Intrinsics-free visual servoing with respect to straight lines

Ezio Malis, J.-J. Borrelly, P. Rives

Intrinsics-free visual servoing with respect to straight lines

Ezio Malis, Jean-Jacques Borrelly and Patrick Rives

INRIA, Sophia Antipolis, France, {first_name}.{second_name}@sophia.inria.fr

Abstract

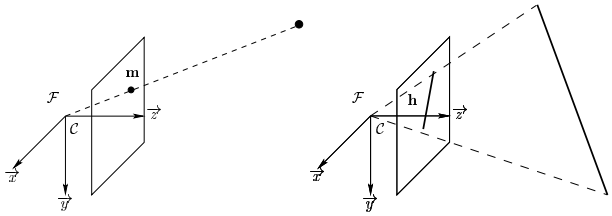
In this paper we propose a new approach for visual-servoing with respect to a set of 3D straight lines. The main difference with respect to previous approaches is that the new scheme can be used with a zooming camera or even if the reference image has been learned with a different camera. The zoom is particularly useful in order to keep the visual features in the camera field of view and/or to bound their size in the image reducing the influence of noise on features extraction. Experiments with a zooming camera have validated the vision-based control law.

1 Introduction

Visual servoing is a flexible technique to position an eye-in-hand camera with respect to an object for manipulation or inspection. Most of vision-based approaches [4] consider interest points as available visual features of the observed objects. On the other hand, man-made environments and objects often have richer geometrical primitives like for example straight lines. It is therefore of great interest to study how it is possible to position an eye-in-hand camera with respect to a set of straight lines. Only few works have extended the vision-based control to straight lines as for example [1, 2, 9] in the case of a single camera and [3] in the case of stereo vision. In this paper, we focus on vision-based robot control using a single camera mounted on the end-effector. When a 3D model of the scene is not available, eye-in-hand visual servoing techniques are based on a "teaching-by-showing" approach. With this approach, the robot is moved to a goal position (i.e. the reference position) and the camera is shown a set of 3D lines (i.e. the reference view). After the camera has been moved, any of the visual servoing schemes proposed in [1, 2, 9] can be used to reposition the camera with respect to the set of lines. For all schemes if the visual features currently observed in the image coincide with the features extracted from the reference image, the camera is back to the reference position with respect to the object. Whatever is the visual servoing method used to achieve the task, that will be true if and only if the camera intrinsic parameters at the convergence are the same parameters of the camera used for learning. Indeed, if the camera used during the servoing is different from the camera used to learn the reference image (or the camera intrinsic parameters have changed) the position of the camera with respect to the object will be completely different from the reference position. The objective of this paper is to extend the basic concepts of the visual servoing scheme proposed in [6] using points as visual features, to the positioning of an eye-in-hand camera with respect to a set of 3D straight lines. In [6], the versatility of the teaching-by-showing technique has been extended to the case when different cameras are used for learning the reference image and for servoing. Indeed, intrinsic parameters may significantly vary during the life of the vision system and/or they can be changed intentionally when using zooming cameras. If so, with current visual servoing techniques the reference image must be shown again. In some applications, learning again the reference image could be very difficult. On the other hand, the visual servoing technique proposed in [6] allows us to learn the reference image once and for all. In [6] three non collinear points are selected in order to build a change of projective coordinates. This change of projective coordinates defines a new projective space which is invariant to camera intrinsic parameters. However, applying the theory developed in [6] is not straightforward since straight lines and points are not dual in the 3D space. While three non collinear points always define an unique plane, this is not true when we consider any three non-parallel straight lines in the 3D space. As a consequence, a different change of projective coordinates must be defined in the case of straight lines. In this paper, we show how to build the invariant space and we propose a control law based on the task function approach [8]. The visual servoing scheme proposed in this paper is called intrinsics-free since it does not depend on camera internal parameters. Thus, contrarily to previous approaches not only we do not suppose that the camera is calibrated but also we can suppose that the camera intrinsic parameters are not fixed and the camera can zoom during the servoing.

2 Theoretical background

Let $\mathcal{C}$ be the center and π the plane of projection. A 3D point is projected to the point with homogeneous coordinates $\mathbf{m} = (x, y, 1)$ (see Figure 1(a)).



(a) Projection of a point (b) Projection of a line

Figure 1: *Projection of 3D features onto a plane.*

A pinhole camera perform the perspective projection of 3D geometric features. An image point with homogeneous coordinates $\mathbf{p} = (u, v, 1)$ measured in the image is:

$$\mathbf{p} = \mathbf{K}\,\mathbf{m} \tag{1}$$

where $\mathbf{K}$ is a non-singular upper triangular matrix containing the camera intrinsics parameters:

$$\mathbf{K} = \begin{bmatrix} f & s & u_0 \\ 0 & f\,r & v_0 \\ 0 & 0 & 1 \end{bmatrix}$$

where f is the focal length in pixels, s is the skew, r is the aspect ratio, u_0 and v_0 are coordinates of the principal point in pixels. A 3D line projects onto plane π to a 2D line with coordinates $\mathbf{h} = (\cos(\theta), \sin(\theta), -\rho)$ (see Figure 1(b)). A point $\mathbf{m}$ belongs to the straight line if and only if:

$$\mathbf{h}^\top \mathbf{m} = \cos(\theta)\; x + \sin(\theta)\; y - \rho = 0 \tag{2}$$

