HAL open science

Design and tracking of desirable trajectories in the image space by integrating mechanical and visibility constraints

Youcef Mezouar, François Chaumette

To cite this version:

Youcef Mezouar, François Chaumette. Design and tracking of desirable trajectories in the image space by integrating mechanical and visibility constraints. IEEE Int. Conf. on Robotics and Automation, ICRA'01, 2001, Seoul, South Korea. pp.731-736. inria-00352136

HAL Id: inria-00352136

https://inria.hal.science/inria-00352136

Submitted on 12 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.

Proceedings of the 2001 IEEE
International Conference on Robotics & Automation
Seoul, Korea - May 21-26, 2001

Design and Tracking of Desirable Trajectories in the Image Space by Integrating Mechanical and Visibility Constraints

Youcef Mezouar
Youcef.Mezouar@irisa.fr

François Chaumette
Francois.Chaumette@irisa.fr

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

Abstract

Since image-based visual servoing is a local feedback control solution, it requires the definition of intermediate subgoals in the sensor space at the task planning level. In this paper, we describe a general technique for specifying and tracking trajectories of an unknown object in the camera image space. First, physically valid C^2 *image trajectories which correspond to quasi-optimal 3D camera trajectory (approaching as much as possible a straight line) are performed.* Both mechanical (joint limits) and visibility constraints *are taken into account at the task planning level. The good behavior of Image-based control when desired and current camera positions are closed is then exploited to design an efficient control scheme. Real time experimental results using a camera mounted on the end effector of a six d-o-f robot confirm the validity of our approach.*

1 Introduction

Classical approaches, using visual information in feedback control loops [6, 8], are point to point-based, i.e the robot must reach a desired goal configuration starting from a given initial configuration. In such approaches, a globally stabilizing feedback control solution is required. However if the initial error is large, such a control may product erratic behavior especially in presence of modeling errors. For a very simple case, Cowan and Koditschek describe in [3] a globally stabilizing method using navigation function. By composing the error function of 3D Cartesian features and image features, Malis et *al* propose a globally stabilizing solution called *2 1/2 D* visual servoing for general setup [11]. Classical image-based visual servoing is a local control solution. It thus requires the definition of intermediate subgoals in the sensor space at the task planning level if the initial error is large. In this approach the robot effector is controlled so that the image features converge to the reference image features. The robot effector trajectory in the 3D Cartesian space is not controlled. Such a control can thus provide inadequate robot motion leading to no optimal or no physically valid robot trajectory [1]. However, it is well known that image-based control is locally stable and robust with respect to modeling errors and noise perturbations. The key idea of our work is to use the local stability and robustness of image-based servoing by specifying trajectories to follow in the image. Indeed, for a trajectory following a local control solution works properly since current and desired configurations remain close. Only few papers deal with path planning in image space. In [7] a trajectory generator using a stereo system is proposed and applied to obstacle avoidance. An alignment task using intermediate views of the object synthesized by image morphing is presented in [16]. A path planning for a straight-line robot translation observed by a weakly calibrated stereo system is performed in [14]. In previous work [12], we have proposed a potential field-based path planning generator that determines the trajectories in the image of a set of points lying on a planar target. In this paper, we propose to plan the trajectory of an **unknown and not necessarily planar** object. Both **mechanical (joint limits) and visibility** constraints are taken into account. Contrarily to others approaches [2, 13] exploiting the robot redundancy, the mechanical and visibility constraints can be ensured even if all the robot degrees of freedom are used to realize the task.

More precisely, we plan the trajectory of $\mathbf{s} = [\mathbf{p}_1^T \cdots \mathbf{p}_n^T]^T$, composed of the $2 \times n$ image coordinates of n points $\mathcal{M}_j$ lying on an unknown target, between the initial configuration $\mathbf{s}_i = [\mathbf{p}_{1i}^T \cdots \mathbf{p}_{ni}^T]^T$ and the desired one $\mathbf{s}^* = [\mathbf{p}_1^{*T} \cdots \mathbf{p}_n^{*T}]^T$. Our approach consists of three phases. In the first one, the discrete geometric camera path (that ensures the physical validity of the image trajectories) is performed as a sequence of N intermediate camera poses which approaches as much as possible a straight line in the Cartesian space. In this phase, the mechanical and visi-

bility constraints are introduced. In the second one, the discrete geometric trajectory of the target in the image and the discrete geometric trajectory of the robot in the joint space are obtained from the camera path. Finally, continuous and derivable geometric paths in the image with an associated timing law $\mathbf{s}^*(t)$ is generated and tracked using an image-based control scheme.

