\mathbf{H}}_{\pi,k}\widehat{\mathbf{A}}^{-1}\mathbf{p}_j^* \tag{13}$$

Considering (H1), (12) and (13), we obtain :

$$\widehat{\mu}_j\widehat{\mathbf{p}}_{j,k} = \mathbf{A}\mathbf{H}_{\pi,k}\mathbf{A}^{-1}\mathbf{p}_j^* = \mu_j\mathbf{p}_{j,k}$$

Therefore, under assumption H1, the trajectories in the image are not disturbed by errors on intrinsic parameters. We will check this nice property on the experimental results given in Section 5.

4 Control Scheme

In order to track the trajectories using an Image-based Control scheme, a vision-based task function $\mathbf{e}(\mathbf{r}(t), t)$ [4] is defined as :

$$\mathbf{e} = \widehat{\mathbf{L}}^+(\mathbf{s}(\mathbf{r}(t)) - \mathbf{s}^*(t)) \tag{14}$$

where $\mathbf{s}$ is composed of the current image coordinates, $\mathbf{s}^*$ is the desired trajectory of $\mathbf{s}$ computed in Sections 2, 3 and $\widehat{\mathbf{L}}^+$ is the pseudo-inverse of a chosen model of $\mathbf{L}$. The value of $\mathbf{L}$ at the current desired position is used for $\widehat{\mathbf{L}}$:

• if the target is known $\widehat{\mathbf{L}} = \mathbf{L}(\mathbf{s}^*_{\mathbf{k}}, \mathbf{Z}^*_k)$ where $\mathbf{Z}^*_k$ is easily obtained from $\boldsymbol{\Upsilon}_k$ and the target model

• else $\widehat{\mathbf{L}} = \mathbf{L}(\mathbf{s}^*_{\mathbf{k}}, \widehat{d}^*)$, $\widehat{d}^*$ being an estimated value of d^*

In order that $\mathbf{e}$ exponentially decreases toward 0 the velocity control law is given by [4] :

$$\mathbf{T} = -\lambda \mathbf{e} - \widehat{\frac{\partial \mathbf{e}}{\partial t}} \tag{15}$$

where λ is a proportional gain and $\widehat{\frac{\partial \mathbf{e}}{\partial t}}$ denotes an estimated value of the time variation of $\mathbf{e}$. If the target is motionless, we obtain from (14) :

$$\frac{\partial \mathbf{e}}{\partial t} = -\widehat{\mathbf{L}}^+ \frac{\partial \mathbf{s}^*}{\partial t} \tag{16}$$

According to (16), we rewrite (15) :

$$\mathbf{T} = -\lambda \mathbf{e} + \widehat{\mathbf{L}}^+ \widehat{\frac{\partial \mathbf{s}^*}{\partial t}}$$

where the term $\widehat{\mathbf{L}}^+ \widehat{\frac{\partial \mathbf{s}^*}{\partial t}}$ allows to compensate tracking error in following the specified trajectory [1]. It can be estimated as follow :

$$\widehat{\mathbf{L}}^+ \widehat{\frac{\partial \mathbf{s}^*}{\partial t}} = \widehat{\mathbf{L}}^+ \frac{\mathbf{s}^*_k - \mathbf{s}^*_{k-1}}{\Delta t}$$

The discretized control law at time $k\Delta t$ can finally be written :

$$\mathbf{T} = -\lambda \widehat{\mathbf{L}}^+ (\mathbf{s}_k - \mathbf{s}^*_k) + \widehat{\mathbf{L}}^+ \frac{\mathbf{s}^*_k - \mathbf{s}^*_{k-1}}{\Delta t}$$

5 Experiments

The methods presented have been tested on a six d-o-f eye-in-hand system. The target is a planar object with four white marks (see Figure 4). Displacement between the initial and final camera positions is very significant ($t_x = 300mm$, $t_y = 550mm$, $t_z = 120mm$, $(\mathrm{u}\theta)_x = 28dg$, $(\mathrm{u}\theta)_y = 78dg$, $(\mathrm{u}\theta)_z = 147dg$) and in this case classical image-based and position-based visual servoing fail. Figure 5(c) shows the importance of repulsive potential without which the visual features get out largely of the camera field of view.