The same straight line measured in the image can be represented by its homogeneous coordinates $\mathbf{g} = (a, b, c)$. An image point $\mathbf{p} = (u, v, 1)$ belongs to the line if and only if:

$$\mathbf{g}^\top \mathbf{p} = a\; u + b\; v + c = 0 \tag{3}$$

Plugging equation (1) into equation (3) we obtain:

$$\mathbf{g}^\top \mathbf{K}\,\mathbf{m} = \mathbf{h}^\top \mathbf{m} = 0 \tag{4}$$

This means that the homogeneous coordinates of the straight line measured in the image are:

$$\mathbf{g} = \eta\; \mathbf{K}^{-\top} \mathbf{h} \tag{5}$$

where $\eta \neq 0$ is an arbitrary scalar factor. In order to have an unique representation of the straight line in the image, it is therefore necessary to normalize the coordinates of $\mathbf{g}$ by fixing the scalar factor. Let us suppose that it is possible to choose η such that:

$$\eta = \alpha(\mathbf{K})\; \beta(\mathbf{h}) \tag{6}$$

We will show in Section 4 how it is possible to choose a normalization which verify this hypothesis.

3 Invariance to camera parameters

In this section we show how it is possible to obtain from the straight lines extracted in the image some measures which are invariant to camera intrinsic parameters. Suppose that we extract n (with $n > 3$) different lines from the image and that the lines are not all parallel and not all coplanar in the 3D space. Consider the following positive symmetric matrix:

$$\mathbf{S}_g = \sum_{i=1}^{n} \mathbf{g}_i \mathbf{g}_i^\top$$

using equations (5) and (6) , this matrix can be written as:

$$\begin{aligned} \mathbf{S}_g &= \sum_{i=1}^{n} \mathbf{g}_i \mathbf{g}_i^\top = \sum_{i=1}^{n} \eta_i^2 \mathbf{K}^{-\top} \mathbf{h}_i \mathbf{h}_i^\top \mathbf{K}^{-1} \\ &= \alpha^2(\mathbf{K}) \mathbf{K}^{-\top} \left(\sum_{i=1}^{n} \beta_i^2(\mathbf{h}_i) \mathbf{h}_i \mathbf{h}_i^\top \right) \mathbf{K}^{-1} \end{aligned} \tag{7}$$

Note that it is extremely important to suppose that η can be written as in equation (6) in order to factor out from the sum the scalar depending only on camera intrinsic parameters. Let $\mathbf{S}_h$ be the following positive symmetric matrix which does not depend on camera intrinsic parameters:

$$\mathbf{S}_h = \sum_{i=1}^{n} \beta_i^2(\mathbf{h}_i) \mathbf{h}_i \mathbf{h}_i^\top = \begin{bmatrix} \sigma_1 & \sigma_2 & \sigma_4 \\ \sigma_2 & \sigma_3 & \sigma_5 \\ \sigma_4 & \sigma_5 & \sigma_6 \end{bmatrix} \tag{8}$$

Let $\boldsymbol{\sigma} = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5, \sigma_6)$ a (6×1) vector given in the Appendix containing the entries of $\mathbf{S}_h$. The matrix $\mathbf{S}_h$ can be factored, using a Cholesky decomposition, as follows:

$$\mathbf{S}_h = \mathbf{T}_h \mathbf{T}_h^\top \tag{9}$$

where $\mathbf{T}_h$ is the following lower triangular matrix:

$$\mathbf{T}_h = \begin{bmatrix} \tau_1 & 0 & 0 \\ \tau_2 & \tau_3 & 0 \\ \tau_4 & \tau_5 & \tau_6 \end{bmatrix}$$

Let $\boldsymbol{\tau} = (\tau_1, \tau_2, \tau_3, \tau_4, \tau_5, \tau_6)$ be a (6×1) vector containing the entries of $\mathbf{T}_h$. The vector $\boldsymbol{\tau}$ given in the Appendix is a function of $\boldsymbol{\sigma}$. If we factorize the matrix $\mathbf{S}_g$ using the Cholesky decomposition we obtain:

$$\mathbf{S}_g = \mathbf{T}_g \mathbf{T}_g^\top \tag{10}$$

Then, from equations (7), (9) and (10) we obtain:

$$\mathbf{T}_g = \alpha(\mathbf{K}) \mathbf{K}^{-\top} \mathbf{T}_h \tag{11}$$

The matrix $\mathbf{T}_g$, measured from equation (10), can be used to define a change of projective coordinates. Consider the lines in the transformed space:

$$\mathbf{l} = \mathbf{T}_g^{-1}\; \mathbf{g} \tag{12}$$

It is easy to prove that the line $\mathbf{l}$ is independent on camera intrinsic parameters. Indeed, from equations (3) and (11) we obtain:

$$\mathbf{l} = \mathbf{T}_g^{-1}\,\mathbf{g} = \frac{\eta}{\alpha(\mathbf{K})}\mathbf{T}_h^{-\top}\mathbf{K}^{\top}\mathbf{K}^{-\top}\mathbf{h} = \beta(\mathbf{h})\mathbf{T}_h^{-\top}\mathbf{h} \tag{13}$$

The expression of the scalar $\beta(\mathbf{h})$ is obtained from the normalization of the homogeneous coordinates $\mathbf{g}$ of the straight line, as shown in the next section.

