2D 1/2 Visual Servoing

Ezio Malis, François Chaumette, Sylvie Boudet

INRIA

INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE

2D 1/2 Visual Servoing

E. Malis, F. Chaumette and S. Boudet

N ° 3387

mars 1998

THÈME 3

ISSN 0249-6399

2D 1/2 Visual Servoing

E. Malis, F. Chaumette and S. Boudet*

Thème 3 — Interaction homme-machine,
images, données, connaissances
Projet Vista

Rapport de recherche n3387 — mars 1998 — 71 pages

Abstract: In this paper, the problem of estimating the partial camera displacement from two images of a static object is studied. The classical approach to linearly estimate the motion parameters is based on the computation of the essential matrix. In this paper, we propose another approach, based on the estimation of an homography matrix related to a chosen plane of an unknown object. Simulations and experiments on a real scene show that this method gives a more robust reconstruction of the motion parameters, especially in the singular cases. The motion parameters are used to design a new vision-based control scheme, called 2D 1/2 visual servoing. Indeed, visual features and data extracted from the partial displacement allow us to design a decoupled control law controlling the six camera d.o.f. The robustness of our visual servoing scheme with respect to camera calibration errors is also analyzed: the necessary and sufficient conditions for local asymptotic stability are easily obtained. Then, thanks to the simple structure of the system, sufficient conditions for global asymptotic stability are established. Finally, experimental results with an eye-in-hand robotic system confirm the improvement in the stability and robustness of the 2D 1/2 visual servoing with respect to classical position-based and image-based visual servoings.

Key-words: Projective Geometry - Homography - Scaled Euclidean Reconstruction - Visual servoing - Stability Analysis

(Résumé : tsvp)

* EDF-DER, 6 Quai Watier, 78 401 Chatou, France. *E-mail: sylvie.boudet@edfgdf.fr*

Unité de recherche INRIA Rennes
IRISA, Campus universitaire de Beaulieu, 35042 RENNES Cedex (France)
Téléphone : 02 99 84 71 00 - International : +33 2 99 84 71 00
Télécopie : 02 99 84 71 71 - International : +33 2 99 84 71 71

Asservissement visuel 2D 1/2

Résumé : Dans ce papier, on étudie le problème de la reconstruction partielle du déplacement d'une caméra entre deux images d'un objet statique. Cette reconstruction est utilisée pour réaliser une nouvelle méthode d'asservissement visuel, appelée asservissement visuel 2D 1/2. Il faut souligner que le déplacement de la caméra et la structure de l'objet peuvent être estimées sans connaissance "a priori" sur le modèle 3D de l'objet, ce qui élargit grandement le domaine d'application de l'asservissement visuel. Puisque les taches robotiques doivent généralement être effectuées à la cadence vidéo, nous nous intéressons ici seulement aux algorithmes linéaires. L'approche classique pour l'estimation linéaire du déplacement de la caméra est basée sur le calcul de la matrice essentielle. Dans cet article, nous proposons une autre approche, basée sur l'estimation d'une matrice d'homographie associée a un plan de référence sur l'objet. Les résultats de simulation et ceux obtenus sur une scène réelle montrent que cette méthode produit une reconstruction plus robuste des paramètres de mouvement, spécialement dans les cas singuliers. Ensuite, nous décrivons l'asservissement visuel 2D 1/2. A partir des information visuelles et des données obtenues à partir du déplacement partiel de la caméra, on construit une loi de commande découplée en boucle fermée contrôlant le 6 d.d.l. de la caméra. On analyse ensuite la robustesse du système par rapport aux erreurs de calibration de la caméra: les conditions nécessaires et suffisantes pour la stabilité asymptotique locale peuvent facilement être obtenues. De plus, grâce à la structure du système, on obtient des conditions suffisantes pour la stabilité asymptotique globale. Enfin, les résultats expérimentaux, obtenus en utilisant un robot avec une caméra embarquée, confirment que l'asservissement visuel 2D 1/2 apporte une amélioration importante du domaine de convergence, de la stabilité, et de la robustesse par rapport aux approches classiques.

Mots-clé : Géometrie projective - Homographie - Reconstruction Euclidienne - Asservissement visuel - Analyse de stabilité

Contents

1 Introduction

Vision feedback control loop was introduced in order to increase the flexibility and the accuracy of robot systems [15, 18]. Consider for example the classical positioning task of an eye-in-hand system with respect to a target. After the image corresponding to the camera desired position has been learnt, and after the camera and/or the target has been moved, an error control vector can be extracted from the two views of the target. A zero error implies that the robot end-effector has reached its desired position with an accuracy regardless to calibration errors. However, these errors influence the way the system converges. In many cases, image features may get out of the camera field of view during the servoing, which thus leads to its failure. For this reason, it is important to study the visual servoing robustness with respect to calibration errors.

Vision-based robot control using an eye-in-hand system is classified into two groups [32, 15, 18]: *position-based* and *image-based* control systems. In a *position-based* control system, the input is computed in the 3D Cartesian space [34] (for this reason, this approach can be called *3D visual servoing*). The pose of the target with respect to the camera is estimated from image features corresponding to the perspective projection of the target in the image. Numerous methods exist to recover the pose of an object (see [6] for example). They are all based on the knowledge of a perfect geometric model of the object and necessitate a calibrated camera to obtain unbiased results. Even if a closed loop control is used, which makes possible the convergence of the system in presence of calibration errors, it is quite impossible to analyze the stability of the system. On the other hand, in an *image-based* control system, the input is computed in the 2D image space (for this reason, this approach can be called *2D visual servoing*) [10]. This local approach is known to be very robust with respect to camera and robot calibration errors [9]. However, its convergence is theoretically ensured only in a region (quite impossible to determine analytically) around the desired position. Except in simple cases, the analysis of the stability robustness with respect to calibration errors is again quite impossible, since the system is coupled and non-linear [9].

Contrarily to the previous approaches, we will see that it is possible to obtain analytical results using a new approach which combines the advantages of 2D and 3D visual servoings and avoids their respective drawbacks. This new

approach is called *2D 1/2 visual servoing* since the used input is expressed in part in the 3D Cartesian space and in part in the 2D image space [25]. More precisely, it is based on the estimation of the camera displacement (the rotation and the scaled translation of the camera) between the current and desired views of an object. It must be emphasized that, contrarily to the 3D visual servoing, the partial camera displacement estimation does not need any 3D model of the target, which increases the versatility and the application area of visual servoing. Since the camera rotation between the two views is computed at each iteration, the rotational and translational control loops can be decoupled. In order to control the three remaining camera d.o.f, we introduce the *extended image coordinates* of a reference point of the target. These ones are obtained from the classical normalized image coordinates by adding a third normalized z coordinate which is measured from the Euclidean reconstruction. It is interesting to note that the interaction matrix which links the time derivative of the extended image coordinates to the camera velocity screw has no singularity. This allows us to obtain the convergence of the positioning task in all the task space (i.e. for any initial camera position) if the camera intrinsic parameters are known. If the camera intrinsic parameters are not perfectly known, the estimated control vector can be analytically computed in function of camera calibration errors. Then, the necessary and sufficient conditions for the local asymptotic stability in presence of camera calibration errors are easily obtained. Moreover, thanks to the simple structure of the system, sufficient conditions for global asymptotic stability are presented. It will be shown experimentally that the 2D 1/2 visual servoing is robust also in presence of errors in the hand-eye calibration.

The paper is organized as follows. In Section 2, we propose a new linear algorithm for the homography matrix estimation and we compare it with the classical linear algorithms used to perform an Euclidean reconstruction. In Section 3, we show how to use the information extracted from the Euclidean partial reconstruction to design our 2D 1/2 visual servoing scheme. Its robustness with respect to camera calibration errors is analyzed in Section 4. The simulation and experimental results are given in Section 5 and Section 6 respectively.