The obtained results (see Figure 5) using the method presented on Section 2 and correct intrinsic parameters are very satisfactory. The positioning task is accurately realized with regular velocities (because the error $\mathbf{s}_k - \mathbf{s}^*_k$ keeps a regular value). After the complete realization of the trajectory, servoing is prolonged with a small gain and a constant reference. We can notice that the desired trajectories and the tracked trajectories are almost similar.

The method presented in Section 3 has been tested with two set of parameters. In Figure 6, intrinsic parameters given by camera manufacturer and real value of d^* has been used and in Figure 7, an error of 20% is added on intrinsic parameters as well as on the parameter d^*. In both cases the results are satisfactory. In particular and as expected, we will note that the planned trajectories are practically identical in both cases.

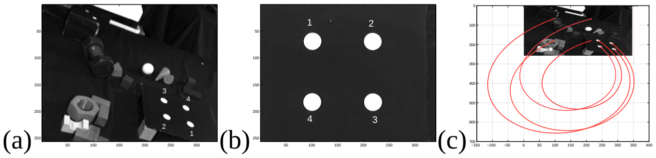

Figure 4: Initial -a-, desired -b- images of the target and trajectories planned without repulsive potential -c-

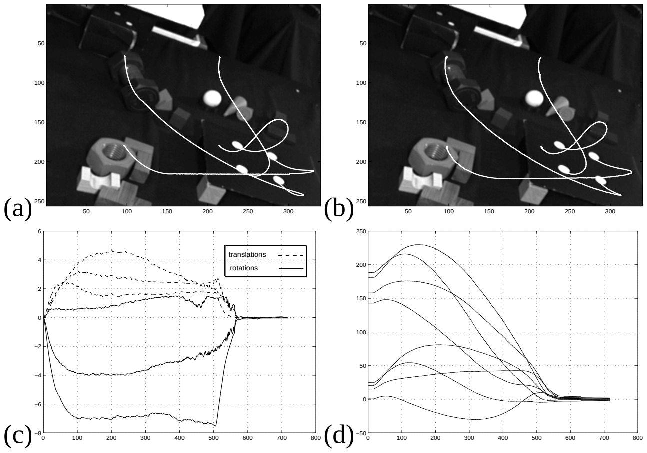

Figure 5: First case, a : planned trajectories, b : followed trajectories, c : velocities (cm/s and dg/s), d : error on pixels coordinates -d-

6 Conclusion

In this paper, we have presented a powerful method to increase the application area of visual servoing to the cases where initial and desired positions of the camera are distant. Experimental results show the validity of our approach and its robustness with respect to modeling errors. Future work will be devoted to introduce supplementary constraints in the planed trajectories : to avoid robot joint limits, kinematic singularities, occlusions and obstacles. Another perspective is to generate the trajectories in image space of more complex features that n points in order to apply our method to real objects.

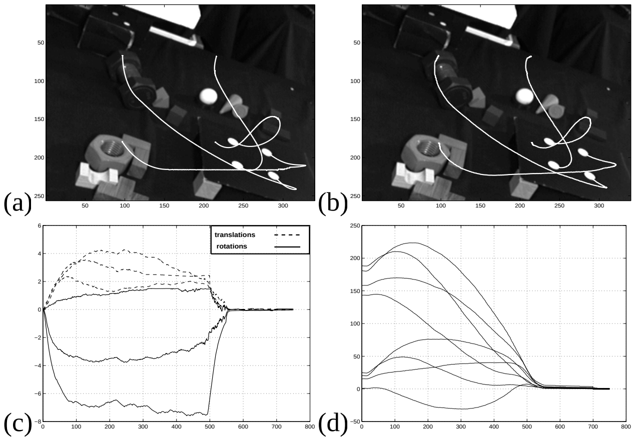

Figure 6: Second case without errors, a : planned trajectories, b : followed trajectories, c : velocities (cm/s and dg/s), d : error on pixels coordinates