4 Normalization of the line coordinates

As already mentioned, it is extremely important to normalize. From equation (3), we obtain the:

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \eta \begin{bmatrix} \frac{1}{f}\cos\theta \\ \frac{1}{fr}\sin\theta - \frac{s}{f^2 r}\cos\theta \\ -\rho - \frac{v_0}{fr}\sin\theta + \frac{sv_0 - fru_0}{f^2 r}\cos\theta \end{bmatrix} \tag{14}$$

The normalization of the homogeneous coordinates can be obtained by fixing arbitrarily one of them or by imposing one constraints. For example, by setting $\|\mathbf{g}\| = \sqrt{a^2+b^2+c^2} = 1$. Unfortunately, using this constraint the scalar η can not be written as in equation (6) since $\eta = \sqrt{(\mathbf{h}^{\top}\mathbf{K}^{-1}\mathbf{K}^{-\top}\mathbf{h})^{-1}}$. A first possible normalization is obtained by fixing one of the coordinates to 1:

$$\begin{cases} a = 1 & \text{if } a \neq 0 \\ b = 1 & \text{if } a = 0 \end{cases}$$

Thus, η can be decomposed as follows:

$$\begin{cases} \eta = f\frac{1}{\cos(\theta)} \Rightarrow \alpha(\mathbf{K}) = f, \beta(\mathbf{h}) = \frac{1}{\cos(\theta)} & \text{if } a \neq 0 \\ \eta = fr\frac{1}{\sin(\theta)} \Rightarrow \alpha(\mathbf{K}) = fr, \beta(\mathbf{h}) = \frac{1}{\sin(\theta)} & \text{if } a = 0 \end{cases}$$

With this normalization one must be careful to check if $a = 0$ (i.e. the straight line is parallel to the $\overrightarrow{u}$ axis). In order to avoid the possible numerical instability one can choose to compute the matrix $\mathbf{S}_h$ by using only lines which are not parallel to the $\overrightarrow{u}$ axis. A second possible normalization can be obtained if we can suppose that the skew is $s = 0$ and aspect ratio is $r = 1$. Indeed, for most of commercial camera, it is generally a very good approximation to suppose that the retinal axes are orthogonal ($s = 0$) and that the pixels are squares ($r = 1$). In that case, one can normalize the homogeneous coordinates of the straight line by setting $a^2 + b^2 = 1$. Thus, the scalar factor is:

$$\eta = f \;\Rightarrow\; \alpha(\mathbf{K}) = f,\; \beta(\mathbf{h}) = 1 \tag{15}$$

For the sake of simplicity, we consider from now that $s = 0$ and $r = 1$ and we use the normalization given in equation (15). The experimental results given in Section 6 show that it is a very good approximation for the camera we have used.

5 Control of the 6 d.o.f. of the robot

Let $\mathbf{t}$ and $\mathbf{R}$ be respectively the translation and the rotation between the reference camera frame $\mathcal{F}^*$ and the current camera frame $\mathcal{F}$. Let $\mathbf{r} = \theta\mathbf{u}$ be the (3×1) vector containing the axis of rotation $\mathbf{u}$ and the angle of rotation θ ($0 \leq \theta < 2\pi$). Then, $\boldsymbol{\xi} = (\mathbf{t}, \mathbf{r})$ is a (6×1) vector containing global coordinates of an open subset $\mathcal{S} \subset \mathbb{R}^3 \times SO(3)$. Suppose that a reference image of the scene, corresponding to the reference position $\boldsymbol{\xi}^* = 0$ has been stored in a previous learning step. The control of the camera is achieved by stacking all the reference lines in a ($3n$×1) vector $\mathbf{s}^*(\boldsymbol{\xi}^*) = (\mathbf{l}_1^*, \mathbf{l}_2^*, \cdots, \mathbf{l}_n^*)$. Similarly, the lines observed in the current image are stacked in the ($3n$×1) vector $\mathbf{s}(\boldsymbol{\xi}) = (\mathbf{l}_1, \mathbf{l}_2, \cdots, \mathbf{l}_n)$. If $\mathbf{s}(\boldsymbol{\xi}) = \mathbf{s}^*(\boldsymbol{\xi}^*)$ then $\boldsymbol{\xi} = \boldsymbol{\xi}^*$ and the camera is back to the reference position whatever the camera intrinsic parameters. The derivative of vector $\mathbf{s}$ is:

$$\dot{\mathbf{s}} = \mathbf{L}\,\mathbf{v} \tag{16}$$

where the ($3n$×6) matrix $\mathbf{L}$ is called the interaction matrix and $\mathbf{v} = (\boldsymbol{\nu}, \boldsymbol{\omega})$ the velocity of the camera. The interaction matrix is obtained by stacking together all the (3×6) interaction matrices corresponding to a single line:

$$\mathbf{L} = (\mathbf{L}_1; \mathbf{L}_2; ...; \mathbf{L}_n)$$

where $\dot{\mathbf{l}}_i = \mathbf{L}_i\,\mathbf{v}$. The derivative of the line $\mathbf{l}_i$ is:

$$\dot{\mathbf{l}}_i = \frac{\dot{\beta}(\mathbf{h}_i)}{\beta(\mathbf{h}_i)}\mathbf{l}_i + \beta(\mathbf{h}_i)\frac{d\mathbf{T}_h^{-1}}{dt}\mathbf{h}_i + \beta(\mathbf{h}_i)\mathbf{T}_h^{-1}\dot{\mathbf{h}}_i \tag{17}$$

From equation (15), we obtain $\beta(\mathbf{h}_i) = 1$ and consequently $\dot{\beta}(\mathbf{h}_i) = 0$. Note also that:

$$\frac{d\mathbf{T}_h^{-1}}{dt} = -\mathbf{T}_h^{-1}\,\frac{d\mathbf{T}_h}{dt}\,\mathbf{T}_h^{-1}$$

thus equation (17) can be written as follows:

$$\dot{\mathbf{l}}_i = \mathbf{T}_h^{-1}\left(\dot{\mathbf{h}}_i - \frac{d\mathbf{T}_h}{dt}\mathbf{l}_i\right)$$

Since $\mathbf{T}_h$ is a function of vector $\boldsymbol{\tau}$, we can write:

$$\frac{d\mathbf{T}_h(\boldsymbol{\tau})}{dt}\,\mathbf{l}_i = \mathbf{F}(\mathbf{l}_i)\,\dot{\boldsymbol{\tau}}$$

where $\mathbf{l}_i = (l_{1i}, l_{2i}, l_{3i})$:

$$\mathbf{F}(\mathbf{l}_i) = \begin{bmatrix} l_{1i} & 0 & 0 & 0 & 0 & 0 \\ 0 & l_{1i} & l_{2i} & 0 & 0 & 0 \\ 0 & 0 & 0 & l_{1i} & l_{2i} & l_{3i} \end{bmatrix}$$

The vector $\boldsymbol{\tau}$ is a function of $\boldsymbol{\sigma}$, thus:

$$\dot{\boldsymbol{\tau}} = \mathbf{J}\,\dot{\boldsymbol{\sigma}}$$

where $\mathbf{J} = \partial\boldsymbol{\tau}/\partial\boldsymbol{\sigma}$ is the Jacobian matrix. Instead of computing directly the matrix $\mathbf{J}$ it is easier to compute the inverse Jacobian $\mathbf{J}^{-1} = \partial\boldsymbol{\sigma}/\partial\boldsymbol{\tau}$:

$$\frac{\partial\boldsymbol{\sigma}}{\partial\boldsymbol{\tau}} = \begin{bmatrix} 2\,\tau_1 & 0 & 0 & 0 & 0 & 0 \\ \tau_2 & \tau_1 & 0 & 0 & 0 & 0 \\ 0 & 2\,\tau_2 & 2\,\tau_3 & 0 & 0 & 0 \\ \tau_4 & 0 & 0 & \tau_1 & 0 & 0 \\ 0 & \tau_4 & \tau_5 & \tau_2 & \tau_3 & 0 \\ 0 & 0 & 0 & 2\,\tau_4 & 2\,\tau_5 & 2\,\tau_6 \end{bmatrix}$$

Thus, the Jacobian is obtained as $\mathbf{J} = \left(\frac{\partial\boldsymbol{\sigma}}{\partial\boldsymbol{\tau}}\right)^{-1}$. The derivative of vector $\boldsymbol{\sigma}$ is related to the derivative of all lines $\mathbf{h}_i$ $(i = 1, 2, ..., n)$:

$$\dot{\boldsymbol{\sigma}} = \sum_{i=1}^{n} \frac{\partial\boldsymbol{\sigma}}{\partial\mathbf{h}_i}\dot{\mathbf{h}}_i \tag{18}$$