2 Motion and Structure from Motion

The problem of recovering the camera displacement between two views is well known as the motion and structure from motion problem and it is, by its own nature, non-linear. Therefore, the classical approach to solve the problem is divided into two steps: using first a linear algorithm to estimate the motion parameters eliminating the non-linear constraints and then refining the estimation using a non-linear estimation algorithm [20]. In this paper, we point out our attention on the first stage, since it is very difficult to obtain a precise initial estimate and the second stage is in general very sensitive to the initial guess. Furthermore, a fast linear algorithm has to be used in robotic applications due to real-time constraints.

Different methods were proposed to linearly solve the motion and structure from motion problem. They are generally based on the computation of the fundamental matrix [24] if pixel image points coordinates are used, or of the essential matrix [21, 14] if normalized image points coordinates are used. This matrix has to satisfy the Huang-Faugeras conditions [17]. These non-linear constraints are imposed "a posteriori", after the linear estimation, when the partial camera displacement is computed. The partial camera displacement can be also estimated from an homography matrix related to a reference plane on the target [12, 35]. The homography matrix can be estimated jointly to the epipole using, for example, the algorithms presented in [3, 16] or after the epipole has been found [29] (if more than two views are available, see [19, 31]). It will be shown in this paper that the motion parameters estimation is more robust from an homography matrix than from the fundamental matrix, especially when the epipole is not defined in the image (for example, if the motion is a pure rotation or if the target is planar [22]). Since the classical algorithms estimate the homography matrix jointly to the epipole, the number of unknowns is not minimal. Furthermore, using the algorithm presented in [3], there are three epipolar configurations where it is impossible to extract the homography matrix. Since the epipole estimation is unnecessary for the homography estimation, we propose an algorithm similar to the one presented in [5], based on virtual parallax, for the direct estimation of the homography matrix. We will see that the method we propose is able to adequately deal with the singular cases. This is particularly important in visual servoing since

a positioning task is achieved when the camera displacement is null, which corresponds to a null pure rotation. Furthermore, since the target is a priori unknown, it may be planar (or all the observed target points may be close to a plane) which also corresponds to a particular case that is more correctly handled by our method.

2.1 The projective space as an extension of the Euclidean space

In this subsection we recall some bases of projective geometry (see [26] for more details). The projective space $\mathbb{P}^n$ can be seen geometrically as an "extension" of the Euclidean space $\mathbb{R}^n$ since it allows us to deal with the manipulation of geometrical objects at "infinity". Indeed, a point $\mathbf{P}$ in a n-dimensional Euclidean space, expressed in a given coordinate system by the vector $\begin{bmatrix} X_1 & X_2 & \cdots & X_n \end{bmatrix}^T \in \mathbb{R}^n$, can be represented in the projective space $\mathbb{P}^n$ using the homogeneous coordinates $\begin{bmatrix} X_1 & X_2 & \cdots & X_n & 1 \end{bmatrix}^T \in \mathbb{R}^{n+1}$. Furthermore, a direction $\begin{bmatrix} X_1 & X_2 & \cdots & X_n \end{bmatrix}^T$ in the Euclidean space, which corresponds to an ideal "point at infinity", can be represented using homogeneous coordinates by the projective point $\begin{bmatrix} X_1 & X_2 & \cdots & X_n & 0 \end{bmatrix}^T$. In the general case, a point in the projective space $\mathbb{P}^n$ is represented by a $n+1$ non-zero vector. Its elements are defined up to a scale factor and at least one of them is non-zero. The Euclidean points can be recovered from projective ones by dividing all the coordinates by the last projective coordinate, obviously only if it is non-zero.

An hyper-plane in the Euclidean space is the set of all points in $\mathbb{R}^n$ whose coordinates satisfy a linear equation. Using homogeneous coordinates, the hyper-plane can be represented in the projective space by a $n+1$ non-zero vector $\mathbf{q}$, defined up to a scale factor. A particular hyper-plane in the Euclidean space is the set of all "points at infinity", which correspond to an ideal "hyper-plane at infinity", and it can be represented by $\mathbf{q}_\infty = \begin{bmatrix} 0 & 0 & \cdots & 0 & 1 \end{bmatrix}^T$. All points in $\mathbb{P}^n$ lying in a hyper-plane satisfy the linear equation $\mathbf{q}^T\mathbf{P} = 0$. This equation presents a symmetry between the hyper-plane coordinates $\mathbf{q}$ and the point coordinates $\mathbf{P}$ since $\mathbf{q}^T\mathbf{P} = \mathbf{P}^T\mathbf{q}$. Indeed, the space of all hyper-planes can be considered to be another projective space called the dual of the original

space $\mathbb{P}^n$. An extremely important *duality principle* follows from this symmetry [26]:

Duality Principle: *For any projective results established using points and hyper-planes, a symmetrical result holds in which the roles of hyper-planes and points are interchanged.*

Any full-rank $(m+1, n+1)$ matrix can be used to perform a projective mapping between two projective spaces $\mathbb{P}^m$ and $\mathbb{P}^n$. An invertible projective mapping of $\mathbb{P}^n$ onto itself is called a collineation and corresponds to a change of projective coordinates obtained using a $(n+1, n+1)$ matrix. If we compare a projective and an Euclidean change of coordinates, we can see that the projective space is less constrained than the Euclidean space. Indeed, an Euclidean change of coordinates, performed by using an homogeneous transformation matrix, must map the hyper-plane at infinity $\mathbf{q}_\infty$ into itself.

For this reason, projective geometry is a powerful tool which allows computational simplifications. However, it must be emphasized that robotics tasks take place in the Euclidean space and then, after computation, the robot control is unavoidably designed in the Euclidean space. The links between projective and Euclidean space, made using homogeneous coordinates, are well known in robotics since a Cartesian change of coordinates can be represented by homogeneous transformation matrices. Furthermore, homogeneous transformation matrices have been used for robot modeling and control [8]. Let us recall briefly some bases on the homogeneous transformation matrices.

Let $\mathcal{F}^*$ be the world reference frame in the Cartesian space. All the geometrical entities expressed in this frame will be designed with the over-script *. The homogeneous transformation matrix between $\mathcal{F}^*$ and a second frame $\mathcal{F}$ is:

$$\mathbf{T}^* = \begin{bmatrix} \mathbf{R} & \mathbf{t} \\ 0 & 1 \end{bmatrix} \tag{1}$$

where $\mathbf{R}$ and $\mathbf{t}$ are respectively the rotation matrix and the translation vector between $\mathcal{F}^*$ and $\mathcal{F}$. The inverse transformation is:

$$(\mathbf{T}^*)^{-1} = \begin{bmatrix} \mathbf{R}^T & -\mathbf{R}^T\mathbf{t} \\ 0 & 1 \end{bmatrix} \tag{2}$$

A point $\mathbf{P}^*$ in the Cartesian space, expressed in frame $\mathcal{F}^*$, is represented using the homogeneous coordinates $\begin{bmatrix} X^* & Y^* & Z^* & W^* \end{bmatrix}^T \in \mathbb{R}^4$. If the fourth

homogeneous coordinate is $W^* = 0$ then the point is "at infinity" (i.e. it represents a direction in the Cartesian space). Otherwise, we can set $W^* = 1$, without loss of generality, to represent a finite point. This point is expressed in the new frame $\mathcal{F}$ as follows:

$$\mathbf{P} = \mathbf{T}^* \mathbf{P}^* \tag{3}$$

A plane in the Cartesian space, expressed in frame $\mathcal{F}^*$, is characterized by its normal vector $\mathbf{n}^*$ and its distance d^* from the origin C^* of the reference frame. Using homogeneous coordinates, it is represented by the vector $\mathbf{q}^* = \left[\begin{array}{cc} \mathbf{n}^* & -d^* \end{array} \right]^T$. The same plane, denoted $\mathbf{q} = \left[\begin{array}{cc} \mathbf{n} & -d \end{array} \right]^T$ in frame $\mathcal{F}$, is expressed as follows:

$$\mathbf{q} = (\mathbf{T}^*)^{-T} \mathbf{q}^* \qquad \Rightarrow \qquad \left[\begin{array}{c} \mathbf{n} \\ -d \end{array} \right] = \left[\begin{array}{c} \mathbf{R}\mathbf{n}^* \\ -d^* - \mathbf{t}^T \mathbf{R}\mathbf{n}^* \end{array} \right] \tag{4}$$

where matrix $(\mathbf{T}^*)^{-T}$ is called the *dual* of $\mathbf{T}^*$.

