An Unified Approach to Model-Based and Model-Free Visual Servoing

Ezio Malis

I.N.R.I.A. - 2004, route des Lucioles - B.P. 93, 06902 Sophia Antipolis Cedex, France.
http://www-sop.inria.fr/icare/personnel/malis/index.html
Ezio.Malis@sophia.inria.fr

Abstract. Standard vision-based control techniques can be classified into two groups: model-based and model-free visual servoing. Model-based visual servoing is used when a 3D model of the observed object is available. If the 3D model is completely unknown, robot positioning can still be achieved using a teaching-by-showing approach. This model-free technique needs a preliminary learning step during which a reference image of the scene is stored. The objective of this paper is to propose an unified approach to vision-based control which can be used with a zooming camera whether the model of the object is known or not. The key idea of the unified approach is to build a reference in a projective space invariant to camera intrinsic parameters which can be computed if the model is known or if an image of the object is available. Thus, only one low level visual servoing technique must be implemented at once.

1 Introduction

Standard vision-based control techniques [7] can be classified into two groups: model-based and model-free visual servoing. Model-based visual servoing is used when a 3D model of the observed scene is available. The explicit exploitation of the CAD model of an object facilitates recognition [13] and tracking [6]. Using both the model and measured image features, one can estimate the pose of the camera with respect to the object frame. Thus, a robot, with a camera mounted on the end-effector, can be driven to any desired position using a standard position-based control law [15]. Obviously, if the 3D structure of the environment is completely unknown, model-based visual servoing can not be used. In that case, robot positioning can still be achieved using a teaching-by-showing approach. This model-free technique, completely different from the previous one, needs a preliminary learning step during which a reference image of the scene is stored. After the camera and/or the object have been moved, several visual servoing methods [1,4,10] have been proposed in order to drive the robot back to the reference position. When the current image observed by the camera is identical to the reference image the robot is back to the desired position. The model-free approach has the advantage of avoiding the knowledge of the model but it cannot be used with a zooming camera. If the camera intrinsic parameters change during the servoing, then the reference image must be learned again.

A. Heyden et al. (Eds.): ECCV 2002, LNCS 2353, pp. 433–447, 2002.

Both model-based and model-free approaches are useful but, depending on the "a priori" knowledge we have of the scene, we must switch between them. The objective of this paper is to propose an unified approach to vision-based control which can be used whether the model of the object is known or not. The key idea of the unified approach is to build a reference in a projective space which can be computed if the model is known or if an image of the object is available. Thus, only one low level visual servoing technique must be implemented at once. The strength of our approach is to keep the advantages of model-based and model-free methods and, at the same time, to avoid some of their drawbacks. In particular, we work in a projective space which is invariant to camera intrinsic parameters. This allows us to use the unified visual servoing approach with a zooming camera, contrarily to standard model-free approaches. There are various ways in which invariance to camera parameters can be obtained. Simple invariants to focal length and principal point have been proposed in [14] where the invariants are computed from interest points. In [9] invariance to all the camera intrinsic parameters has been obtained by choosing three points to build a projective transformation. Consequently, the selection of the three points raised the problem of the best choice. The problem is solved in this paper by building the projective transformation from all points available in the image. Moreover, the computation of the invariants is extended in this paper to generic non-planar curves in the image. Experiments on simulated data demonstrate the validity of the unified approach and the improvements over existing methods.

2 Theoretical Background

2.1 Model of the Object

For simplicity, I consider in the paper that the model of the object can be described by a set of 3D points. The theory can be generalized to different geometric shapes such straight lines or conics. In this paper, I suppose also that the object is non-planar. If the object is planar the method presented in the paper can work under the same hypotheses done for standard methods but not in the general case when intrinsic parameters at the convergence are different from the parameters during the learning [9]. Let $\mathcal{F}_0$ be a frame attached to a non-planar object. Suppose that the model is represented by the homogeneous coordinates of a discrete set of n 3D points $\boldsymbol{\mathcal{X}}_i = (X_i, Y_i, Z_i, 1)$ $(i = \{1, 2, ..., n\})$ with respect to $\mathcal{F}_0$. I consider also the case when the set of points is continuous and describes a generic closed curve in the 3D space, for example the boundary of a generic surface. In that case, the model is represented by a parametric representation of the curve $\boldsymbol{\mathcal{X}}(\tau) = (X(\tau), Y(\tau), Z(\tau), 1)$ where τ is a parameter of the representation.

2.2 Perspective Projection

Let $\mathcal{C}$ be the center of projection coinciding with the origin $\mathcal{O}$ of frame $\mathcal{F}$. Let the plane of projection be parallel to the plane $(\vec{x}, \vec{y})$. Without loss of generality

we can suppose that the distance between the two planes is 1. A 3D point $\boldsymbol{\mathcal{X}}_i = (X_i, Y_i, Z_i, 1) \in \mathbb{P}^3$ is projected to the point $\mathbf{m}_i \in \mathbb{P}^2$:

$$\mathbf{m}_i = \frac{1}{Z_i} \begin{bmatrix} \mathbf{R}_0 & \mathbf{t}_0 \end{bmatrix} \boldsymbol{\mathcal{X}}_i = (x_i, y_i, 1) \tag{1}$$

where $\mathbf{R}_0$ and $\mathbf{t}_0$ are respectively the rotation and the translation between frame $\mathcal{F}_0$ and $\mathcal{F}$. The point $\mathbf{m}_i$ is defined in the projective coordinate system $\mathcal{M} \in \mathbb{P}^2$. Similarly, the 3D curve $\boldsymbol{\mathcal{X}}(\tau)$ projects to the 2D curve $\mathbf{m}(\tau) = (x(\tau), y(\tau), 1)$.

2.3 Camera Model

Pinhole cameras perform a perspective projection of a 3D point. The information measured by the camera is a point $\mathbf{p}_i$ which depends on its internal parameters:

$$\mathbf{p}_i = \mathbf{K}\mathbf{m}_i = (u_i, v_i, 1) \tag{2}$$

where the triangular matrix $\mathbf{K}$ contains the camera internal parameters:

$$\mathbf{K} = \begin{bmatrix} f & s & u_0 \\ 0 & rf & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{3}$$

f is the focal length (pixels), u_0 and v_0 are the coordinates of the principal point (pixels), s is the skew and r is the aspect ratio. Note that the non-singular (3×3) matrix $\mathbf{K}$ defines a projective transformation (homography) from the projective coordinate system $\mathcal{M} \in \mathbb{P}^2$ to the projective coordinate system $\mathcal{P} \in \mathbb{P}^2$.