The derivative of the homogeneous coordinates of a line is related to the velocity of the camera by the following relationship:

$$\dot{\mathbf{h}} = \mathbf{H}\,\mathbf{v}$$

where, after setting $s\theta = \sin(\theta)$ and $c\theta = \cos(\theta)$:

$$\mathbf{H} = \begin{bmatrix} -\gamma\,c\theta\,s\theta, -\gamma\,s^2\theta, \gamma\,\rho\,s\theta, \rho\,c\theta\,s\theta, \rho\,s^2\theta, s\theta \\ \gamma\,c^2\theta, \gamma\,c\theta\,s\theta, -\gamma\,\rho\,c\theta, -\rho\,c^2\theta, -\rho\,c\theta\,s\theta, -c\theta \\ -\kappa\,c\theta, -\kappa\,s\theta, \kappa\,\rho, -(1+\rho^2)\,s\theta, (1+\rho^2)\,c\theta, 0 \end{bmatrix}$$

where, if $\boldsymbol{\pi} = (n_1, n_2, n_3, -d)$ are the homogeneous coordinates of one of the two planes defining the line $\mathbf{h}$ in the 3D space, then $\gamma = (n_1\ s\theta - n_2\ c\theta)/d$ and $\kappa = (n_1\,\rho\,c\theta + n_2\,\rho\,s\theta + n_3)/d$ (see [2] for more details). Thus, equation (18) can also be written as:

$$\dot{\boldsymbol{\sigma}} = \mathbf{B}\,\mathbf{v}$$

where $\mathbf{B} = \sum_{i=1}^{n} \frac{\partial\boldsymbol{\sigma}}{\partial\mathbf{h}_i}\mathbf{H}_i$. Finally, the interaction matrix relative to the straight line $\mathbf{l}_i$ is:

$$\mathbf{L}_i = \mathbf{T}_h^{-1}\,(\mathbf{H}_i - \mathbf{F}_i\,\mathbf{J}\mathbf{B})$$

This matrix depends on the planes distribution $\boldsymbol{\pi} = (\boldsymbol{\pi}_1, \boldsymbol{\pi}_2, ..., \boldsymbol{\pi}_n)$ (similarly to the interaction matrix of the image-based visual servoing [2]) and on the camera intrinsic parameters $\mathbf{K}(t)$ which can eventually vary during the servoing. In order to control the camera, we can use the task function approach [8] which has already been validated for the image-based visual servoing in [2]. Consider the following (6×1) task function:

$$\mathbf{e} = \widehat{\mathbf{L}}^{+}(\mathbf{s} - \mathbf{s}^*) \tag{19}$$

where $\widehat{\mathbf{L}}^{+}$ is the pseudo-inverse of an approximation of $\mathbf{L}$ (since $\mathbf{K}$ and $\boldsymbol{\pi}$ are unknown only approximations $\widehat{\mathbf{K}}$ and $\widehat{\boldsymbol{\pi}}$ are used to compute the interaction matrix). Despite we use only approximations $\widehat{\mathbf{K}}$ and $\widehat{\boldsymbol{\pi}}$, the control law is stable as it will be shown by the experimental results. We are currently working to the analysis of the robustness domain as done in [5] in the case of feature points. Differentiating equation (19) we obtain:

$$\dot{\mathbf{e}} = \mathbf{A}\mathbf{v} \tag{20}$$

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

$$\mathbf{v} = -\lambda\mathbf{e} \tag{21}$$

where λ is a positive scalar tuning the speed of the convergence. Using this control law, the closed-loop equation is:

$$\dot{\mathbf{e}} = -\lambda\mathbf{A}\mathbf{e}$$

It is well known from control theory [8] that if $\mathbf{A} > 0$ then the task function $\mathbf{e}$ converge to zero. Thus, if $\|\mathbf{s} - \mathbf{s}^*\|$ is sufficiently small and the interaction matrix $\mathbf{L}$ is full rank then $\mathbf{e} = 0$ implies $\mathbf{s} = \mathbf{s}^*$. The local asymptotic stability of the system can be proved considering the linearized system:

$$\dot{\mathbf{e}} = -\lambda\mathbf{A}(\boldsymbol{\xi}^*)\mathbf{e}$$

where $\mathbf{A}(\boldsymbol{\xi}^*) = \widehat{\mathbf{L}}^{+}(\boldsymbol{\xi}^*)\mathbf{L}(\boldsymbol{\xi}^*)$. The system is locally stable if $\mathbf{A}(\boldsymbol{\xi}^*) > 0$ since in that case $\mathbf{A}(\boldsymbol{\xi}^*)$ has eigenvalues with positive real part. However, to prove the local asymptotic convergence of $\mathbf{e}$ to zero, we need also to show that $\mathbf{s} - \mathbf{s}^*$ never belongs to $\mathrm{Ker}(\widehat{\mathbf{J}}^{+})$. This means that it exists a neighborhood $\mathcal{U}$ of $\boldsymbol{\xi}^*$ such that $\mathbf{e} = \widehat{\mathbf{L}}^{+}(\boldsymbol{\xi}^*)(\mathbf{s} - \mathbf{s}^*) \neq 0$, $\forall\boldsymbol{\xi} \in \mathcal{U}$ (i.e. $\mathbf{e} = 0$ only if $\mathbf{s}(\boldsymbol{\xi}) = \mathbf{s}^*$). Let us suppose that $\mathbf{s}(\boldsymbol{\xi}) \neq \mathbf{s}^*$ and therefore $\boldsymbol{\xi} \neq \boldsymbol{\xi}^* = 0$. The Taylor development of $\mathbf{s}(\boldsymbol{\xi})$ in a neighborhood of $\boldsymbol{\xi}^* = 0$ is:

$$\mathbf{s} - \mathbf{s}^* = \mathbf{L}(\boldsymbol{\xi}^*)\,\boldsymbol{\xi} + O^2(\boldsymbol{\xi}) \tag{22}$$