2.1.1 Perspective projection

Let the *center of projection* C^* be the origin of the world coordinate system. The projection plane Π is parallel to the $(\overrightarrow{x}, \overrightarrow{y})$ plane and its distance f from C^* is called the *focal length*.

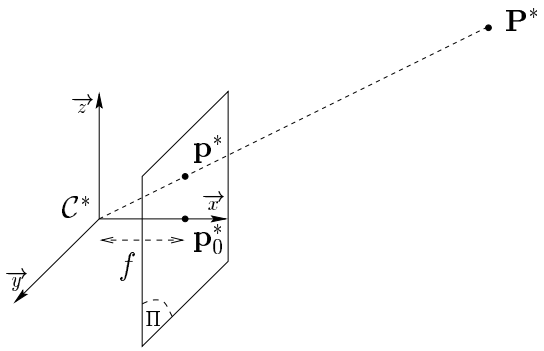


Figure 1: Perspective projection

The perspective projection of a point $\mathbf{P}^* \in \mathbb{P}^3$ to a point $\mathbf{p}^* \in \mathbb{P}^2$ onto the projection plane Π (see Figure 1) can be written as:

$$\lambda^* \mathbf{p}^* = \mathbf{Q}^* \mathbf{P}^* \tag{5}$$

where $\mathbf{Q}^* = \left[\begin{array}{cc} \mathbf{I}_3 & \mathbf{0} \end{array} \right]$ is the (3×4) projection matrix and λ^* is a non-zero scalar. In the Euclidean space we have $\lambda^* = Z^*/f$. The intersection $\mathbf{p}_0^*$ of the $\overrightarrow{z}$ axis and the image plane (see Figure 1) is called the *principal point*.

As seen previously, a change of frame is represented by an homogeneous matrix. Then, the point $\mathbf{P}^*$ is projected in a new coordinate system as:

$$\lambda \mathbf{p} = \mathbf{Q} \mathbf{P}^* \tag{6}$$

where $\mathbf{Q} = \mathbf{Q}^* \mathbf{T}^* = \left[\begin{array}{cc} \mathbf{R} & \mathbf{t} \end{array} \right]$ is the (3×4) projection matrix expressed in the new frame and λ is a non-zero scalar. In the Euclidean space, we have $\lambda = Z/f$.

2.1.2 Camera model

The camera, described using the pinhole model, performs a perspective projection of a point $\mathbf{P} \in \mathbb{P}^3$ to an image point $\mathbf{m} \in \mathbb{P}^2$ with pixel homogeneous coordinates $\mathbf{m} = \left[\begin{array}{ccc} u & v & 1 \end{array} \right]^T$ (see [33] for different camera models). The camera is supposed to be without lens distortion (see [36] for camera model with distortion), then the relationship between the coordinates of $\mathbf{P}$ and $\mathbf{m}$ is linear. As seen previously, the point $\mathbf{P}$ is projected in a point with normalized homogeneous coordinates $\mathbf{p} = \left[\begin{array}{ccc} x & y & 1 \end{array} \right]^T$. The projective transformation between the normalized coordinates and the pixel coordinates of an image point is known to be:

$$\mathbf{m} = \mathbf{A} \mathbf{p} \tag{7}$$

where $\mathbf{A}$ is the intrinsic parameters matrix which performs the change of projective coordinates in $\mathbb{P}^2$:

$$\mathbf{A} = \left[\begin{array}{ccc} f k_u & f k_u \cot(\phi) & u_0 \\ 0 & f k_v / \sin(\phi) & v_0 \\ 0 & 0 & 1 \end{array} \right] = \left[\begin{array}{ccc} \alpha_u & \alpha_{uv} & u_0 \\ 0 & \alpha_v & v_0 \\ 0 & 0 & 1 \end{array} \right] \tag{8}$$

where u_0 and v_0 are the pixels coordinates of the principal point, k_u and k_v are the scaling factors along the $\overrightarrow{x}$ and $\overrightarrow{y}$ axes (in pixels/meters), and ϕ is the angle between these axes. Then, applying the change of projective coordinates, from equation (6), we obtain:

$$\lambda \mathbf{m} = \mathbf{AQP} \tag{9}$$

If the camera intrinsic parameters are perfectly known, we can work indistinctly in the normalized nor pixel coordinate systems. It must be emphasized that a normalization of data is needed when working with pixels coordinates [14]. Even if the camera is not perfectly calibrated, using normalized coordinates gives more robust results.

2.2 Projective reconstruction from two images of a static object

2.2.1 The relationship between two perspective projections

Consider two perspective projections of a static object (see Figure 2). In a visual servoing positioning task, the goal is to design a control law such that the current camera pose ($\mathcal{F}$) reaches a desired pose ($\mathcal{F}^*$). In order to link the two perspective projections of the object, a finite point $\mathbf{P}^*$ can be computed as follows:

$$\mathbf{P}^* = \lambda^* \mathbf{Q}^{*+} \mathbf{p}^* + \mathbf{Q}^{*\perp} \tag{10}$$

where $\mathbf{Q}^{*+} = \mathbf{Q}^{*T}(\mathbf{Q}^*\mathbf{Q}^{*T})^{-1} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{0} \end{bmatrix}^T$ is the pseudo-inverse of matrix $\mathbf{Q}^*$ and $\mathbf{Q}^{*\perp} = \begin{bmatrix} 0 & 0 & 0 & W^* \end{bmatrix}^T$ is the vector of its null space (i.e. $\mathbf{Q}^*\mathbf{Q}^{*\perp} = 0$) containing the homogeneous coordinate W^*. The Euclidean distance $d(\mathbf{P}^*, \mathbf{q}^*)$ between the point $\mathbf{P}^*$ and a reference plane π (see Figure 2), described by the vector $\mathbf{q}^* = \begin{bmatrix} \mathbf{n}^* & -d^* \end{bmatrix}^T$, can be written as: $d(\mathbf{P}^*, \mathbf{q}^*) = \mathbf{q}^{*T}\mathbf{P}^*$. Then the homogeneous coordinate W^* can be expressed as $W^* = \lambda^* \mathbf{n}^{*T}\mathbf{p}^*/d^* - d(\mathbf{P}^*, \mathbf{q}^*)/d^*$. Equation (10) can thus be written as:

$$\mathbf{P}^* = \lambda^* \mathbf{Q}_\pi^{*+} \mathbf{p}^* + \mathbf{Q}_\pi^{*\perp} \tag{11}$$

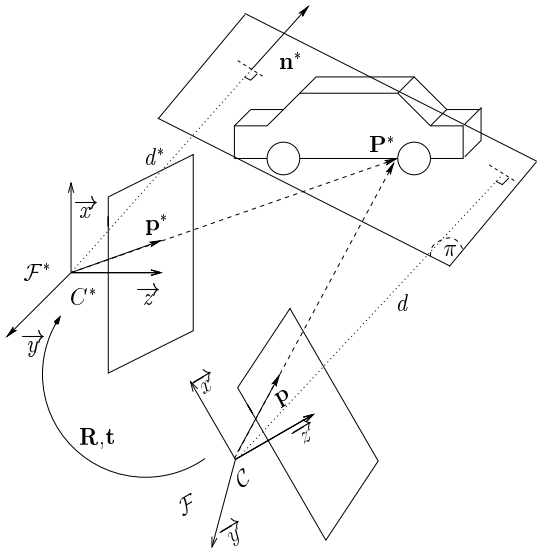


Figure 2: Camera displacement

where $\mathbf{Q}_\pi^{*+} = \begin{bmatrix} \mathbf{I}_3 & \mathbf{n}^*/d^* \end{bmatrix}^T$ and $\mathbf{Q}_\pi^{*\perp} = \begin{bmatrix} \mathbf{0} & \epsilon \end{bmatrix}^T$ (with $\epsilon = -d(\mathbf{P}^*, \mathbf{q}^*)/d^*$). Using equation (6), the relationship between the projected point in the two images can be written as:

$$\lambda \mathbf{p} = \lambda^* \mathbf{Q}\mathbf{Q}_\pi^{*+}\mathbf{p}^* + \mathbf{Q}\mathbf{Q}_\pi^{*\perp} = \lambda^* \mathbf{H}_e \mathbf{p}^* + \epsilon \mathbf{t} \tag{12}$$

where $\mathbf{H}_e = \mathbf{Q}\mathbf{Q}_\pi^{*+} = \mathbf{R} + \mathbf{t}\mathbf{n}^{*T}/d^*$ is the homography matrix related to the reference plane. If the reference plane is $\mathbf{q}_\infty$, we have $\mathbf{n}^*/d^* = 0$ and $\epsilon = 1$, then equation (12) can be written as:

$$\lambda \mathbf{p} = \lambda^* \mathbf{H}_{e_\infty} \mathbf{p}^* + \mathbf{t} \tag{13}$$

where $\mathbf{H}_{e_\infty} = \mathbf{R}$ is the homography matrix related to the plane at infinity (since it maps points lying in the plane at infinity between the two projections).