3 Two Separate Approaches to Vision-Based Control

In this paper, I consider the positioning of a single camera mounted on the robot end-effector. Suppose that we want to move the camera to a reference position with respect to the object. Note that, any feature or parameter in the reference frame will be marked with an asterisk symbol. Let $\mathcal{F}^*$ be the reference frame and let $\mathbf{R}_0^*$ and $\mathbf{t}_0^*$ be respectively the rotation and the translation between $\mathcal{F}^*$ and $\mathcal{F}_0$. The choice of the vision-based approach depends on the knowledge we have of the model of the object. For the sake of simplicity, I consider in this section that the object is described with a discrete set of points.

3.1 Model-Free Approach

If we do not know the model of the object we must use a model-free approach which is based on a teaching-by-showing step. It consists in driving the robot to the desired position and storing the corresponding reference image features $\mathbf{p}_i^*$, $\forall i \in \{1, 2, ..., n\}$. The approach is model-free since we do not need to know the model of the object to measure $\mathbf{p}_i^*$, but only a reference image taken during the preliminary learning step. After the robot or the object have been moved, the

problem is how to drive the robot to the desired position from the initial position using the current image points $\mathbf{p}_i$, $\forall i \in \{1, 2, ..., n\}$. As already mentioned in the introduction, several methods have been proposed for visual servoing. Generally speaking, the problem can be solved by minimizing a (6×1) error vector which can be computed as a function of the current and reference points:

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}(\mathbf{p}_1, \mathbf{p}_1^*, \mathbf{p}_2, \mathbf{p}_2^*, ..., \mathbf{p}_n, \mathbf{p}_n^*) \tag{4}$$

The error vector is null if and only if $\mathbf{p}_i = \mathbf{p}_i^*$, $\forall i \in \{1, 2, ..., n\}$. In that case, the robot is back to the reference position $\mathbf{R}_0^*$ and $\mathbf{t}_0^*$. Note that, the equivalence $\mathbf{R}_0 = \mathbf{R}_0^*$ and $\mathbf{t}_0 = \mathbf{t}_0^* \Leftrightarrow \mathbf{p}_i = \mathbf{p}_i^*$,$\forall i \in \{1, 2, ..., n\}$ is true only if the camera parameters do not change during the servoing. Otherwise, the reference points must be learned again.

3.2 Model-Based Approach

If the model of the object is known we do not need a preliminary learning step in order to drive the camera to the reference position $\mathbf{R}_0^*$ and $\mathbf{t}_0^*$. The model-based approach uses the model of the object $\boldsymbol{\mathcal{X}}_i$ and the current image points $\mathbf{p}_i$, $\forall i \in \{1, 2, ..., n\}$ in order to estimate the current camera pose. If $n = 4$ one must know the camera intrinsic parameters [2]. In this paper, I suppose that the camera intrinsic parameters are unknown. On the other hand, I suppose that a large number of points are available. In this case, we can compute the camera pose and the camera intrinsic parameters at the same time [5]. Indeed, from equation (1) and equation (2) we obtain $\zeta_i \mathbf{p}_i = \mathbf{P}\boldsymbol{\mathcal{X}}_i$, where ζ_i is an unknown scalar factor and the unknown projection matrix $\mathbf{P}$ is given by $\mathbf{P} = \mathbf{K}\left[\mathbf{R}_0 \ \mathbf{t}_0\right]$. From the equations above, we estimate $\mathbf{P}$ and then extract from this matrix the camera parameters $\mathbf{K}$ and the current camera position $(\mathbf{R}_0, \mathbf{t}_0)$ with respect to the object frame $\mathcal{F}_0$ [5]. Since we know the reference position $(\mathbf{R}_0^*, \mathbf{t}_0^*)$ with respect to $\mathcal{F}_0$ we can compute the displacement of the camera with respect to the reference position. Again, there are several ways of controlling the pose of the camera. The problem is generally solved by building a (6×1) error vector which can be computed as a function of the current and reference camera poses:

$$\boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}(\mathbf{t}_0, \mathbf{t}_0^*, \mathbf{R}_0, \mathbf{R}_0^*) \tag{5}$$

The error vector is null if and only if $\mathbf{t}_0 = \mathbf{t}_0^*$ and $\mathbf{R}_0 = \mathbf{R}_0^*$. The model-based approach can be used with a zooming camera since the intrinsic parameters are estimated on-line and they are not used to control the camera pose.

3.3 Incompatibility of the Two Approaches

There is an incompatibility problem which prevents the two approaches being used together. First of all, it is evident that the model-based task function in equation (5) can be measured if and only if the model of the object is known. On the other hand, the task function given in equation (4) can not be used with zooming cameras. The aim of this paper is to propose an unified approach which can be used in both cases. The strength of our approach is to keep the advantages of model-based and model-free methods and to avoid their drawbacks.

4 An Unified Visual Servoing Approach

The key idea of the new unified visual servoing approach is to always work in a projective space $\mathcal{Q} \in \mathbb{P}^2$ which can be computed from points belonging to the image space $\mathcal{P} \in \mathbb{P}^2$ (if the model is unknown) or points belonging to the projective space $\mathcal{M} \in \mathbb{P}^2$ (if the model is known). The approach does not need the explicit calibration of the camera and can be used even if the camera is zooming. Indeed, we will show that the projective space $\mathcal{Q}$ is invariant on camera intrinsic parameters. Invariance to camera intrinsic parameters is obtained by computing a projective transformation.

4.1 Computing the Transformation When the Model Is Unknown

When the model is unknown we use a teaching-by-showing technique. Suppose that the camera has been driven to frame $\mathcal{F}$. Thus, n points can be extracted from the corresponding image. Using all the image points, with projective coordinates $\mathbf{p}_i = (u_i, v_i, 1)$, we compute the following symmetric (3×3) matrix:

$$\mathbf{S}_p = \frac{1}{n}\sum_{i=1}^{n}\mathbf{p}_i\mathbf{p}_i^\top = \frac{1}{n}\begin{bmatrix} \sum_{i=1}^{n} u_i^2 & \sum_{i=1}^{n} u_i v_i & \sum_{i=1}^{n} u_i \\ \sum_{i=1}^{n} u_i v_i & \sum_{i=1}^{n} v_i^2 & \sum_{i=1}^{n} v_i \\ \sum_{i=1}^{n} u_i & \sum_{i=1}^{n} v_i & \sum_{i=1}^{n} 1 \end{bmatrix} = \begin{bmatrix} \sigma_{11} & \sigma_{12} & \sigma_{13} \\ \sigma_{12} & \sigma_{22} & \sigma_{23} \\ \sigma_{13} & \sigma_{23} & 1 \end{bmatrix} \tag{6}$$