Multiplying by $\boldsymbol{\xi}^T\widehat{\mathbf{L}}^{+}(\boldsymbol{\xi}^*)$ (where $\boldsymbol{\xi}^T\widehat{\mathbf{L}}^{+}(\boldsymbol{\xi}^*) \neq 0$ since $\boldsymbol{\xi} \neq 0$ and $\widehat{\mathbf{L}}^{+}(\boldsymbol{\xi}^*)$ is full rank) both sides of equation (22) we obtain:

$$\boldsymbol{\xi}^T\widehat{\mathbf{L}}^{+}(\boldsymbol{\xi}^*)(\mathbf{s} - \mathbf{s}^*) = \boldsymbol{\xi}^T\widehat{\mathbf{L}}^{+}(\boldsymbol{\xi}^*)\mathbf{L}(\boldsymbol{\xi}^*)\boldsymbol{\xi} + O^3(\boldsymbol{\xi})$$

remember that if $\mathbf{A} = \widehat{\mathbf{L}}^{+}(\boldsymbol{\xi}^*)\mathbf{L}(\boldsymbol{\xi}^*) > 0$ then $\boldsymbol{\xi}^T\mathbf{A}\boldsymbol{\xi} \geq 2\sigma\|\boldsymbol{\xi}\|^2$, where $\sigma > 0$ is the minimum singular value of the positive definite matrix $\mathbf{A} + \mathbf{A}^T$. If $\widehat{\mathbf{L}}^{+}(\boldsymbol{\xi}^*)(\mathbf{s} - \mathbf{s}^*) = 0$ then:

$$0 \geq 2\sigma\|\boldsymbol{\xi}\|^2 + O^3(\boldsymbol{\xi})$$

that means:

$$\|\boldsymbol{\xi}\|^2 \leq |O^3(\boldsymbol{\xi})|$$

which is impossible since, by definition of $O^3(\boldsymbol{\xi})$, it exists a neighborhood of $\boldsymbol{\xi}^*$ in which:

$$\|\boldsymbol{\xi}\|^2 > |O^3(\boldsymbol{\xi})|$$

Therefore, $\mathbf{e} = \widehat{\mathbf{L}}^{+}(\boldsymbol{\xi}^*)(\mathbf{s} - \mathbf{s}^*) \neq 0$ in a neighborhood of $\boldsymbol{\xi}^*$ and the system is locally asymptotically stable.

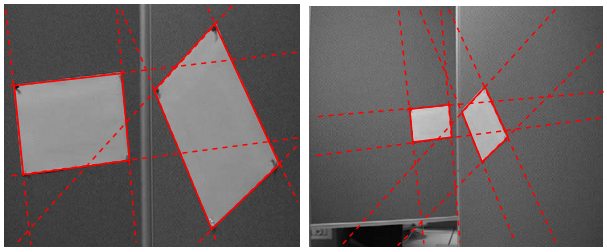
6 Experimental results

The visual servoing scheme proposed in the paper has been tested on the 3 d.o.f. system Argés at INRIA Sophia-Antipolis. The Argés monocular system is an experimental platform used to develop active vision algorithms. The hardware is made of:

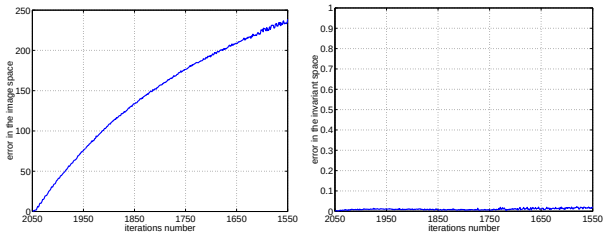
- a Computer controlled CCD Camera Acom1 with a f=5.9 to 47.2 mm zoom-lens, motor iris, numerical auto-focus, white balance, plus rs232C and video interface;
- a Pan-tilt turret, from RobotSoft, with resolution of 3.0 minutes of arcs, a 4 lbs capacity and a speed up to 300 deg/sec, using constant current bipolar motor drives, via a rs232C interface;
- a linear degree of freedom, from CharlyRobot, with a resolution of 0.1 mm, using a slow screw driven control;

6.1 Stationary zooming camera

To test whether the theory presented in the paper is a reasonable approximation it is important to prove the invariance of space $\mathcal{Q}$ to changes in camera intrinsic parameters. In this experiment, the focal length of the stationary camera changes continuously approximatively from 2250 pixels to 1550 pixels. The corresponding initial and final images are given in Figure 2(a) and (b). For each image, 8 lines (corresponding to the 4 borders of the 2 rectangles) are extracted. The first image (corresponding to the image in Figure 2(a)) is chosen as the reference image.