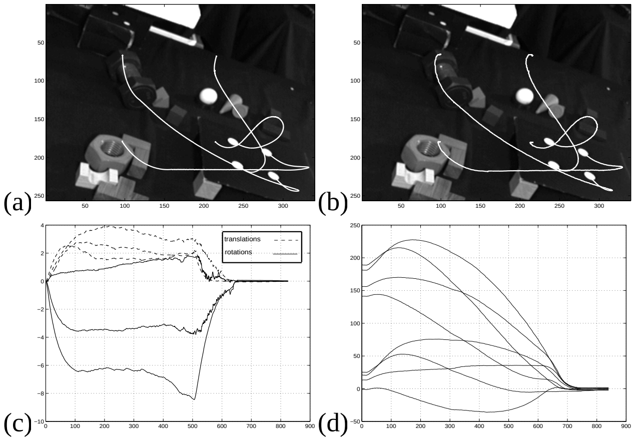

Figure 7: Second case with errors, a : planned trajectories, b : followed trajectories, c : velocities (cm/s and dg/s), d : error on pixels coordinates

References

[1] F. Berry, P. Martinet, and J. Gallice. Trajectory generation by visual servoing. *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2:1066–1072, 1997.

[2] F. Chaumette. Potential problems of stability and convergence in image-based and position-based visual servoing. *The Confluence of Vision and Control D. Kriegman, G. Hager, A. Morse (eds), LNCIS Series, Springer Verlag*, 237:66–78, 1998.

[3] D. Dementhon and L.S. Davis. Model-based object pose in 25 lines of code. *Int. Journal of Computer Vision, 15(1/2): 123-141*, June 1995.

[4] B. Espiau, F. Chaumette, and P. Rives. A new approach to visual servoing in robotics. *IEEE Trans. on Robotics and Automation, 8(3) : 313-326*, 1992.

[5] O. Faugeras and F. Lustman. Motion and structure from motion in a piecewise planar environment. *Int. Journal of Pattern Recognition and Artificial Intelligence*, 2(3):485–508, 1988.

[6] K. Hashimoto. *Visual Servoing: Real Time Control of Robot Manipulators Based on Visual Sensory Feedback*. World Scientific Series in Robotics and Automated Systems, Vol 7,World Scientific Press, Singapor, 1993.

[7] K. Hosoda, K. Sakamoto, and M. Asada. Trajectory generation for obstacle avoidance of uncalibrated stereo visual servoing without 3d reconstruction. *Proc. IEEE/RSJ Int. Conference on Intelligent Robots and Systems*, 1(3):29–34, August 1995.

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

[9] O. Khatib. Real time obstacle avoidance for manipulators and mobile robots. *Int. Journal of Robotics Research*, 5(1):90–98, 1986.

[10] J. C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, 1991.

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

[12] E. Marchand and G.D. Hager. Dynamic sensor planning in visual servoing. *IEEE Int. Conference on Robotics and Automation*, 3:1988–1993, May 1998.

[13] B.J. Nelson and P.K. Khosla. Integrating sensor placement and visual tracking strategies. *IEEE Int. Conference on Robotics and Automation*, 2:1351–1356, May 1994.

[14] R. Singh, R. M. Voyle, D. Littau, and N. P. Papanikolopoulos. Alignement of an eye-in-hand system to real objects using virtual images. *Workshop on Robust Vision for Vision-Based Control of Motion, IEEE Inter. Conf. on Robotics and Automation*, May 1998.

[15] L.E Weiss, A.C Sanderson, and C.P Neuman. Dynamic sensor-based control of robot with visual feedback. *IEEE Journal of Robotics and Automation*, 3(5):404–417, October 1987.