This matrix has a simple geometric meaning. Indeed, σ_{13} and σ_{23} are the coordinates of the centroid of the n points, while σ_{11}, σ_{12} and σ_{22} are the second order moments of the set of points. If the observed points are not collinear and $n > 3$ then matrix $\mathbf{S}_p$ is symmetric positive definite and it can be written, using a Cholesky decomposition, as:

$$\mathbf{S}_p = \mathbf{T}_p\,\mathbf{T}_p^\top \tag{7}$$

where $\mathbf{T}_p$ is the following (3×3) non-singular upper triangular matrix:

$$\mathbf{T}_p = \begin{bmatrix} \sqrt{\sigma_{11} - \sigma_{13}^2 - \frac{(\sigma_{12} - \sigma_{13}\sigma_{23})^2}{\sigma_{22} - \sigma_{23}^2}} & \sqrt{\frac{(\sigma_{12} - \sigma_{13}\sigma_{23})^2}{\sigma_{22} - \sigma_{23}^2}} & \sigma_{13} \\ 0 & \sqrt{\sigma_{22} - \sigma_{23}^2} & \sigma_{23} \\ 0 & 0 & 1 \end{bmatrix} \tag{8}$$

This matrix defines a projective transformation from the projective image space $\mathcal{P} \in \mathbb{P}^2$ to a new projective space $\mathcal{Q} \in \mathbb{P}^2$. Each point $\mathbf{q}_i \in \mathcal{Q}$ is computed as:

$$\mathbf{q}_i = \mathbf{T}_p^{-1}\mathbf{p}_i = (a_i, b_i, 1) \tag{9}$$

I will show in Section 4.3 that $\mathbf{q}_i$ is independent on camera intrinsic parameters. In conclusion, a set of reference points $\mathbf{q}_i^*$ can be computed by using a reference image $\mathbf{p}_i^*$ in the previous equations (all parameters in the reference frame are marked with an asterisk).

4.2 Computing the Transformation When the Model Is Known

Suppose that we want to drive the robot to a desired position ($\mathbf{R}_0$, $\mathbf{t}_0$) with respect to the object frame $\mathcal{F}_0$. If the model $\boldsymbol{\mathcal{X}}_i$ is also known, for each point it is possible to compute, from equation (1), its perspective projection $\mathbf{m}_i = (x_i, y_i, 1)$. Using all the projected points, we compute the following symmetric (3×3) matrix:

$$\mathbf{S}_m = \frac{1}{n}\sum_{i=1}^{n}\mathbf{m}_i\mathbf{m}_i^\top = \frac{1}{n}\begin{bmatrix} \sum_{i=1}^{n} x_i^2 & \sum_{i=1}^{n} x_i y_i & \sum_{i=1}^{n} x_i \\ \sum_{i=1}^{n} x_i y_i & \sum_{i=1}^{n} y_i^2 & \sum_{i=1}^{n} y_i \\ \sum_{i=1}^{n} x_i & \sum_{i=1}^{n} y_i & \sum_{i=1}^{n} 1 \end{bmatrix} \tag{10}$$

Again, it is possible to decompose $\mathbf{S}_m$ using a Cholesky decomposition as:

$$\mathbf{S}_m = \mathbf{T}_m\,\mathbf{T}_m^\top \tag{11}$$

where $\mathbf{T}_m$ is an upper triangular matrix having the same form of matrix $\mathbf{T}_p$. Again, the non-singular matrix $\mathbf{T}_m$ can be used to define a projective transformation from the projective space $\mathcal{M} \in \mathbb{P}^2$ to the projective space $\mathcal{Q} \in \mathbb{P}^2$:

$$\mathbf{q}_i = \mathbf{T}_m^{-1}\mathbf{m}_i \tag{12}$$

Note that $\mathbf{T}_m$ does not depend on $\mathbf{K}$ since it is computed from $\mathbf{S}_m$. Thus, $\mathbf{q}_i$ is independent on camera intrinsic parameters. Again, a set of reference points $\mathbf{q}_i^*$ can be computed by using a reference position ($\mathbf{R}_0^*$, $\mathbf{t}_0^*$) in the previous equations.

4.3 Equivalence between the Two Computations

I show now that the points obtained from equation (9) and equation (12) are equivalent (therefore, the vector computed from equation (9) is also independent on camera intrinsic parameters). Since $\mathbf{p}_i = \mathbf{K}\mathbf{m}_i$, the matrix $\mathbf{S}_p$ can be written as a function of $\mathbf{S}_m$ and of the camera intrinsic parameters $\mathbf{K}$:

$$\mathbf{S}_p = \frac{1}{n}\sum_{i=1}^{n}\mathbf{p}_i\mathbf{p}_i^\top = \frac{1}{n}\sum_{i=1}^{n}\mathbf{K}\mathbf{m}_i\mathbf{m}_i^\top\mathbf{K}^\top = \mathbf{K}\left(\frac{1}{n}\sum_{i=1}^{n}\mathbf{m}_i\mathbf{m}_i^\top\right)\mathbf{K}^\top = \mathbf{K}\,\mathbf{S}_m\,\mathbf{K}^\top \tag{13}$$

Thus, from equations (7), (11) and (13) we obtain:

$$\mathbf{T}_p = \mathbf{K}\mathbf{T}_m \tag{14}$$

The equivalence between equation (9) and equation (12) is straightforward:

$$\mathbf{q}_i = \mathbf{T}_p^{-1}\mathbf{p}_i = \mathbf{T}_m^{-1}\mathbf{K}^{-1}\mathbf{p}_i = \mathbf{T}_m^{-1}\mathbf{K}^{-1}\mathbf{K}\mathbf{m}_i = \mathbf{T}_m^{-1}\mathbf{m}$$

In conclusion, we can compute the same vectors $\mathbf{q}_i$ from the knowledge of model of the object and the desired position (equation (12)) or from image points (equation (9)). The new projective space $\mathcal{Q}$ is independent on camera intrinsic parameters. Thus, the zoom of the camera can be controlled separately.