The paper is organized as follows. In Section 2, we recall some fundamentals. The method of path planning is presented in Section 3. In Section 4, we show how to use an image-based control approach to track the trajectories. In section 5, a timing law is associated with the geometric path. The experimental results are given in Section 6.

2 Potential field method

Our path planning strategy is based on the potential field method [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 ($\boldsymbol{\Upsilon}_*$) 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 $\boldsymbol{\Upsilon}$. Each segment of the path is oriented along the negated gradient of the potential function:

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

where k is the increment index and ε_k a positive scaling factor denoting the length of the kth increment. In our case, the control objective is formulated to transfer the system to a desired point in the sensor space and to provide robot motion satisfying the following constraints:

1. all the considered image features remain in the camera field of view
2. the robot joint positions remain between their limits

To deal with the first constraint, a repulsive potential $V_{rs}(\mathbf{s})$ is defined in the image. The second constraint is introduced through a repulsive potential $V_{rq}(\mathbf{q})$ defined in the joint space. The total force is given by:

$$\mathbf{F} = -\vec{\nabla} V_{ak} - \gamma_k \vec{\nabla} V_{rsk} - \chi_k \vec{\nabla} V_{rqk}$$

The scaling factors γ_k and χ_k allow us to adjust the relative influence of the different forces.

3 Trajectories planning

We consider that the target model is not available. In this case the camera pose can not be estimated. Only a scaled Euclidean reconstruction can be obtained by performing a partial pose estimation as described in the next subsection. This partial pose estimation and the relations linking two views of a static object are then exploited to design a path of the projection of the unknown object in the image.

3.1 Scaled Euclidean reconstruction

Let $\mathcal{F}^*$ and $\mathcal{F}$ be the frames attached to the camera in its desired and current positions. The rotation matrix and the translation vector between $\mathcal{F}$ and $\mathcal{F}^*$ are denoted ${}^*\mathbf{R}_c$ and ${}^*\mathbf{t}_c$, respectively. A target point $\mathcal{M}_j$ with homogeneous coordinates $\mathbf{M}_j = [X_j\, Y_j\, Z_j\, 1]$ (resp. $\mathbf{M}_j^*$) in $\mathcal{F}$ (resp. $\mathcal{F}^*$) is projected in the camera image onto a point with homogeneous normalized and pixel coordinates $\mathbf{m}_j = [x_j\, y_j 1]^T$ (resp. $\mathbf{m}_j^*$) and $\mathbf{p}_j = [u_j\, v_j\, 1]^T = \mathbf{A} m_j$ (resp. $\mathbf{p}_j^*$), where $\mathbf{A}$ denotes the intrinsic parameters matrix of the camera.

Consider a 3D reference plane Π given in $\mathcal{F}^*$ by $\pi^T = [\mathbf{n}^* - d^*]$, where $\mathbf{n}^*$ is its unitary normal in $\mathcal{F}^*$ and d^* is the distance from Π to the origin of $\mathcal{F}^*$ (see Fig. 1). It is well known that there is a projective homography matrix $\mathbf{G}_\pi$ such that:

$$\alpha_j \mathbf{p}_j = \mathbf{G}_\pi \mathbf{p}_j^* + \beta_j \mathbf{e} \text{ with } \mathbf{e} = -\mathbf{A}\, {}^*\mathbf{R}_c^T\, {}^*\mathbf{t}_c \tag{2}$$

where α_j is a positive scaling factor and β_j is a scaling factor null if the target point is linked with Π. More precisely, if we define the signed distance $d(\mathcal{M}_j, \Pi) = \pi\, \mathbf{M}_j^*$, we have:

$$\beta_j = -\frac{d(\mathcal{M}_j, \Pi)}{Z_j^* d^*} \tag{3}$$

If at least four matched points belonging to Π are known, $\mathbf{G}_\pi$ can be estimated by solving a linear system. Else, at least height points (3 points to define Π and 5 outside of Π) are necessary to estimate the homography matrix by using for example the linearized algorithm proposed in [11]. Assuming that the camera calibration is known, the Euclidean homography $\mathbf{H}_\pi$ of plane Π is estimated as follow:

$$\mathbf{H}_\pi = \mathbf{A}^{-1} \mathbf{G}_\pi \mathbf{A} \tag{4}$$

and it can be decomposed into a rotation matrix and a rank 1 matrix [5]:

$$\mathbf{H}_\pi = {}^*\mathbf{R}_c^T - {}^*\mathbf{R}_c^T \mathbf{t}_{d^*} \mathbf{n}^{*T} \quad \text{where} \quad \mathbf{t}_{d^*} = \frac{{}^*\mathbf{t}_c}{d^*} \tag{5}$$

From $\mathbf{H}_\pi$ and the image features, it is thus possible to determine the camera motion parameters (that is the rotation ${}^*\mathbf{R}_c$ and the scaled translation $\mathbf{t}_{d^*}$) and the vector n^* [5]. The structure of the observed scene can also be determined. For example, the ratio between the Z-coordinates of a 3D point $\mathcal{M}_j$ expressed in $\mathcal{F}$ and the distance d^*, $\rho_j = Z_j / d^*$ can be obtained from ${}^*\mathbf{R}_c$, $\mathbf{t}_{d^*}$ and the image features [11].These parameters are important since they are used in the path planning generator and in the control scheme.

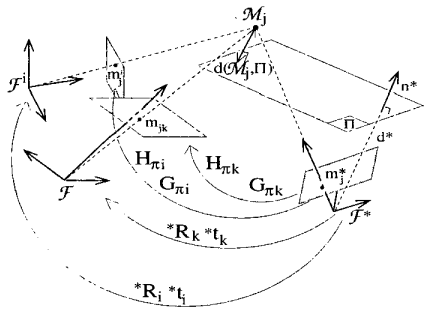

Figure 1: Scaled camera trajectory

3.2 Scaled 3-D Cartesian trajectory

The homography $\mathbf{G}_{\pi i}$ is computed from $\mathbf{s}^i$ and $\mathbf{s}^*$. According to (4), we obtain $\mathbf{H}_{\pi i}$. Then $\mathbf{n}^*$ is estimated, as well as the rotation $^*\mathbf{R}_i$ and the scaled translation $\mathbf{t}_{d^*i} = {}^*\mathbf{t}_i/d^*$ between $\mathcal{F}^i$ (frame linked to the camera in its initial position) and $\mathcal{F}^*$. If we choose as partial parameterization of the workspace $\boldsymbol{\Upsilon} = [\mathbf{t}_{d^*}^T \ (\mathbf{u}\theta)^T]^T$ where $\mathbf{u}$ and θ are the normalized rotation axis and the rotation angle extracted from $^*\mathbf{R}_c$, we obtain at the initial and desired robot configurations $\boldsymbol{\Upsilon}_i = [\mathbf{t}_{d^*i}^T \ (\mathbf{u}\theta)_i^T]^T$ and $\boldsymbol{\Upsilon}_* = \mathbf{0}_{6\times1}$. We thus have to determine a path starting at the initial configuration $\boldsymbol{\Upsilon}_{k=0} = \boldsymbol{\Upsilon}_i$ and ending at $\boldsymbol{\Upsilon}_* = \mathbf{0}_{6\times1}$.

3.3 Trajectories in the joint space

To anticipate the possible encounter of a joint limit and to avoid it, we have to estimate the trajectory of the robot in the joint space. Indeed, the value of the joint coordinates at each iteration is used in the computation of the repulsive potential related to the joint limits avoidance. If the manipulator position in the joint space is represented by $\mathbf{q} = [q_1 \cdots q_m]^T$, we have:

$$\frac{\partial \mathbf{q}}{\partial \boldsymbol{\Upsilon}} = \frac{\partial \mathbf{q}}{\partial \mathbf{r}}\frac{\partial \mathbf{r}}{\partial \boldsymbol{\Upsilon}} = \mathbf{J}^+(\mathbf{q})\mathbf{M}_{\boldsymbol{\Upsilon}}(d^*) \tag{6}$$

where $\mathbf{J}(\mathbf{q})$ and $\mathbf{M}_{\boldsymbol{\Upsilon}}(d^*)$ denote the robot Jacobian and the parameterization Jacobian respectively. The parameterization Jacobian can be computed directly from $\boldsymbol{\Upsilon}$ and d^* [12]:

$$\mathbf{M}_{\boldsymbol{\Upsilon}}(d^*) = \begin{bmatrix} d^*\,{}^*\mathbf{R}_c^T & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{L}_w^{-1} \end{bmatrix}$$