(a) Initial Image (b) Final Image



(c) Image 3 (d) Image 4

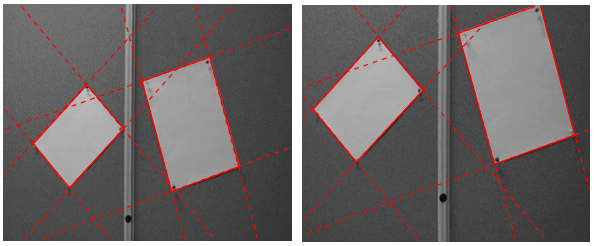
Figure 2: *Experiment using a zooming camera. The focal length changes continuously approximatively from 2250 pixels to 1550 pixels.*

We prove the invariance to camera intrinsic parameters using the normalization of equation (15). A line in the image has coordinates $(\cos(\psi_i), \sin(\psi_i), \nu_i)$. Figure 2(c) shows that the global error (i.e. the sum of the errors $(\psi_i - \psi_i^*)^2 + (\nu_i - \nu_i^*)^2 \ \forall i \in \{1, 2, ..., 8\}$) in the image space increase while the camera is zooming. When considering the coordinates $(\cos(\phi_i), \sin(\phi_i), \mu_i)$ of the lines $\mathbf{l}_i$ in the invariant space, we can observe in Figures 2(d) that the global error (i.e. the sum of the errors $(\phi_i - \phi_i^*)^2 + (\mu_i - \mu_i^*)^2 \ \forall i \in \{1, 2, ..., 8\}$) in the invariant space is not only close to zero but also practically constant. This means that the model with three parameters (i.e. $s = 0$ and $r = 1$) is a very good approximation for our camera. Obviously, the error is not exactly null because of noise in features extraction. This experiment proves that even if the camera is zooming, the error in the invariant space remain constant (except for noise). Thus, one can control the camera directly in the invariant space as it is shown in the next experiment.

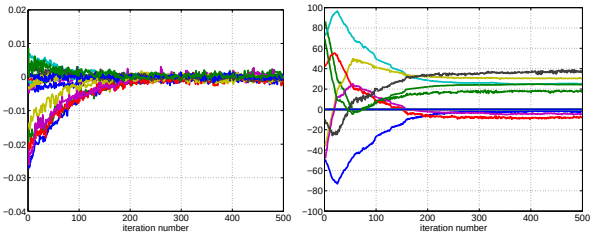
6.2 Intrinsics-free visual servoing

We consider the positioning task of the eye-in-hand camera with respect to a set of 8 straight lines. The reference image (see Figure 3(a)) is learned with a focal length $f^* = 1364$ pixels. Then, the camera is moved to its initial position. The initial displacement is $t_x = 140$ mm for the linear degree of freedom, $r_x = 3$ degree and $r_y = 3$ degrees. At the initial position, the camera zoom in and the focal length is changed to $f = 1671$ pixels. The corresponding initial image is given in Figure 3(b). The focal length of the camera is fixed during the servoing (in this experiment, the camera does not zoom). However, it is different from the reference focal length. Thus, despite the camera can be repositioned, the final image will be different from the reference one. Finally, since the camera parameters are unknown we use in the control law a very bad approximation of the focal length $\widehat{f} = 700$ and we suppose that the principal point is in the center of the image. During the servoing, the lines are tracked using the algorithm proposed in [7]. The control law is plotted in Figures 3(e) and (f). Despite the camera internal parameters $\widehat{\mathbf{K}}$ and the plane distribution $\widehat{\boldsymbol{\pi}}$ are not exactly known, the control law is stable and converges. The error in the invariant space (Figure 3(b)) is zeroed (except for noise) and the camera is back to the reference position (i.e. the translational and rotational errors in Figures 3(g) and (h) converge to zero with an accuracy of 1 *mm* and 0.1 *degrees*). While the straight lines observed at the convergence are different from the reference lines (i.e. the error in the image does not converge to zero as shown in Figure 3(d)), the final and reference lines measured in the invariant

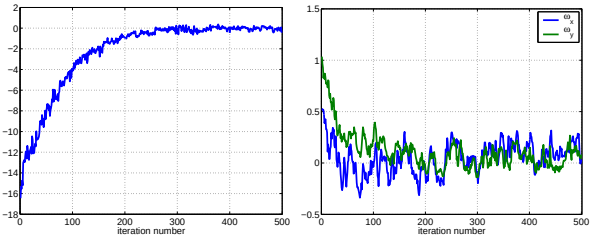
space coincide (except for noise).