4.4 Extension to a Generic Non-planar Curve in Space

The extension of the method to a generic non-planar curve in space is straightforward. If the model of the curve is unknown we can measure the curve Θ in an image taken from frame $\mathcal{F}$. Thus, if $\mathbf{p}$ is a running point on Θ, the sum in equation (6) can be generalized to the following integral:

$$\mathbf{S}_p = \frac{\iint_\Theta \mathbf{p}\,\mathbf{p}^\top du\,dv}{\iint_\Theta du\,dv} = \frac{1}{\iint_\Theta du\,dv} \begin{bmatrix} \iint_\Theta u^2\,du\,dv & \iint_\Theta uv\,du\,dv & \iint_\Theta u\,du\,dv \\ \iint_\Theta uv\,du\,dv & \iint_\Theta v^2\,du\,dv & \iint_\Theta v\,du\,dv \\ \iint_\Theta u\,du\,dv & \iint_\Theta v\,du\,dv & \iint_\Theta du\,dv \end{bmatrix} \tag{15}$$

Since the curve is generic $\mathbf{S}_p > 0$ and it can be decomposed as $\mathbf{S}_p = \mathbf{T}_p\mathbf{T}_p^\top$. The (3×3) triangular matrix $\mathbf{T}_p$ defines again a change of projective coordinates from $\mathcal{P}$ to $\mathcal{Q}$ in $\mathbb{P}^2$ (i.e. $\mathbf{q} = \mathbf{T}_p^{-1}\mathbf{p}$ where $\mathbf{q}$ belongs to the transformed curve).

If a parametric representation of the curve $\boldsymbol{\mathcal{X}}(\tau)$ and the position $(\mathbf{R}_0, \mathbf{t}_0)$ of frame $\mathcal{F}$ with respect $\mathcal{F}_0$ are known, then it is possible to compute the projection $\mathbf{m}(\tau) = (x(\tau), y(\tau), 1) \in \Omega$. Thus, the sum in equation (10) can be generalized to the following integral:

$$\mathbf{S}_m = \frac{\iint_\Omega \mathbf{m}\,\mathbf{m}^\top dxdy}{\iint_\Omega dxdy} = \frac{1}{\iint_\Omega dxdy} \begin{bmatrix} \iint_\Omega x^2\,dxdy & \iint_\Omega xy\,dxdy & \iint_\Omega x\,dxdy \\ \iint_\Omega xy\,dxdy & \iint_\Omega y^2\,dxdy & \iint_\Omega y\,dxdy \\ \iint_\Omega x\,dxdy & \iint_\Omega y\,dxdy & \iint_\Omega dxdy \end{bmatrix} \tag{16}$$

Since the curve is generic $\mathbf{S}_m > 0$ and it can be decomposed as $\mathbf{S}_m = \mathbf{T}_m\mathbf{T}_m^\top$. The (3×3) triangular matrix $\mathbf{T}_m$ defines again a change of projective coordinates from $\mathcal{M}$ to $\mathcal{Q}$ in $\mathbb{P}^2$ (i.e. $\mathbf{q} = \mathbf{T}_m^{-1}\mathbf{m}$ where $\mathbf{q}$ belongs to the transformed curve).

I show now the equivalence of the two ways of computing the space $\mathcal{Q}$. Since $\mathbf{p} = \mathbf{K}\mathbf{m}$ then:

$$\mathbf{S}_p = \frac{\iint_\Theta \mathbf{p}\,\mathbf{p}^\top du\,dv}{\iint_\Theta du\,dv} = \frac{\iint_\Omega \mathbf{K}\mathbf{m}\,\mathbf{m}^\top\mathbf{K}^\top |J| dx\,dy}{\iint_\Omega |J| dx\,dy}$$

where the Jacobian J is:

$$J = \begin{vmatrix} \frac{\partial x}{\partial u} & \frac{\partial x}{\partial v} \\ \frac{\partial y}{\partial u} & \frac{\partial y}{\partial v} \end{vmatrix} = \frac{1}{k_{11}k_{22}}$$

Since J and $\mathbf{K}$ are independent on x and y (they depends on camera intrinsic parameters) they can be factored out from the integrals:

$$\mathbf{S}_p = \frac{\iint_\Theta \mathbf{p}\,\mathbf{p}^\top du\,dv}{\iint_\Theta du\,dv} = \mathbf{K}\frac{\iint_\Omega \mathbf{m}\,\mathbf{m}^\top dx\,dy}{\iint_\Omega dx\,dy}\mathbf{K}^\top = \mathbf{K}\mathbf{S}_m\mathbf{K}^\top$$

Finally, as done for the discrete set of points, the Cholesky decomposition of $\mathbf{S}_p$ gives a matrix $\mathbf{T}_p$ such that $\mathbf{T}_p = \mathbf{K}\mathbf{T}_m$. Thus, $\mathbf{q} = \mathbf{T}_p^{-1}\mathbf{p} = \mathbf{T}_m^{-1}\mathbf{K}^{-1}\mathbf{p} = \mathbf{T}_m^{-1}\mathbf{K}^{-1}\mathbf{K}\mathbf{m} = \mathbf{T}_m^{-1}\mathbf{m}$. This equation show the equivalence between the two ways of building space $\mathcal{Q}$ and that this space is independent on camera internal parameters. Again, one can build a reference curve $\mathbf{q}^* \in \mathcal{Q}$ from a reference curve $\mathbf{p}^* \in \mathcal{P}$ in the image or from the model and a reference position $(\mathbf{R}_0^*, \mathbf{t}_0^*)$.

5 The Control Law of the Unified Visual Servoing

The control of the camera pose is achieved by minimizing an error computed in the space $\mathcal{Q}$ invariant to camera intrinsic parameters. For the sake of generality, I present a control approach which can be used both with a discrete set of points and a 3D curve. Indeed, the curve can be sampled and considered as a collection of points to design the control. However, in the case of a real 3D curve it is probably better not to use points in the control law due to potential problems in matching between contours. The control of a generic set of points is achieved by stacking all the reference points of space $\mathcal{Q}$ in a $(3n\times1)$ vector $\mathbf{s}^* = (\mathbf{q}_1^*, \mathbf{q}_2^*, \cdots, \mathbf{q}_n^*)$. Similarly, the current points are stacked in the $(3n\times1)$ vector $\mathbf{s} = (\mathbf{q}_1, \mathbf{q}_2, \cdots, \mathbf{q}_n)$. If $\mathbf{s} = \mathbf{s}^*$ then the camera is back to the reference position whatever the camera intrinsic parameters. The derivative of vector $\mathbf{s}$ is:

$$\dot{\mathbf{s}} = \mathbf{L}\,\mathbf{v} \tag{17}$$

where the $(3n\times6)$ matrix $\mathbf{L}$ is called the interaction matrix and $\mathbf{v} = (\boldsymbol{\nu}, \boldsymbol{\omega})$ the velocity of the camera. The interaction matrix depends on current camera intrinsic parameters $\mathbf{K}$ and on the current depth distribution $\mathbf{Z} = (Z_1, Z_2, ..., Z_n)$ (both unknowns), on the current image coordinates $\mathbf{p}_i \in \mathcal{P}$ and on the invariant points $\mathbf{q}_i \in \mathcal{Q}$. In order to control the camera, we can use the task function approach [11] which has already been validated for image-based visual servoing in [4]. Consider the following (6×1) task function:

$$\mathbf{e} = \widehat{\mathbf{L}}^{+}(\mathbf{s} - \mathbf{s}^*) \tag{18}$$

where $\widehat{\mathbf{L}}^{+}$ is the pseudo-inverse of an approximation of $\mathbf{L}$ (since $\mathbf{K}$ and $\mathbf{Z}$ are unknown only rough approximations $\widehat{\mathbf{K}}$ and $\widehat{\mathbf{Z}}$ are used to compute the interaction matrix). Differentiating equation (18) we obtain:

$$\dot{\mathbf{e}} = \mathbf{J}\mathbf{v} \tag{19}$$

where the $(3n\times6)$ matrix $\mathbf{J}$ is called the Jacobian of the task. In order to control the movement of the camera we can use the following control law:

$$\mathbf{v} = -\lambda\mathbf{e} \tag{20}$$

where λ is a positive scalar tuning the speed of the convergence. Using this control law, the closed-loop equation is $\dot{\mathbf{e}} = -\lambda\mathbf{J}\mathbf{e}$. It is well known from control theory [11] that if $\mathbf{J} > 0$ then the task function $\mathbf{e}$ converge to zero and, in the absence of local minima and singularities, so does the error $\mathbf{s}-\mathbf{s}^*$. If the presence of local minima and singularities is due to the choice of the control scheme, then it can be avoided by choosing a different control law. The problem to know if, and in which case, a local minimum can be found is beyond the aim of this paper and it will be addressed in future work.

As already mentioned, the main improvement over existing techniques is the possibility to use a zooming camera during the servoing. A simple control strategy for the zoom is the following. Let $d = \sqrt{min_i(u_i^2, v_i^2, (u_i - \overline{u})^2, (v_i - \overline{v})^2)}$ be

the distance in pixels of the closest point to the border of the image (the size of the image being $\overline{u} \times \overline{v}$). We can keep the distance to a desired value d^* by using the following control law for the focal length:

$$\dot{f} = \gamma(d - d^*) \tag{21}$$

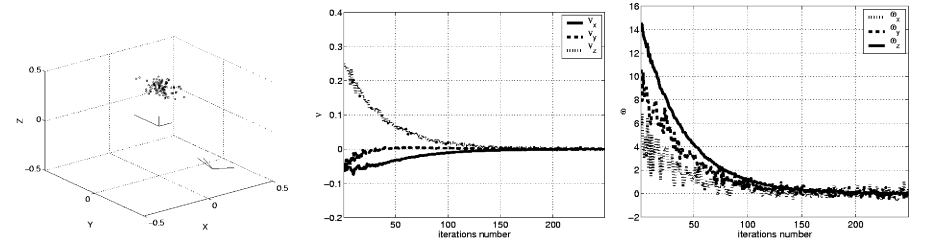
where γ is a positive scalar tuning the speed of the zoom. If $d < d^*$ (a point is too close to the border and it could get out of the image) then $\dot{f} < 0$ and the camera zoom out. If $d > d^*$ then $\dot{f} > 0$ and the camera zoom in to obtain a better resolution. Finally, if $d = d^*$ then $\dot{f} = 0$ and the camera does not zoom.

6 Experiments Using Simulated Data

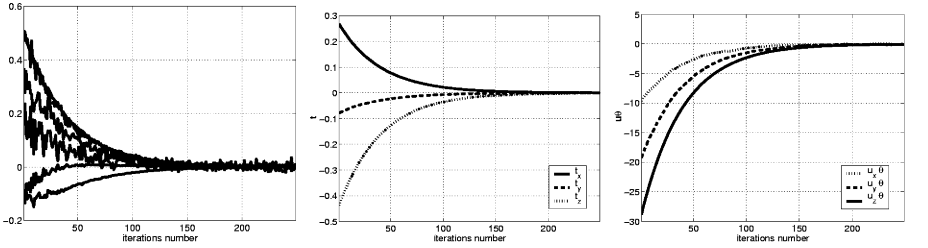
In this section, I validate the unified visual servoing approach using both a discrete and a continuous set of points. The problem of matching/tracking features, common to all visual servoing techniques, is beyond the aim of this paper and it has been already investigated in the literature. For the model-based approach, we need to match the model to the current image [8]. With the model-free approach, we need to match feature points [16] or curves [12] between the initial and reference views. Finally, when the camera is zooming we need to match images with different resolutions [3]. In the experiments, I focus on the general properties of the vision-based control approaches, therefore I consider that the matching/tracking problem has been solved for all visual servoing methods.

6.1 The Model Is Known

In the first experiment, I suppose that the 3D coordinates of a set of $n = 100$ points (randomly distributed in a sphere with 10 cm radius) are known with respect to the absolute frame $\mathcal{F}_0$ (see Figure 1(a)). A Gaussian noise with standard deviation $\sigma = 1$ is added to the current image. Suppose that we want to position the robot in a desired reference position $(\mathbf{R}_0^*, \mathbf{t}_0^*)$. In that case, we can estimate the current pose of the camera (Figure 1(d)) and use a standard position-based visual control law [15]. This control law (plotted in Figure 1(b) and (c)) makes the current frame converge to the reference frame since the rotation and the translation errors converge to zero (see Figures 1(e) and (f)). The aim of this experiment is to show that the unified approach proposed in the paper is able to execute exactly the same task. From the model and the reference position $(\mathbf{R}_0^*, \mathbf{t}_0^*)$ we compute the reference points $\mathbf{q}_i^*$ in space $\mathcal{Q}$ (see the points marked with a circle in Figure 2(a)). From the current image observed by the camera, we compute the points $\mathbf{q}_i$ (for example, the points marked with a square in Figure 2(a) corresponds to the initial image). The unified approach makes the errors $\mathbf{q}_i - \mathbf{q}_i^*$ converge to zero (except for noise) as shown in Figure 2(d) using the control law plotted in Figures 2(b) and (c). Consequently, the translation (Figure 2(e)) and the rotation (Figure 2(f)) errors between the current and the reference camera frames converge to zero. Obviously, the final points in the projective space $\mathcal{Q}$ (the points marked with a cross in Figure 2(a)) coincides to the reference points except for noise. Despite the presence of the noise the positioning accuracy for both methods is less than 1 mm and 0.1 degrees.