The computation of $\mathbf{L}_w^{-1}$ can be found in [11][1]:

$$\mathbf{L}_w^{-1} = \mathbf{Id}_{3\times3} + \frac{\theta}{2}\mathrm{sinc}^2\left(\frac{\theta}{2}\right)[\mathbf{u}]_\wedge + (1-\mathrm{sinc}(\theta))[\mathbf{u}]_\wedge^2$$

The trajectory of the robot coordinates in the joint space are then obtained from the trajectory of $\boldsymbol{\Upsilon}$ by a linearization of (6) around $\mathbf{q}_k$:

$$\mathbf{q}_{k+1} = \mathbf{q}_k + \mathbf{J}^+(\mathbf{q}_k)\mathbf{M}_{\boldsymbol{\Upsilon}_k}(d^*)(\boldsymbol{\Upsilon}_{k+1} - \boldsymbol{\Upsilon}_k)$$

3.4 Image trajectories

The homography matrix $\mathbf{G}_{\pi k}$ of plane Π relating the current and desired images can be computed from $\boldsymbol{\Upsilon}_k$ using (4) and (5):

$$\mathbf{G}_{\pi k} = \mathbf{A}({}^*\mathbf{R}_k^T - {}^*\mathbf{R}_k^T\mathbf{t}_{d^*k}\mathbf{n}^{*T})\mathbf{A}^{-1} \tag{7}$$

According to (2) the image coordinates of the points $\mathcal{M}_j$ at time k are given by:

$$\mu_{jk}\mathbf{p}_{jk} = \mathbf{G}_{\pi k}\mathbf{p}_j^* + \beta_j\mathbf{e}_k \tag{8}$$

where (refer to (2) and (3)):

$$\beta_j\mathbf{e}_k = \frac{d(\mathcal{M}_j,\Pi)}{Z_j^*}\mathbf{A}^*\mathbf{R}_k^T\mathbf{t}_{d^*k}$$

using the previous relation, (8) can be rewritten:

$$\mu_{jk}\mathbf{p}_{jk} = \mathbf{G}_{\pi k}\mathbf{p}_j^* + \frac{d(\mathcal{M}_j,\Pi)}{Z_j^*}\mathbf{A}^*\mathbf{R}_k^T\mathbf{t}_{d^*k} \tag{9}$$

Furthermore, if the relation (9) is applied between the desired and initial camera positions, we obtain [2]:

$$\frac{d(\mathcal{M}_j,\Pi)}{Z_j^*} = \mathrm{sign}\left(\frac{(\mu_j\mathbf{p}_{ji} - \mathbf{G}_{\pi i}\mathbf{p}_j^*)_1}{(\mathbf{A}\,{}^*\mathbf{R}_i^T\mathbf{t}_{d^*i})_1}\right)\frac{\|\mathbf{G}_{\pi i}\mathbf{p}_j^* \wedge \mathbf{p}_{ji}\|}{\|\mathbf{A}^*\mathbf{R}_i^T\mathbf{t}_{d^*i} \wedge \mathbf{p}_{ji}\|} \tag{10}$$

The equations (7), (9) and (10) allow to compute $\mu_{jk}\mathbf{p}_{jk}$ from $\boldsymbol{\Upsilon}_k$ and the initial and desired visual features. The image coordinates $\mathbf{p}_{jk}$ are then computed by dividing $\mu_j\mathbf{p}_{jk}$ by its last component. Furthermore, the ratio ρ_{jk}, which will be used in the repulsive force and in the control law, can easily be obtained from $\boldsymbol{\Upsilon}_k$ and $\mathbf{m}_{jk} = \mathbf{A}^{-1}\mathbf{p}_{jk}$.