2.2.2 The epipolar constraint

Equation (12) can be written in the retinal coordinate system, using equation (7), as follows:

$$\lambda \mathbf{m} = \lambda^* \mathbf{A}\mathbf{H}_e\mathbf{A}^{-1}\mathbf{m}^* + \mathbf{A}\mathbf{t} = \lambda^* \mathbf{H}_p \mathbf{m}^* + \epsilon \mathbf{e} \tag{14}$$

where $\mathbf{e} = \mathbf{At}$ is the epipole in the second image and $\mathbf{H}_p = \mathbf{A}\mathbf{H}_e\mathbf{A}^{-1}$ is the projective homography expressed in the image coordinate system. Consider now three 3D points $\mathbf{P}_i$ on the object. These points are chosen such that they are not collinear in the images. It is well known that the resulting image point with projective coordinates $\mathbf{m}_i$ in $\mathcal{F}$, is related to the corresponding image point with projective coordinates $\mathbf{m}_i^*$ in $\mathcal{F}^*$, by an homography such that [12]:

$$\alpha_i \mathbf{m}_i = \mathbf{H}_p \mathbf{m}_i^* \qquad \{i = 1, 2, 3\} \tag{15}$$

where $\mathbf{H}_p$ is a (3×3) projective homography matrix and $\alpha_i = \lambda_i / \lambda_i^*$ is a non-zero scalar. Indeed, equation (15) can be obtained from equation (14) since $\epsilon = -d(\mathbf{P}^*, \mathbf{q}^*)/d^* = 0$ for all points lying on plane π defined by the three selected points. Let us remark that $\mathbf{H}_p$ is defined up to a scalar factor, therefore one of the elements of $\mathbf{H}_p$ can be set to 1 without loss of generality. In the case of a planar object, as already stated, equation (15) is available for each feature point, thus, if more than three points are available, the homography matrix can be estimated solving a linear homogeneous system in the 8 unknown elements of $\mathbf{H}_p$.

In the general case, the structure of the target is not known. If a target point $\mathbf{P}_j$ does not belong to π, the line $(\mathbf{C}^*\mathbf{P}_j)$ and the plane π intersect in the virtual 3D point $\mathbf{P}_j'$ (see Figure 3).

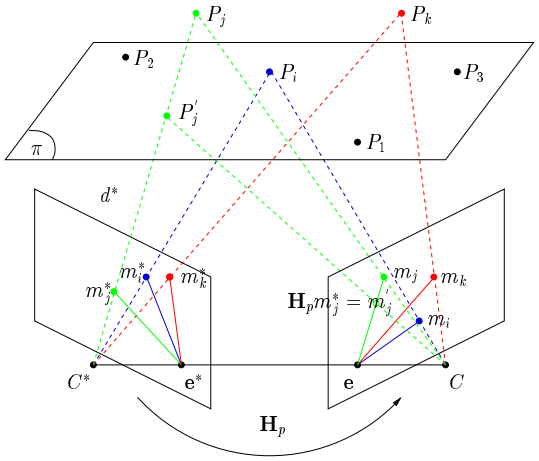


Figure 3: Epipolar Geometry and Virtual Parallax

$\mathbf{P}_j^{'}$ and $\mathbf{P}_j$ project on the same point $\mathbf{m}_j^*$ in the first image and on two different points ($\mathbf{m}_j$ and the virtual point $\mathbf{m}_j^{'} = \mathbf{H}_p\mathbf{m}_j^*$) in the second image (parallax effect). The line defined by $\mathbf{m}_j$ and $\mathbf{m}_j^{'}$ is represented, using projective coordinates, by $\mathbf{l}_j = \mathbf{m}_j \wedge \mathbf{m}_j^{'}$ [11]:

$$\mathbf{l}_j = \mathbf{m}_j \wedge \mathbf{H}_p\mathbf{m}_j^* \tag{16}$$

The four points $\mathbf{C}$, $\mathbf{C}^*$, $\mathbf{P}_j$ and $\mathbf{P}_j^{'}$ are coplanar, therefore the points $\mathbf{m}_j$, $\mathbf{m}_j^{'}$ and the epipole $\mathbf{e}$ (i.e. the projection of C^* in the second image), are collinear:

$$\mathbf{l}_j^T\mathbf{e}_j = 0 \tag{17}$$

This equation can be written, using equation (16), as:

$$\mathbf{l}_j^T\mathbf{e}_j = \mathbf{m}_j^T\mathbf{F}\mathbf{m}_j^* = 0 \tag{18}$$

where $\mathbf{F} = [\mathbf{e}]_\times\mathbf{H}_p$ is the fundamental matrix ($[\mathbf{e}]_\times$ is the crossproduct matrix associated to the vector $\mathbf{e}$). In the general case, $\mathbf{F}$ is rank 2, which implies a non-linear constraint on the elements of $\mathbf{F}$ [24].

2.2.3 Fundamental matrix estimation

The classical approach to compute the epipolar geometry is the eight point algorithm [21, 14]. Indeed, it is possible to simplify the problem performing only a linear estimation. Equation (18) is true for each pair of points ($\mathbf{m}_j$, $\mathbf{m}_j^*$), then, if n pairs are available, it is possible to obtain a linear system:

$$\mathbf{C}_f\mathbf{f} = 0 \tag{19}$$

where $\mathbf{C}_f$ is a $(n\times 9)$ coefficient matrix and $\mathbf{f} = [\ f_{11}\ \ f_{12}\ \ f_{13}\ \ f_{21}\ \ f_{22}\ \ f_{23}\ \ f_{31}\ \ f_{32}\ \ f_{33}\ ]^T$ are the 9 unknown elements of $\mathbf{F}$. System (19) is homogeneous and, since $\mathbf{F}$ is defined up to a scale factor, a minimum of 8 pairs of points are necessary to solve (19). In presence of noise, the linearized estimation problem can be written:

$$\min_{\mathbf{f}} \|\mathbf{C}_f\mathbf{f}\| \quad \text{subject to} \quad \|\mathbf{f}\| = 1 \tag{20}$$

The solution of this problem is obtained by performing the Singular Values Decomposition of the coefficient matrix $\mathbf{C}_f^T\mathbf{C}_f = \mathbf{V}\mathbf{S}\mathbf{V}^T$ of dimension (9×9). The solution $\overline{\mathbf{f}}$ of the system is the column of $\mathbf{V}$ corresponding to the minimal singular value of $\mathbf{S}$ (0 in absence of noise).

Let us remark that, if the epipole is undefined in the image (for example if the motion is a pure rotation or if the target is planar [22]), the fundamental matrix is also undefined, which implies an unstable estimation near these particular cases. We will see in Section 2.2.5 that the method we propose is able to adequately deal with these problems.

The linear algorithm does not take into account the rank 2 constraint on the fundamental matrix. This constraint is generally taken into account using a non-linear algorithm [7, 24]. Since the aim of this paper is to perform linear algorithms for real-time vision-based robotic tasks, the non-linear criteria are not detailed. However, we will compare our results with those obtained using the non-linear method described in [7]. We will see later that, if the camera intrinsic parameters are known, the rank 2 constraint can be taken into account a posteriori when computing the essential matrix.