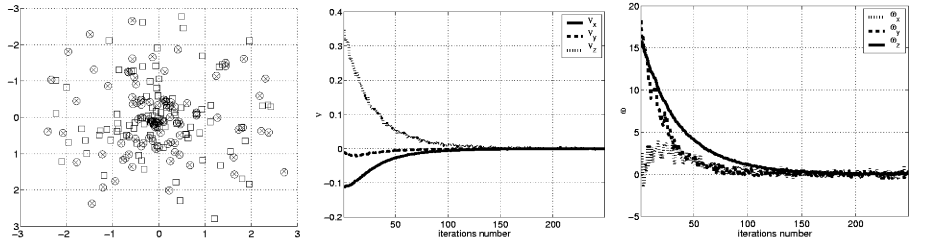


(a) Model (b) Translation speed (c) Rotation speed

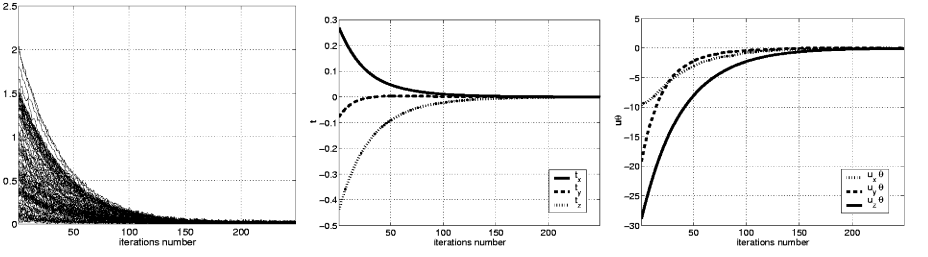


(d) Camera pose: $\mathbf{t}$, $\mathbf{u}\theta$ (e) Translation error (f) Rotation error

Fig. 1. Model-based visual servoing with a standard position-based control law. The translation and rotation errors are measured respectively in m and deg, while speeds are measured respectively in $\frac{m}{s}$ and $\frac{deg}{s}$.



(a) Points in space $\mathcal{Q}$ (b) Translation speed (c) Rotation speed



(d) Errors $\|\mathbf{q}_i - \mathbf{q}_i^*\|$ (e) Translation error (f) Rotation error

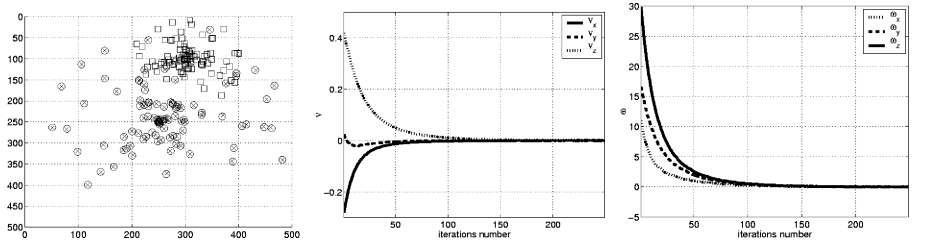
Fig. 2. The control law of the unified approach when the model is known. The translation and rotation errors are measured respectively in m and deg, while speeds are measured respectively in $\frac{m}{s}$ and $\frac{deg}{s}$.

6.2 The Model Is Unknown

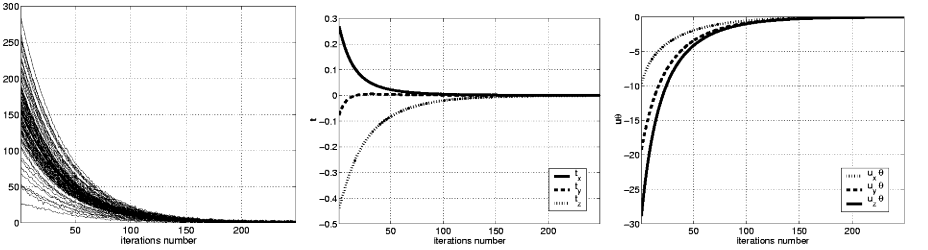
In this experiment, we consider the same set of points and Gaussian noise with $\sigma = 1$ added to image features. On the other hand, I suppose that we do not know the model of the object. Using the teaching-by-showing scheme, we store the reference image (the points marked with a circle in Figure 3(a)). Then, the camera is moved to its initial position (the corresponding points are marked with a square in Figure 3(a)). Using a standard image-based control law [4] (plotted in Figures 3(b and (c), the error $\mathbf{p}_i - \mathbf{p}_i^*$ (see Figure 3(d)) is servoed to zero (except for noise). Indeed, the final image (the points marked with a cross in Figure 3(a)) coincides to the reference image except for noise. Consequently, the translational and rotational errors in Figures 3(e) and (f) converge to zero with an accuracy of 1 mm and 0.1 degrees.

Since the model of the object is unknown, the position-based control law used in the previous experiment cannot be used here. On the other hand, the unified method proposed in the paper is able to achieve this same task. Indeed, from the initial and reference images we can compute the vectors $\mathbf{q}_i$ (the points marked with a square in Figure 4(a)) and $\mathbf{q}_i^*$ (the points marked with a circle in Figure 4(a)) in the projective space $\mathcal{Q}$. Using the control law plotted in Figures 4(b) and (c), the error $\mathbf{q}_i - \mathbf{q}_i^*$ (Figure 4(d)) is zeroed (except for noise) and the camera is back to the reference position (i.e. the translational and rotational errors in Figures 4(e) and (f) converge to zero with an accuracy of 1 mm and 0.1 degrees). Again, the points obtained at the convergence (the points marked with a cross in Figure 4(a)) coincide (except for noise) with the reference points in the invariant space. Similarly to the image-based method, despite the camera internal parameters $\widehat{\mathbf{K}}$ and the depth distribution $\widehat{\mathbf{Z}}$ are not exactly known, the control law is stable and converges. Obviously, if the calibration errors are big the performance of the visual servoing decreases (long time of convergence, unpredictable behavior). Note that Figures 2 and 4 are almost identical, the only difference being the random noise, since in both experiments we use the same initial and reference camera frames. This proves that the unified approach has the same behavior whether the model of the object is known or not.