3.5 Reaching the goal

The potential field V_a is defined as a parabolic function in order to minimize the distance between the current position and the desired one: $V_a(\boldsymbol{\Upsilon}) = \frac{1}{2}\|\boldsymbol{\Upsilon} - \boldsymbol{\Upsilon}_*\|^2 = \frac{1}{2}\|\boldsymbol{\Upsilon}\|^2$. The function V_a is positive or null and attains its minimum at $\boldsymbol{\Upsilon}_*$ where $V_a(\boldsymbol{\Upsilon}_*) = 0$. It generates a force $\mathbf{F}_a$ that converges linearly toward the goal configuration:

$$\mathbf{F}_a(\boldsymbol{\Upsilon}) = -\vec{\nabla}V_a = -\boldsymbol{\Upsilon} \tag{11}$$

When the repulsive potentials are not needed, the transition equation can be written (refer to (1) and (11)):

$$\boldsymbol{\Upsilon}_{k+1} = \left(1 - \frac{\varepsilon_k}{\|\boldsymbol{\Upsilon}_k\|}\right)\boldsymbol{\Upsilon}_k$$

[1] $[\mathbf{u}]_\wedge$ denotes the antisymmetric matrix associated to the vector $\mathbf{u}$

[2] $(\mathbf{v})_j$ is the jth components of $\mathbf{v}$

Thus, Υ_k is lying on the straight line passing by Υ_i and Υ_*. As a consequence, the translation performed by the camera is a real straight line since Υ_k is defined with respect to a motionless frame (that is $\mathcal{F}_*$). However, the object can get out of the camera field of view and the robot can attain its joint limits along this trajectory. In order that the object remains in the camera field of view and that the robot remains in its mechanical limits, two repulsive forces are introduced by deviating the camera trajectory when needed.

3.6 Mechanic and visibility constraints

A. Joint limits avoidance. The robot configuration $\mathbf{q}$ is called acceptable if each of its components is sufficiently far away from its corresponding joints limits. We note $\mathcal{L}$ the subset of the joint space of acceptable configurations. The repulsive potential $V_{rq}(\mathbf{q})$ is defined as:

$$\begin{cases} V_{rq}(\mathbf{q}) &= \sum_{j=1}^{m} \frac{1}{(1-\frac{q_j}{q_{jmax}})(1-\frac{q_j}{q_{jmin}})} \quad \text{if } \mathbf{q} \notin \mathcal{L} \\ &= 0 \quad \text{if } \mathbf{q} \in \mathcal{L} \end{cases}$$

q_{jmin} and q_{jmax} being the minimum and the maximum allowable joint values for the jth joint. The artificial repulsive force deriving from V_{rq} is :

$$\mathbf{F}_{rq}(\Upsilon) = -\left(\frac{\partial V_{rq}(\mathbf{q})}{\partial \Upsilon}\right)^T = -\left(\frac{\partial V_{rq}(\mathbf{q})}{\partial \mathbf{q}}\frac{\partial \mathbf{q}}{\partial \mathbf{r}}\frac{\partial \mathbf{r}}{\partial \Upsilon}\right)^T$$

The previous equation can be rewritten:

$$\mathbf{F}_{rq}(\Upsilon) = -\mathbf{M}_{\Upsilon}^T(d^*)\mathbf{J}^{+T}(\mathbf{q})\left(\frac{\partial V_r(\mathbf{q})}{\partial \mathbf{q}}\right)^T$$

B. Visibility constraint. One way to create a potential barrier around the camera field of view is to define V_{rs} as an increasing function of the distance between the object projection and the image limits. A general description of such a function is given in [12].