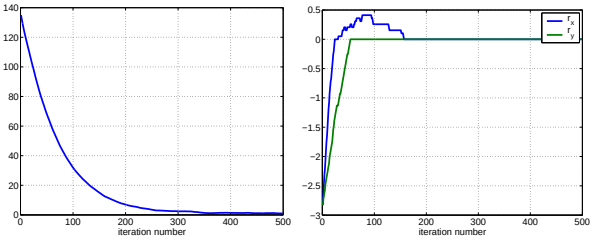
(a) Reference image (b) Initial image



(c) Invariant space error (d) Image space error



(e) Translation velocity (f) Rotation velocity



(g) Translation error (h) Rotation error

Figure 3: *Intrinsics-free visual servoing.*

7 Conclusion

The intrinsics-free visual servoing approach proposed in the paper can be used to position a eye-in-hand camera with respect to a set of 3D straight lines even if the reference image has been learned with different camera intrinsic parameters. The same approach can be used with a zooming camera in order to enlarge the camera field of view and/or to bound the size of the objects in the image. We are currently improving the visual servoing approach with an effective strategy to control the zoom of the camera.

Acknowledgments

This work is a part of the National Research Program ROBEA funded by CNRS and INRIA.

Appendix

The entries of matrix $\mathbf{S}_h$ are the entries of vector $\boldsymbol{\sigma}$:

$$\sigma_1 = \sum_{i=1}^{n} \beta_i^2(\mathbf{h}_i) h_{1i}^2 \qquad \sigma_2 = \sum_{i=1}^{n} \beta_i^2(\mathbf{h}_i) h_{1i}\ h_{2i}$$
$$\sigma_3 = \sum_{i=1}^{n} \beta_i^2(\mathbf{h}_i) h_{2i}^2 \qquad \sigma_4 = \sum_{i=1}^{n} \beta_i^2(\mathbf{h}_i) h_{1i}\ h_{3i}$$
$$\sigma_5 = \sum_{i=1}^{n} \beta_i^2(\mathbf{h}_i) h_{2i}\ h_{3i} \qquad \sigma_6 = \sum_{i=1}^{n} \beta_i^2(\mathbf{h}_i) h_{3i}^2$$

The vector $\boldsymbol{\tau}$ is a function of $\boldsymbol{\sigma}$:

$$\tau_1 = \sqrt{\sigma_1},\ \tau_2 = \frac{1}{\sqrt{\sigma_1}}\,\sigma_2,\ \tau_3 = \frac{1}{\sqrt{\sigma_1}}\sqrt{\sigma_1\sigma_3 - \sigma_2^2}$$

$$\tau_4 = \frac{1}{\sqrt{\sigma_1}}\,\sigma_4,\ \tau_5 = \frac{1}{\sqrt{\sigma_1}}\sqrt{\frac{(\sigma_1\sigma_5 - \sigma_4\sigma_2)^2}{\sigma_1\sigma_3 - \sigma_2^2}}$$

$$\tau_6 = \frac{1}{\sqrt{\sigma_1}}\sqrt{\sigma_1\sigma_6 - \sigma_4^2 - \frac{(\sigma_1\sigma_5 - \sigma_4\sigma_2)^2}{\sigma_1\sigma_3 - \sigma_2^2}}$$

Conversely, $\boldsymbol{\sigma}$ can be written as a function of $\boldsymbol{\tau}$:

$$\begin{array}{lll} \sigma_1 = \tau_1^2 & , \sigma_2 = \tau_1\tau_2 & , \sigma_3 = \tau_2^2 + \tau_3^2 \\ \sigma_4 = \tau_1\tau_4 & , \sigma_5 = \tau_2\tau_4 + \tau_3\tau_5 & , \sigma_6 = \tau_4^2 + \tau_5^2 + \tau_6^2 \end{array}$$

References

[1] N. Andreff, B. Espiau, and R. Horaud. Visual servoing from lines. *IEEE Int. Conf. on Robotics and Automation*, vol. 3, pp. 2070-5, San Francisco, CA, April 2000.

[2] 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.

[3] G. Hager. A modular system for robust positioning using feedback from stereo vision. *IEEE Trans. on Robotics and Automation*, 13(4):582–595, Aug. 1997.

[4] 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.

[5] E. Malis. Vision-based control invariant to camera intrinsic parameters: stability analysis and path tracking. In *Int. Conf. on Robotics and Automation*, vol. 1, pp. 217–222, Washington, D.C., USA, May, 2002.

[6] E. Malis. Vision-based control using different cameras for learning the reference image and for servoing. In *IEEE/RSJ Int. Conf. on Intelligent Robots Systems*, volume 3, pages 1428–1433, Maui, Hawaii, Nov. 2001.

[7] P. Rives, J.J. Borrelly. Underwater Pipe Inspection Task using Visual Servoing Techniques. In *IEEE/RSJ Int. Conf. on Intelligent Robots Systems*, volume 1, pages 63–68, Grenoble, France, Sept. 1997.

[8] C. Samson, M. LeBorgne, B. Espiau. *Robot Control: the task function approach*, vol. 22 of Oxford Engineering Science Series. Clarendon Press, Oxford, UK, 1990.

[9] G. Urban, J.P.and Motyl and J. Gallice. Real-time visual servoing using controlled illumination. *Int. Journal of Robotics Research*, 13(1):93–100, February 1994.