2.2.4 Epipole and homography matrix estimation

To simplify the computation of matrix $\mathbf{F}$, Boufama et al. [3] perform a change of projective coordinates using 4 points in each image. These points are chosen such that not any three of them are collinear in the images. Let $\mathbf{M} = \mathbf{M}(\mathbf{m}_1, \mathbf{m}_2, \mathbf{m}_3, \mathbf{m}_4)$ and $\mathbf{M}^* = \mathbf{M}^*(\mathbf{m}_1^*, \mathbf{m}_2^*, \mathbf{m}_3^*, \mathbf{m}_4^*)$ be the matrices of change of coordinates. The image points $\tilde{\mathbf{m}}_j = \left[\begin{array}{ccc} \tilde{u}_j & \tilde{v}_j & \tilde{w}_j \end{array}\right]^T$ and $\tilde{\mathbf{m}}_j^* = \left[\begin{array}{ccc} \tilde{u}_j^* & \tilde{v}_j^* & \tilde{w}_j^* \end{array}\right]^T$ in the new coordinate system can be computed by $\tilde{\mathbf{m}}_j = \mathbf{M}^{-1}\mathbf{m}_j$ and $\tilde{\mathbf{m}}_j^* = \mathbf{M}^{*-1}\mathbf{m}_j^*$. Using the first three points, the matrices of change of coordinates can be written as:

$$
\begin{aligned}
\mathbf{M} &= \left[\begin{array}{ccc} \mathbf{m}_1 & \mathbf{m}_2 & \mathbf{m}_3 \end{array}\right] \begin{bmatrix} \lambda_1 & 0 & 0 \\ 0 & \lambda_2 & 0 \\ 0 & 0 & \lambda_3 \end{bmatrix} \\
\mathbf{M}^* &= \left[\begin{array}{ccc} \mathbf{m}_1^* & \mathbf{m}*_2 & \mathbf{m}_3^* \end{array}\right] \begin{bmatrix} \lambda_1^* & 0 & 0 \\ 0 & \lambda_2^* & 0 \\ 0 & 0 & \lambda_3^* \end{bmatrix}
\end{aligned} \tag{21}
$$

where the unknown scalars λ_i and λ_i^* can be computed, up to a scalar factor, using the fourth point:

$$\begin{aligned}\begin{bmatrix}\lambda_1\\ \lambda_2\\ \lambda_3\end{bmatrix} &= \lambda_4 \begin{bmatrix}\mathbf{m}_1 & \mathbf{m}_2 & \mathbf{m}_3\end{bmatrix}^{-1}\mathbf{m}_4\\ \begin{bmatrix}\lambda_1^*\\ \lambda_2^*\\ \lambda_3^*\end{bmatrix} &= \lambda_4^* \begin{bmatrix}\mathbf{m}_1^* & \mathbf{m}_2^* & \mathbf{m}_3^*\end{bmatrix}^{-1}\mathbf{m}_4^*\end{aligned} \tag{22}$$

Choosing $\begin{bmatrix}\tilde{\mathbf{m}}_1 & \tilde{\mathbf{m}}_2 & \tilde{\mathbf{m}}_3\end{bmatrix} = \begin{bmatrix}\tilde{\mathbf{m}}_1^* & \tilde{\mathbf{m}}_2^* & \tilde{\mathbf{m}}_3^*\end{bmatrix} = \mathbf{I}_3$ for the first three points, the homography matrix $\tilde{\mathbf{H}}_p$, related to the plane π defined by these three points, is diagonal when expressed in the new coordinate system:

$$\tilde{\mathbf{H}}_p = \mathbf{M}^{-1}\mathbf{H}_p\mathbf{M}^* = diag(\tilde{h}_u, \tilde{h}_v, \tilde{h}_w) \tag{23}$$

Then, the fundamental matrix can be written in the new coordinate system as:

$$\tilde{\mathbf{F}} = [\tilde{\mathbf{e}}]_\times \tilde{\mathbf{H}}_p \tag{24}$$

where $\tilde{\mathbf{e}} = \mathbf{M}^{-1}\mathbf{e}$ is the epipole in the new coordinate system. Equation (24) is polynomial of degree two in four unknowns (i.e., two unknowns for the epipole and two unknowns for the diagonal homography matrix since they are defined up to a scale factor). It is possible to write equation (24) as [3]:

$$\mathbf{C}_{\tilde{f}}\tilde{\mathbf{f}} = 0 \tag{25}$$

where $\tilde{\mathbf{f}} = \begin{bmatrix}\tilde{e}_x\tilde{h}_w & \tilde{e}_x\tilde{h}_v & \tilde{e}_y\tilde{h}_u & \tilde{e}_y\tilde{h}_w & \tilde{e}_z\tilde{h}_v & \tilde{e}_z\tilde{h}_u\end{bmatrix}^T$. The new system is a linear homogeneous system in 6 unknowns. Then at least five points not belonging to π are needed to solve linearly the problem. If m ($m \geq 5$) points are available, the matrix $\mathbf{C}_{\tilde{f}}$ of coefficients is of dimension ($m \times 6$). The problem is solved by performing the Singular Values Decomposition of the matrix $\mathbf{C}_{\tilde{f}}^T\mathbf{C}_{\tilde{f}} = \mathbf{V}\mathbf{S}\mathbf{V}^T$ (of dimension 6×6). Once again, the solution $\bar{\tilde{\mathbf{f}}}$ of the system is the column of $\mathbf{V}$ corresponding to the minimal singular value of $\mathbf{S}$ (0 in absence of noise). After the vector $\bar{\tilde{\mathbf{f}}}$ is obtained, the original unknowns can be estimated. Unfortunately, there are three singular cases where

the homography matrix cannot be estimated with this algorithm. Indeed, if $\tilde{\mathbf{e}} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}^T$, only $\tilde{h}_v/\tilde{h}_w$ is known; if $\tilde{\mathbf{e}} = \begin{bmatrix} 0 & 1 & 0 \end{bmatrix}^T$, only $\tilde{h}_u/\tilde{h}_w$ is known; and, if $\tilde{\mathbf{e}} = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$, only $\tilde{h}_u/\tilde{h}_v$ is known. In these cases, in absence of noise, the estimation of the homography matrix is impossible with this algorithm. If these particular cases can be detected, another algorithm can be used. However, in presence of noise, the estimation will not be accurate since zero values estimation is very sensitive to numerical errors.

The main advantage of this algorithm is that, even degenerating in the singular cases, it can provide the homography matrix which is never undefined, contrarily to the fundamental matrix. Finally, in this algorithm, the homography matrix depends on the epipole estimation, and the number of unknowns is not minimal. For these reasons, we propose in the next section a method that determines directly the homography matrix without estimating the epipole.

2.2.5 Projective homography estimation

Our approach, similar to the one proposed in [5], is based on the constraint that all the epipolar lines meet in the epipole. Hence, for each set of three epipolar lines (16), we have:

$$\begin{vmatrix} \mathbf{l}_j & \mathbf{l}_k & \mathbf{l}_l \end{vmatrix} = \begin{vmatrix} \mathbf{m}_j \wedge \mathbf{H}_p\mathbf{m}_j^* & \mathbf{m}_k \wedge \mathbf{H}_p\mathbf{m}_k^* & \mathbf{m}_l \wedge \mathbf{H}_p\mathbf{m}_l^* \end{vmatrix} = 0 \qquad (26)$$

However, equation (26) is non-linear with respect to the elements of the projective homography matrix. In order to simplify the computation of $\mathbf{H}_p$, a change of projective coordinates is performed.