4 Performing C^2 timing law

In the previous subsection we have obtained discrete trajectories. In order to design continuous and derivable curves and thus to improve the dynamic behavior of the system, we use cubic B-spline interpolation. The spline interpolation problem is usually stated as: given data points $\mathcal{S} = \{\mathbf{s}_k / k \in 1 \cdots N\}$ and a set of parameter values $\mathcal{T} = \{t_k / k \in 1 \cdots N\}$, we have to determine a cubic B-spline curve $\mathbf{s}(t)$ such that $\mathbf{s}(t_k) = \mathbf{s}_k, \forall t_k$. In practice, parameter values are rarely given. In our case, we can adjust them to the distribution of the vector of image features $\mathbf{s}_k$ or using the distribution of the camera positions Υ_k. In order to control efficiently the camera velocity, it is more reasonable to use the distribution of the camera positions. The time values are thus chosen spacing proportionally to the distances between camera positions (see Fig. 2):

$$\frac{\Delta t_k}{\Delta t_{k+1}} = \frac{t_{k+1} - t_k}{t_{k+2} - t_{k+1}} = \frac{\|\Upsilon_{k+1} - \Upsilon_k\|}{\|\Upsilon_{k+2} - \Upsilon_{k+1}\|}$$

Considering the transition equation (1), we obtain:

$$\Delta t_{k+1} = \frac{\varepsilon_k}{\varepsilon_{k+1}}\Delta t_k \quad \text{with} \quad \Delta t_0 = T$$

T being the time between two consecutive frames (chosen for example as the video rate). In practice, ε_k is chosen constant, we thus have $t_k = kT$. Given the data vectors $\mathbf{s}_k$ and the parameters values t_k, the image data can be interpolated by using a natural cubic B-spline interpolation and we obtain a C^2 function $\mathbf{s}(t)$ defined for $(k-1)\Delta T \leq t \leq k\Delta T$ by:

$$\mathbf{s}(t) = \mathbf{A}_k t^3 + \mathbf{B}_k t^2 + \mathbf{C}_k t + \mathbf{D}_k \tag{12}$$

where the $n \times n$ diagonal matrices $\mathbf{A}_k$, $\mathbf{B}_k$, $\mathbf{C}_k$, $\mathbf{D}_k$ are obtained from $\mathcal{S}$ and $\mathcal{T}$.
Finally, the ratio ρ appears in the control law. By using the same process, $\Gamma(t) = [\rho_1(t) \cdots \rho_n(t)]$ is computed from $\mathcal{R} = \{\Gamma_k / k \in 1 \cdots N\}$ and $\mathcal{T}$.

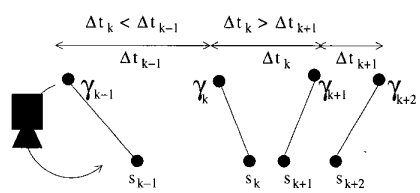

Figure 2: Controlling the time along the camera trajectory

5 Tracking the trajectories

To track the trajectories using an image-based control scheme, we use the task function approach introduced by Samson et *al* in [15]. A vision-based task function $\mathbf{e}$ to be regulated to $\mathbf{0}$ is defined by [4]:

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

The time varying vector $\mathbf{s}^*(t)$ is the desired trajectory of $\mathbf{s}$ computed in the previous sections. $\mathbf{L}$ denotes the well known interaction matrix (also called image Jacobian) and the matrix $\widehat{\mathbf{L}}^+$ represents the pseudo-inverse of a chosen model of $\mathbf{L}$. More precisely, when $\mathbf{s}$ is composed of the image coordinates of n points, two successive rows of the image Jacobian are given by:

$$\begin{bmatrix} -\frac{1}{d^*\rho_j} & 0 & \frac{x_j}{d^*\rho_j} & x_j y_j & -(1+x_j^2) & y_j \\ 0 & -\frac{1}{d^*\rho_j} & \frac{y_j}{d^*\rho_j} & (1+y_j^2) & -x_j y_j & -x_j \end{bmatrix}$$

The value of $\mathbf{L}$ at the current desired position is used for $\widehat{\mathbf{L}}$, that is $\widehat{\mathbf{L}} = \mathbf{L}(\mathbf{s}^*(t), \mathbf{\Gamma}^*(t), \widehat{d}^*)$, $\widehat{d}^*$ being an estimated value of d^* and $\mathbf{\Gamma}^*(t) = [\rho_1^*(t) \cdots \rho_n^*(t)]$ is the obtained trajectory for $\mathbf{\Gamma}$. An exponential decay of $\mathbf{e}$ toward $\mathbf{0}$ can be obtained by imposing $\dot{\mathbf{e}} = -\lambda \mathbf{e}$ (λ being a proportional gain), the corresponding control law is:

$$\mathbf{T}_c = -\lambda \mathbf{e} - \frac{\partial \mathbf{e}}{\partial t}$$

where $\mathbf{T}_c$ is the camera velocity sent to the robot controller. If the target is known to be motionless, we have $\frac{\partial \mathbf{e}}{\partial t} = -\widehat{\mathbf{L}}^+ \frac{\partial \mathbf{s}^*}{\partial t}$ and the camera velocity can be rewritten:

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

where the term $\widehat{\mathbf{L}}^+ \frac{\partial \mathbf{s}^*}{\partial t}$ allows to compensate the tracking error. More precisely, we have from (12):

$$\frac{\partial \mathbf{s}^*}{\partial t} = 3\mathbf{A}_k t^2 + 2\mathbf{B}_k t + \mathbf{C}_k \quad \text{for} \quad (k-1)\Delta T \leq t \leq k\Delta T$$

This control law posses nice degrees of robustness with respect to modeling errors and noise perturbations since the error function used as input remains small and is directly computed from visual features.

6 Experimental Results

The proposed method has been tested on a six d-o-f eye-in-hand system. The specified visual task consists in a positioning task with respect to an unknown object. The target is a marked object with nine white marks lying on three differerent planes (see Fig. 3). The extracted visual features are the image coordinates of the center of gravity of each mark. The images corresponding to the desired and initial camera positions are given in Figs. 3(a) and 3(b) respectively. The corresponding camera displacement is very important ($t_x = 820mm$, $t_y = 800mm$, $t_z = 450mm$, $(\mathrm{u}\theta)_x = 37dg$, $(\mathrm{u}\theta)_y = 45dg$, $(\mathrm{u}\theta)_z = 125dg$). In this case classical image-based and position-based visual servoing fail.

On all the following plots, joint positions are normalized between [-1;1], where -1 and 1 represents the joint limits.

In order to emphasize the importance of the introduced constraints in the trajectories, we first perform the path planning without repulsive potential. The results are given in Fig. 3. We can see that the visual features get out largely of the camera field of view (Fig.3(c)) and the axis q_5 attains its joint limit (Fig.3(d)) . Then, only the repulsive potential associated to the visibility constraint has been activated (see Fig. 4). In that case, even if the visibility constraint is ensured (Fig 4(a)) the servoing can not be realized because the axis q_5 reaches its joint limit (Fig 4(b)). In Fig. 5, the two repulsive potentials are activated. The target remains in the camera field of view (see Fig. 5(a) and 5(c)) and all axes avoid their joint limit (see Fig. 5(b) and 5(d)). We can notice that the planned trajectories and the realized trajectories in the image are almost similar, that shows the efficiency of our control scheme. The error on the image coordinates of each target point between its current and desired location is given in Fig. 5(f). We can note the convergence of the coordinates to their desired value, which demonstrates the correct realization of the positioning task.

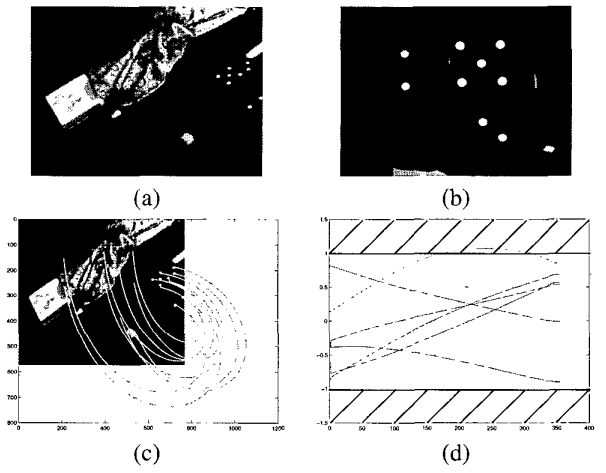

Figure 3: (a) Initial and (b) Desired images; Planned trajectories without any repulsive potential (c) in the image, (d) in the joint space

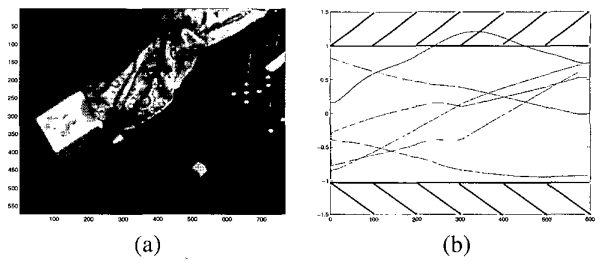

Figure 4: Planned trajectories without repulsive potential associated to the joint limits avoidance (a) in the image, (b) in the joint space