When compared with standard methods, the unified approach is slightly less sensitive to noise than standard position-based visual servoing. This is probably due to the reconstruction of the pose of the camera in the position-based scheme (remark the level of noise in Figure 1(d)). Even if the unified approach does not need any reconstruction step, it needs to compute the projective transformation used to obtain $\mathbf{q}_i$ from $\mathbf{p}_i$. This, can explain why the unified approach is slightly more sensitive to noise than standard image-based visual servoing. A possible solution to this problem is to use a robust method for rejection of noise. For example, enforcing the epipolar geometry between two views of the object can give some constraints to reject points with a high level of noise. Finally, a problem common to all visual servoing methods is that some points can get out of the camera field of view during the servoing. With the unified approach, a possible solution to this problem is to use a zoom in order to bound the size of the object in the image as it is shown in the next experiment.

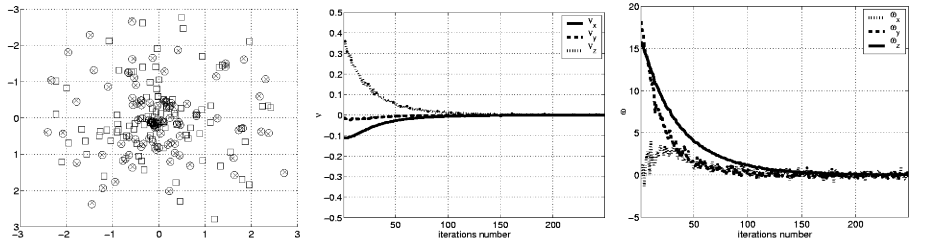


(a) Points in space $\mathcal{P}$ (b) Translation speed (c) Rotation speed

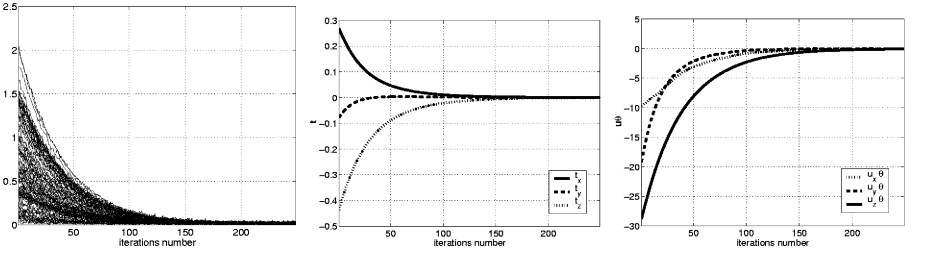


(d) Errors $\|\mathbf{p}_i - \mathbf{p}_i^*\|$ (e) Translation error (f) Rotation error

Fig. 3. Model-free visual servoing with an image-based control law. The points $\mathbf{p}_i$ are measure in pixels. The translation and rotation errors are measured respectively in m and deg, while speeds are measured respectively in $\frac{m}{s}$ and $\frac{deg}{s}$.



(a) Points in space $\mathcal{Q}$ (b) Translation speed (c) Rotation speed



(d) Errors $\|\mathbf{q}_i - \mathbf{q}_i^*\|$ (e) Translation error (f) Rotation error

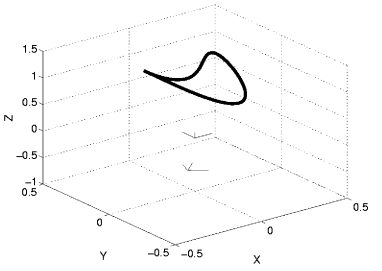
Fig. 4. The control law of the unified approach when the model is unknown. The points $\mathbf{q}_i$ have no units since they are normalized projective coordinates. The translation and rotation errors are measured respectively in m and deg, while speeds are measured respectively in $\frac{m}{s}$ and $\frac{deg}{s}$.

6.3 The Model Is Unknown and the Camera Is Zooming

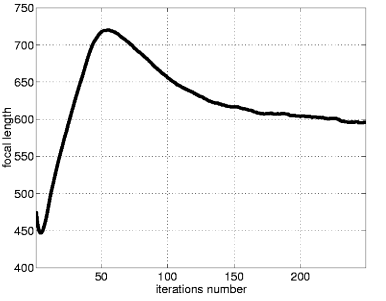
This experiment demonstrates the improvements provided by the unified approach over the existing methods and in particular the possibility of using a zooming camera when the model of the object is unknown. We want to position the robot with respect to the 3D curve given in Figure 5(a). I suppose now that the model is unknown (i.e. the model-based approach cannot be used) and that the camera is zooming during the servoing (i.e. the model-free approach cannot be used). On the other hand, we can still use a teaching-by-showing technique.

The reference curve (the dashed curve in Figure 5(c)) is taken with a camera having the following internal parameters: $f^* = 600$, $s^* = 0$, $r^* = 1.5$, $u_0^* = 300$ and $v_0^* = 250$. A Gaussian noise with $\sigma = 1$ pixel is added to the images. In order to reduce noise in the off-line learning step, the reference curve is the average curve obtained from several images. From the reference contour, we compute the matrix $\mathbf{S}_p^*$ (using $\mathbf{p}_i^*$ instead of $\mathbf{p}_i$ in equation (15)) and the transformation $\mathbf{T}_p^*$ which allows us to compute the transformed reference contour in the space $\mathcal{Q}$ (see the dashed curve in Figure 5(d)). After the robot has been moved to the initial position (the initial displacement of the camera is $\mathbf{t} = (-0.067, -0.031, -0.514)$ m and $\mathbf{r} = (-13.4, 20.0, -26.7)$ degrees), the initial image (the smallest curve in Figure 5(c)) is taken with a completely different camera at a lower resolution: $f = 500$, $s = 0$, $r = 1$, $u_0 = 250$ and $v_0 = 200$. Remark that not only the focal length and principal point are different but also the aspect ratio is changed. Using the current curve, we compute the matrix $\mathbf{S}_p$ and we extract the transformation $\mathbf{T}_p$ which allows us to compute the initial contour in the space $\mathcal{Q}$ (see Figure 5(d)). From the current and reference contours in space $\mathcal{Q}$ we compute the control law (i.e. the velocity sent to the robot controller) plotted in Figure 5(e) and (f). The control law is stable and the robot is driven back to the desired position as shown by Figure 5(g) and (h) where the error converges to zero. Due to the the noise in the image, the error is not exactly zero but the positioning accuracy is again less than 1 mm and 0.1 degrees.