Linear method without enforcing the non-linear constraints Couapel [5] performs a change of projective coordinates using 4 points in each image as done by Boufama [3]. However, the fourth point is the center of gravity of the first 3 points which is not an optimal choice since the center of gravity of the projected points is in general not equal to the projection of the center of gravity of the 3D points. Choosing again $\begin{bmatrix} \tilde{\mathbf{m}}_1 & \tilde{\mathbf{m}}_2 & \tilde{\mathbf{m}}_3 \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{m}}_1^* & \tilde{\mathbf{m}}_2^* & \tilde{\mathbf{m}}_3^* \end{bmatrix} = \mathbf{I}_3$, the homography matrix $\tilde{\mathbf{H}}_p$ in the new coordinates system is diagonal: $\tilde{\mathbf{H}}_p = \mathbf{M}^{-1}\mathbf{H}_p\mathbf{M}^* = diag(\tilde{h}_u, \tilde{h}_v, \tilde{h}_w)$. Equation (26) can be written in the new

coordinate system as:

$$\left| \begin{array}{ccc} \tilde{\mathbf{m}}_j \wedge \tilde{\mathbf{H}}_p \tilde{\mathbf{m}}_j^* & \tilde{\mathbf{m}}_k \wedge \tilde{\mathbf{H}}_p \tilde{\mathbf{m}}_k^* & \tilde{\mathbf{m}}_l \wedge \tilde{\mathbf{H}}_p \tilde{\mathbf{m}}_l^* \end{array} \right| = 0 \tag{27}$$

After computation, (27) can be written under the form:

$$\mathbf{A}_{\tilde{h}} \mathbf{y} = \mathbf{b}_{\tilde{h}} \tag{28}$$

where $\mathbf{A}_{\tilde{h}}$ is the coefficient matrix, $\mathbf{b}_{\tilde{h}}$ is a vector and, after setting $\tilde{h}_w = 1$:

$$\mathbf{y} = \left[\begin{array}{ccccc} \tilde{h}_u & \tilde{h}_v & \tilde{h}_u/\tilde{h}_v & \tilde{h}_v/\tilde{h}_u & 1/\tilde{h}_v \quad 1/\tilde{h}_u \end{array} \right]^T \tag{29}$$

The solution of the linear system is found as $\overline{\mathbf{y}} = \mathbf{A}_{\tilde{h}}^+ \mathbf{b}_{\tilde{h}}$ (where $\mathbf{A}_{\tilde{h}}^+$ is the pseudo-inverse of $\mathbf{A}_{\tilde{h}}$). Then, Couapel obtains the unknowns $\tilde{h}_u = \overline{y}_1$, $\tilde{h}_v = \overline{y}_2$ and the following controls (called *errors of coherence*) on the quality of the solution $\overline{y}_1 \overline{y}_6 = \overline{y}_2 \overline{y}_5 = \overline{y}_3 \overline{y}_4 = 1$. Of course, these non-linear constraints are not necessarily satisfied with noisy measurements.

Linear method with enforcing the non-linear constraints In contrast with Boufama [3] and Couapel [5], the change of coordinates matrices $\mathbf{M}$ and $\mathbf{M}^*$ are constructed using only three reference points. In this case, the transformation matrices are $\mathbf{M} = \left[\begin{array}{ccc} \mathbf{m}_1 & \mathbf{m}_2 & \mathbf{m}_3 \end{array} \right]$ and $\mathbf{M}^* = \left[\begin{array}{ccc} \mathbf{m}_1^* & \mathbf{m}_2^* & \mathbf{m}_3^* \end{array} \right]$ (let us remember that any (3×3) matrix can be used for change of coordinates in $\mathbb{P}^2$). Choosing again $\left[\begin{array}{ccc} \tilde{\mathbf{m}}_1 & \tilde{\mathbf{m}}_2 & \tilde{\mathbf{m}}_3 \end{array} \right] = \left[\begin{array}{ccc} \tilde{\mathbf{m}}_1^* & \tilde{\mathbf{m}}_2^* & \tilde{\mathbf{m}}_3^* \end{array} \right] = \mathbf{I}_3$, the homography matrix $\tilde{\mathbf{H}}_p$ in the new coordinates system is diagonal: $\tilde{\mathbf{H}}_p = \mathbf{M}^{-1} \mathbf{H}_p \mathbf{M}^* = diag(\tilde{h}_u, \tilde{h}_v, \tilde{h}_w)$. Let us remark that the change of coordinates normalizes the data, which is very important to obtain a robust estimation in the projective domain [14]. Equation (26) is homogeneous and polynomial of degree three. Contrarily to (24), equation (27) does not depend on the epipole and contains only three unknowns. After computation, (27) can be written under the form:

$$\mathbf{C}_{\tilde{h}} \mathbf{x} = 0 \tag{30}$$

where the elements of the coefficient matrix $\mathbf{C}_{\tilde{h}}$ are given in Appendix A and:

$$\mathbf{x} = \left[\begin{array}{ccccccc} \tilde{h}_u^2 \tilde{h}_v & \tilde{h}_v^2 \tilde{h}_u & \tilde{h}_u^2 \tilde{h}_w & \tilde{h}_v^2 \tilde{h}_w & \tilde{h}_w^2 \tilde{h}_u & \tilde{h}_w^2 \tilde{h}_v & \tilde{h}_u \tilde{h}_v \tilde{h}_w \end{array} \right]^T \tag{31}$$

There are $n!/(6(n-3)!)$ possibilities to choose three different epipolar lines in a set of n epipolar lines (one line for each point in the image). We thus obtain $m = n!/(6(n-3)!)$ equations and seven unknowns. At least eight points (three reference points and five supplementary points) are thus needed to solve the problem, exactly as in the previous algorithms. The matrix $\mathbf{C}_{\tilde{h}}$ of coefficients is of dimension $(m \times 7)$. The problem is now solved by performing the Singular Values Decomposition of the matrix $\mathbf{C}_{\tilde{h}}^T \mathbf{C}_{\tilde{h}} = \mathbf{V}\mathbf{S}\mathbf{V}^T$ (of dimension 7×7). Finally, the original unknowns can be computed by solving the following linear homogeneous system:

$$\begin{bmatrix} -\overline{x}_2 & \overline{x}_5 & -\overline{x}_7 & \overline{x}_7 & -\overline{x}_4 & \overline{x}_4 & -\overline{x}_6 & \overline{x}_6 & 0 \\ \overline{x}_1 & 0 & \overline{x}_3 & 0 & \overline{x}_7 & 0 & \overline{x}_5 & 0 & -\overline{x}_6 \\ 0 & -\overline{x}_3 & 0 & -\overline{x}_1 & 0 & -\overline{x}_2 & 0 & -\overline{x}_7 & \overline{x}_3 \end{bmatrix}^T \begin{bmatrix} \tilde{h}_u \\ \tilde{h}_v \\ \tilde{h}_w \end{bmatrix} = 0 \quad (32)$$

Contrarily to the algorithm described in the previous section, the homography matrix can be more accurately estimated because the dimension of the problem is reduced (the epipole estimation is separated from the homography estimation). We will see in Sections 5 and 6 that this method provides indeed more accurate results.

It must also be emphasized that, even if the singular cases for epipolar geometry are the same that for the fundamental matrix, the homography matrix is always defined. Consider for example the case of a planar target. Any image point can be chosen as epipole. Then, any vector $\mathbf{f}$ is a solution of system (19). In absence of noise, this singular case can be detected and another algorithm can be used. However, in presence of noise, any solution may be taken as the good solution. On the other hand, since the object is planar, there exists only one homography matrix related to the chosen three reference points. The obtained solution will thus be an approximation of the real homography matrix. Considering this particularly is important since the structure of the observed object is a priori unknown.

Consider now the case of a pure rotation. The solution of system (19) is $\mathbf{f} = 0$. Once again, in absence of noise, this singular case can be detected and another algorithm can be used. However, in presence of noise, the fundamental matrix is estimated by imposing the constraint $\|\mathbf{f}\| = 1$ since $\mathbf{f}$ is obtained as a column of an orthonormal matrix. Without a correct detection of the singularity, this can lead to very unstable results since the right solution would

be $\|\mathbf{f}\| = 0$. On the other hand, the solutions of systems (30) and (32) always satisfy the constraints $\|\mathbf{x}\| = 1$ and $\|\tilde{\mathbf{h}}\| = 1$ respectively. These constraints, imposed when performing the SVD of the coefficient matrices, have to be ensured even in the singular cases. Then, in presence of noise, the motion estimation around these singular configurations will be more accurate when performed from the homography matrix than from the fundamental matrix. This is particularly important in visual servoing since a positioning task is achieved when the camera displacement is null, which correspond to a null pure rotation.