7 Conclusion

In this paper, we have presented a method ensuring the convergence for all initial camera position. By coupling an image-based trajectory generator and an image-based servoing, the proposed method extends the well known stability of image-based servoing when initial and desired camera location are close to the case where they are distant. The obtained trajectories provide some good expected

properties. First, along these trajectories the target remains in the camera field of view and the robot remains in its mechanical limits. Second the corresponding robot motion is physically realizable and the camera trajectory is a straight line outside the area where the repulsive forces are needed. Experimental results show the validity of our approach. Future work will be devoted to generate the trajectories in image space of complex features in order to apply our method to complex objects.

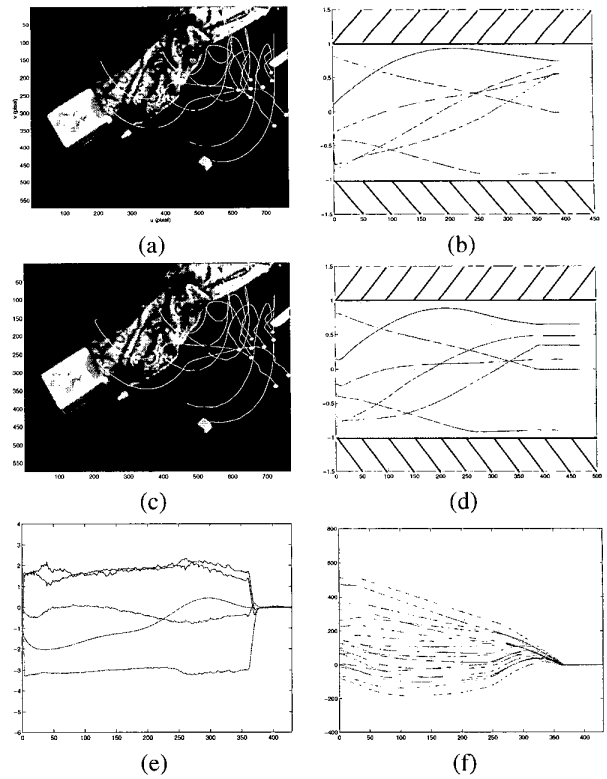

Figure 5: Planned trajectories with both repulsive potential (a) in the image, (b) in the joint space; realized trajectories (c) in the image, (d) in the joint space, (e) camera translational (cm/s) and rotational (dg/s) velocities versus iteration number, (f) errors in the image versus iteration number

References

[1] 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.

[2] F. Chaumette and E. Marchand. A new redundancy-based iterative scheme for avoiding joint limits: Application to visual sevoing. In *Proc. IEEE Int. Conf. on Robotics and Automation*, volume 2, pages 1720–1725, San Francisco, California, April 2000.

[3] N.J. Cowan and D.E. Koditschek. Planar image based visual servoing as a navigation problem. *IEEE International Conference on Robotics and Automation*, pages 611–617, May 1999.

[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 and F. Chaumette. 2 1/2 d visual servoing with respect to unknown objects through a new estimation scheme of camera displacement. *International Journal of Computer Vision*, June 2000.

[12] Y. Mezouar and F. Chaumette. Path planning in image space for robust visual servoing. *IEEE Int. Conference on Robotics and Automation*, 3:2759–2764, April 2000.

[13] B.J. Nelson and P.K. Khosla. Strategies for increasing the tracking region of an eye-in-hand system by singularity and joint limits avoidance. *Int. Journal of Robotics Research, 14(3):255-269*, June 1995.

[14] A. Ruf and R. Horaud. Visual trajectories from uncalibrated stereo. *IEEE International Conference on Intelligent Robots and Systems*, pages 83–91, 1997.

[15] C. Samson, B. Espiau, and M. Le Borgne. *Robot Control : The Task Function Approach.* Oxford University Press, 1991.

[16] 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 Int. Conf. on Robotics and Automation*, May 1998.