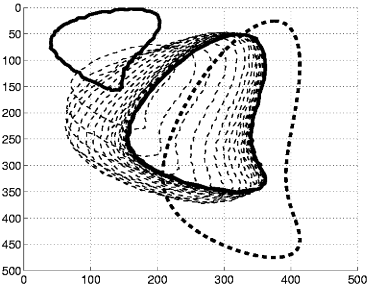
The behavior of the focal length, fixed by the control law of equation (21) with $d^* = 50$ pixels and $\gamma = 0.5$, is plotted in Figure 5(b). During the servoing the camera zooms out (from iteration 1 to 5 and from iteration 55 to 250 in Figure 5(b)) if the object is getting out the field of view. The camera zooms in (from iteration 5 to 54 in Figure 5(b)) if the object is too small in the image (thus, we can always have a good resolution). At the convergence, the camera focal length is $f \approx 600$ (see Figure 5(b)). However, the curve at the convergence (see Figure 5(c)) is not identical to the reference curve not only because the cameras have slightly different focal length but specially because the aspect ratio and principal point are completely different. A different principal point translates the curve while a different aspect ratio stretches the curve. On the other hand, the final curve in the space $\mathcal{Q}$ is identical, except for noise and sampling errors, to the reference curve since $\mathcal{Q}$ is independent on all camera internal parameters (see Figure 5(d)). Sampled trajectories of the curve in the image $\mathcal{P}$ and in the space $\mathcal{Q}$ are plotted respectively in Figure 5(c) and (d) (dashed lines).



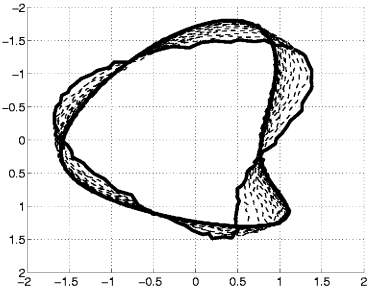
(a) Initial and reference frames



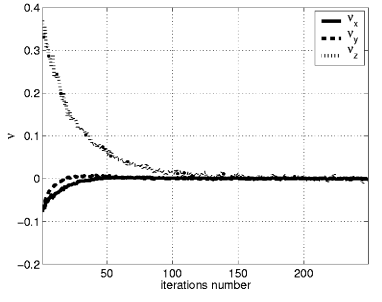
(b) Varying focal length



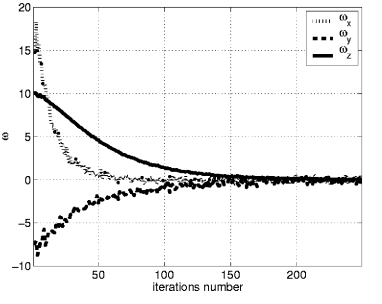
(c) Curve trajectory in $\mathcal{P}$



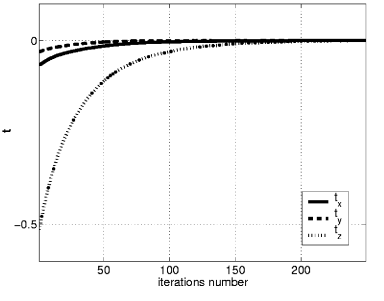
(d) Curve trajectory in $\mathcal{Q}$



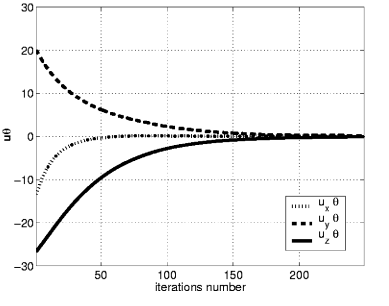
(e) Translation speed ($\frac{m}{s}$)



(f) Rotation speed ($\frac{deg}{s}$)



(g) Translation error (m)



(h) Rotation error (deg)

Fig. 5. The unified approach when model is unknown and the camera is zooming.

7 Conclusions

The approach proposed in the paper unifies model-based and model-free visual servoing techniques since it can be used whether the model of the object is known or not. The unified visual servoing scheme will be useful especially when a zooming camera is mounted on the end-effector of the robot. In that case, model-free visual servoing techniques cannot be used. As shown in the experiments, using the zoom during servoing is very important. The zoom can be used to enlarge the field of view of the camera if the object is getting out of the image and to bound the size of the object to improve the robustness of features extraction.

References

1. R. Basri, E. Rivlin, and I. Shimshoni. Visual homing: surfing on the epipole. *International Journal of Computer Vision*, 33(2):22–39, 1999.
2. D Dementhon and L. S. Davis. Model-based object pose in 25 lines of code. *International Journal of Computer Vision*, 15(1/2):123–141, June 1995.
3. Y. Dufournaud, C. Schmid, and R. Horaud. Matching images with different resolutions. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, pages 618–618, Hilton Head Island, South Carolina, USA, June 2000.
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, June 1992.
5. O. Faugeras. *Three-dimensional computer vision: a geometric viewpoint.* MIT Press, Cambridge, MA, 1993.
6. D.B. Gennery. Visual tracking of known three-dimensional objects. *International Journal of Computer Vision*, 7(3):243–270, April 1992.
7. 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, October 1996.
8. D. Jacobs and R. Basri. 3-d to 2-d pose determination with regions. *International Journal of Computer Vision*, 34(2-3):123–145, 1999.
9. E. Malis. Visual servoing invariant to changes in camera intrinsic parameters. In *Int. Conf. on Computer Vision*, vol. 1, pp. 704–709, Vancouver, Canada, July 2001.
10. E. Malis, F. Chaumette, and S. Boudet. 2 1/2 d visual servoing with respect to unknown objects through a new estimation scheme of camera displacement. *International Journal of Computer Vision*, 37(1):79–97, June 2000.
11. C. Samson, M. Le Borgne, B. Espiau. *Robot Control: the Task Function Approach*, vol. 22 of *Oxford Engineering Science Series.* Clarendon Press, Oxford, UK, 1991.
12. C. Schmid and A. Zisserman. The geometry and matching of lines and curves over multiple views. *International Journal of Computer Vision*, 40(3):199, 234 2000.
13. I. Shimshoni and J. Ponce. Probabilistic 3d object recognition. *International Journal of Computer Vision*, 36(1):51–70, January 2000.
14. M. Werman, S. Banerjee, S. Dutta Roy, and M. Qiu. Robot localization using uncalibrated camera invariants. In *IEEE Int. Conf. on Computer Vision and Pattern Recognition*, volume II, pages 353–359, Fort Collins, CO, June 1999.
15. W. J. Wilson, C. C. W. Hulls, and G. S. Bell. Relative end-effector control using cartesian position-based visual servoing. *IEEE Trans. on Robotics and Automation*, 12(5):684–696, October 1996.
16. Z. Zhang, R. Deriche, O. Faugeras, and Quang-Tuan Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. Technical Report 2273, INRIA, May 1994.