2.3 From projective to scaled Euclidean reconstruction

The matrix $\mathbf{F}$ is estimated using pixel image coordinates. When the camera is calibrated and the normalized image coordinates are used, the two images are related by the essential matrix $\mathbf{E}$ [21]. Similarly, the corresponding matrix of $\mathbf{H}_p$ in the calibrated domain can be called the Euclidean homography matrix $\mathbf{H}_e$.

2.3.1 Known camera calibration

The Essential matrix The essential matrix is written as a function of the calibration parameters and of the fundamental matrix as follows:

$$\mathbf{E} = \mathbf{A}^T \mathbf{F} \mathbf{A} \tag{33}$$

where $\mathbf{A}$ is given by (8). The essential matrix estimation problem is non-linear since $\mathbf{E}$ has to satisfy the Huang-Faugeras constraints [17]: $\sigma_1 = \sigma_2$ and $\sigma_3 = 0$ (where σ_1, σ_2 and σ_3 are the singular values of $\mathbf{E}$). Indeed, $\mathbf{E}$ can be also written as the product of a skew-symmetric matrix and a rotation matrix:

$$\mathbf{E} = [\mathbf{t}]_\times \mathbf{R} \tag{34}$$

These constraints can be imposed a posteriori by using the algorithm of Tsai and Huang [30] to reconstruct the motion parameters. Indeed, the essential matrix can be decomposed using the Singular Values Decomposition as $\mathbf{E} = \mathbf{U}\mathbf{S}\mathbf{V}^T$, where $\mathbf{U}$ and $\mathbf{V}$ are two orthonormal matrices and $\mathbf{S} = diag(\sigma_1, \sigma_2, \sigma_3)$ a diagonal matrix. After the Huang-Faugeras constraints are imposed $\mathbf{S} =$

$diag(\sigma, \sigma, 0)$ and this matrix can be decomposed as $\mathbf{S} = [\mathbf{t}']_\times \mathbf{R}'$, where $[\mathbf{t}']_\times$ is a skew-symmetric matrix and $\mathbf{R}'$ a rotation matrix. Finally, the essential matrix is decomposed as $\mathbf{E} = (\mathbf{U}[\mathbf{t}']_\times \mathbf{U}^T)(\mathbf{U}\mathbf{R}'\mathbf{V}^T)$, where $[\mathbf{t}]_\times = \mathbf{U}[\mathbf{t}']_\times \mathbf{U}^T$ is the skew-symmetric matrix associated to the translation vector $\mathbf{t}$ and $\mathbf{R} = \mathbf{U}\mathbf{R}'\mathbf{V}^T$ is the rotation matrix. This method has been proved to be optimal by Hartley [13].

Therefore, from the estimated essential matrix, the rotation matrix $\mathbf{R}$ and the direction of translation $\mathbf{t}/\|\mathbf{t}\|$ can be directly calculated. Then, using three reference points (which define the reference plane π in space), the following extrinsic parameters can also be computed: the scaled norm of translation $\|\mathbf{t}\|/d^*$ (where d^* is the distance between $\mathbf{C}^*$ and π) and the normal vector $\mathbf{n}^*$ to π. These parameters are important since they are used in the design of our control scheme.

The Euclidean homography matrix The Euclidean homography matrix can be written as a function of the calibration parameters and of the projective homography matrix as follows:

$$\mathbf{H}_e = \mathbf{A}^{-1}\mathbf{H}_p\mathbf{A} \tag{35}$$

Furthermore, the homography matrix can be written as a function of the motion parameters (as the sum of a rotation matrix and of a rank 1 matrix) [12]:

$$\mathbf{H}_e = \mathbf{R} + \frac{\mathbf{t}}{d^*}\mathbf{n}^{*T} \tag{36}$$

From the estimated homography matrix, $\mathbf{R}$, $\mathbf{t_d} = \mathbf{t}/d^*$, and $\mathbf{n}^*$ can thus be directly calculated without any additional estimation. To reconstruct the motion parameters, one of the algorithms proposed in [12] or [35] can be used. Unfortunately, in the most general case, we have two different solutions. If the target is known to be planar, the indetermination can be eliminated if an additional information is available (for example from the normal vector to the reference plane). If the target is not planar, the indetermination is eliminated by considering another reference plane and choosing the common solution between the two pairs [12]. In visual servoing, this has to be done only once, at the beginning of the process since the normal vector $\mathbf{n}^*$ do not change during

the servoing. Let us notice that the structure of the reference plane can be directly reconstructed from the homography matrix. For example, the following parameters will be used to design the 2D 1/2 visual servoing:

- the ratio r between distances d and d^* from the optical centers C and C^* to π:

$$r = \frac{d}{d^*} = \det(\mathbf{H}_e) = 1 + \mathbf{n}^{*T}\mathbf{R}\frac{\mathbf{t}}{d^*} \tag{37}$$

- the ratio ρ between the Z coordinate of a 3D reference point lying on π and d^*:

$$\rho = \frac{Z}{d^*} = \frac{r}{\mathbf{n}^T\mathbf{p}} \tag{38}$$

Obviously, the same parameters can be also reconstructed using the extrinsic parameters estimated from the essential matrix. However, simulation and experimental results show that a more robust estimation is obtained from the homography matrix. In the following section, we show how to use the data extracted from the scaled Euclidean reconstruction to design a visual servoing scheme with particularly interesting properties. In the following subsection, we will show how it is possible, in the coarse calibrated domain, to obtain an analytical form of the estimated motion parameters as a function of the real motion parameters and of the camera calibration errors.

2.3.2 Coarse camera calibration

The Essential matrix If the camera is not perfectly calibrated and $\widehat{\mathbf{A}}$ is used instead of $\mathbf{A}$, the estimated essential matrix is:

$$\widehat{\mathbf{E}} = \widehat{\mathbf{A}}^T\mathbf{F}\widehat{\mathbf{A}} = \delta\mathbf{A}^T\mathbf{E}\delta\mathbf{A}^{-1} \tag{39}$$

where $\delta\mathbf{A} = \widehat{\mathbf{A}}^{-1}\mathbf{A}$. Indeed, in presence of calibration errors, the measured image point $\widehat{\mathbf{p}}$ can be written in function of the real image point $\mathbf{p} = \mathbf{A}^{-1}\mathbf{m}$ as:

$$\widehat{\mathbf{p}} = \widehat{\mathbf{A}}^{-1}\mathbf{m} = \delta\mathbf{A}\mathbf{p} \tag{40}$$

The homography matrix If the camera is not perfectly calibrated and $\widehat{\mathbf{A}}$ is used instead of $\mathbf{A}$, the estimated homography matrix is:

$$\widehat{\mathbf{H}}_e = \widehat{\mathbf{A}}^{-1}\mathbf{H}_p\widehat{\mathbf{A}} = \delta\mathbf{A}\mathbf{H}_e\delta\mathbf{A}^{-1} \tag{41}$$

The estimated homography matrix can be decomposed as the sum of a matrix similar to a rotation matrix and of a rank 1 matrix:

$$\widehat{\mathbf{H}} = \widehat{\mathbf{H}}_\infty + \widehat{\mathbf{t}}_{d^*}\widehat{\mathbf{n}}^{*T} \tag{42}$$

where $\widehat{\mathbf{H}}_\infty = \delta\mathbf{A}\mathbf{R}\delta\mathbf{A}^{-1}$, $\widehat{\mathbf{t}}_{d^*} = \|\mathbf{n}^{*T}\delta\mathbf{A}^{-1}\|\delta\mathbf{A}\mathbf{t}_{d^*}$ and $\widehat{\mathbf{n}}^{*T} = \frac{\mathbf{n}^{*T}\delta\mathbf{A}^{-1}}{\|\mathbf{n}^{*T}\delta\mathbf{A}^{-1}\|}$. The eigenvalues of $\mathbf{R}$ depend on the angle of rotation θ, and its eigenvector corresponding to the unitary eigenvalues is the axis of rotation $\mathbf{u}$. Matrix $\widehat{\mathbf{H}}_\infty$ is not a rotation matrix, but is similar to $\mathbf{R}$, which implies that the two matrices have the same eigenvalues and the eigenvectors of $\widehat{\mathbf{H}}_\infty$ are the eigenvectors of $\mathbf{R}$ multiplied by matrix $\delta\mathbf{A}$. The estimated rotation angle $\widehat{\theta}$ and the estimated rotation axis $\widehat{\mathbf{u}}$, extracted directly from $\widehat{\mathbf{H}}_\infty$, can be written as a function of the real parameters and of the calibration errors:

$$\widehat{\theta} = \theta \qquad \text{and} \qquad \widehat{\mathbf{u}} = \frac{\delta\mathbf{A}\mathbf{u}}{\|\delta\mathbf{A}\mathbf{u}\|} \tag{43}$$

It must be emphasized that, as well as the rotation angle θ, the ratio r is computed without error:

$$\widehat{r} = \det(\widehat{\mathbf{H}}) = \det(\mathbf{H}) = r \tag{44}$$

Consequently, since $\widehat{\mathbf{n}} = \widehat{\mathbf{R}}^{-T}\widehat{\mathbf{n}}^* = \frac{\delta\mathbf{A}^{-T}\mathbf{n}}{\|\delta\mathbf{A}^{-T}\mathbf{n}^*\|}$, $\widehat{\rho}$ is given by:

$$\widehat{\rho} = \frac{\widehat{r}}{\widehat{\mathbf{n}}^T\widehat{\mathbf{p}}} = \frac{r}{\mathbf{n}^T\mathbf{p}} = \|\delta\mathbf{A}^{-T}\mathbf{n}^*\|\rho \tag{45}$$

The proposed decomposition gives the analytical computation of the estimated parameters as a function of the real parameters and of the camera calibration errors. This will allow us to find analytical conditions on the stability of our visual servoing scheme and on its robustness with respect to camera calibration errors.

3 Application to vision-based robot control

In this section, we briefly recall the classical 3D and 2D visual servoing schemes and discuss their respective advantages and drawbacks. Then, we show how the recontruction results obtained in the previous section can be used to design the *2D 1/2 visual servoing* scheme, which combines the advantages of 2D and 3D visual servoings and avoids their respective drawbacks.

3.1 3D visual servoing

Let $\mathcal{F}_0$ be the coordinate frame attached to the target, $\mathcal{F}_1 = \mathcal{F}^*$ and $\mathcal{F}_2 = \mathcal{F}$ be the coordinate frames attached to the camera in its desired and current position respectively (see Figure 4).

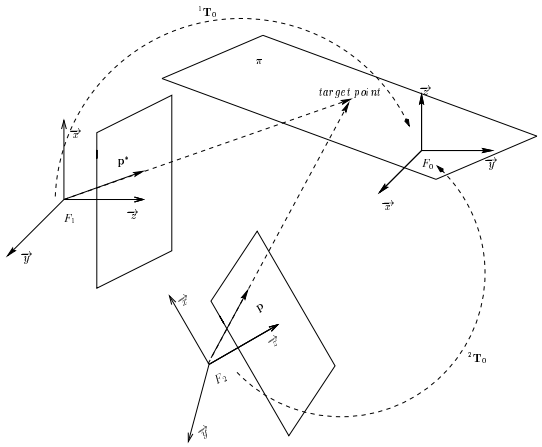


Figure 4: Modelisation of camera displacement for 3D visual servoing

Knowing the coordinates, expressed in $\mathcal{F}_0$, of at least four points of the target [6] (i.e. the 3D model of the target is supposed to be perfectly known), it is possible from their projection to compute the desired camera pose (represented in Figure 4 by the transformation matrix $\mathbf{T}_0^1$) and the current camera pose (represented by $\mathbf{T}_0^2$). The camera displacement to reach the desired position is thus easily obtained, and can be performed either in open loop or, more robustly, in closed-loop. The corresponding block diagram is given in

Figure 5. The main advantage of this approach is that it directly controls the camera trajectory in the Cartesian space. However, since there is no control in the image, the image features used in the pose estimation may get out of the image (especially if the robot or the camera are coarse calibrated), which thus leads to the failure of the servoing. Let us also note that, if the camera is coarse calibrated, or if errors exist in the 3D model of the target, the current and desired camera poses will not be accurately estimated. Since the error made on the pose estimation cannot be computed analytically as a function of the camera calibration errors, it is quite impossible to analyze the stability of the system.

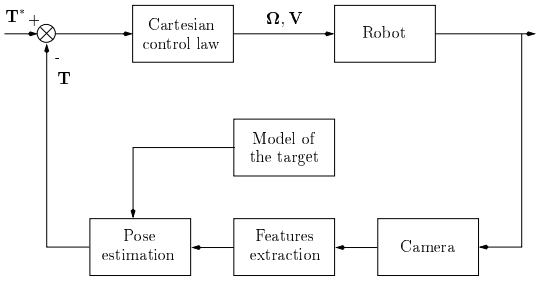


Figure 5: Block diagram of the 3D visual servoing

3.2 2D visual servoing

The control error function is now expressed directly in the 2D image space (see Figure 6). Let $\mathbf{s}$ be the current value of visual features observed by the camera and $\mathbf{s}^*$ be the desired value of $\mathbf{s}$ to be reached in the image. The time variation of $\mathbf{s}$ is related to camera velocity $\mathbf{T} = \left[\begin{array}{cc} \mathbf{V}^T & \Omega^T \end{array} \right]^T$ by [10]:

$$\dot{\mathbf{s}} = \mathbf{L}(\mathbf{s}, Z)\mathbf{T} \tag{46}$$

where $\mathbf{L}(\mathbf{s}, Z)$ is the interaction matrix related to $\mathbf{s}$. For example, if the chosen features are the coordinates $\mathbf{s} = \left[\begin{array}{cc} x & y \end{array} \right]^T = \left[\begin{array}{cc} X/Z & Y/Z \end{array} \right]^T$ in the image of a 3D point $\mathbf{P}$ of coordinates $\left[\begin{array}{ccc} X & Y & Z \end{array} \right]^T$ in the camera frame, the interaction matrix related to $\mathbf{s}$ is given by:

$$\mathbf{L}(\mathbf{s}, Z) = \left[\begin{array}{cccccc} -1/Z & 0 & x/Z & xy & -(1+x^2) & y \\ 0 & -1/Z & y/Z & (1+y^2) & -xy & -x \end{array} \right] \tag{47}$$

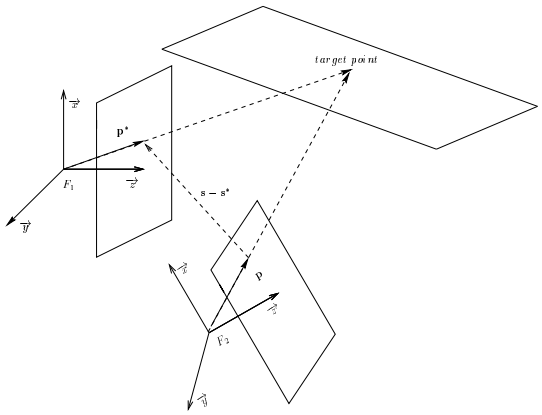


Figure 6: Modelisation of camera displacement for 2D visual servoing

The interaction matrix for more complex image features can be found in [10]. The vision-based task $\mathbf{e}$ (to be regulated to 0), corresponding to the regulation of $\mathbf{s}$ to $\mathbf{s}^*$, is defined by:

$$\mathbf{e} = \mathbf{C}(\mathbf{s} - \mathbf{s}^*) \tag{48}$$

where $\mathbf{C}$ is a matrix which has to be selected such that $\mathbf{CL}(\mathbf{s}, Z) > 0$ in order to ensure the global stability of the control law. The optimal choice is to consider $\mathbf{C}$ as the pseudo-inverse $\mathbf{L}(\mathbf{s}, Z)^+$ of the interaction matrix. The matrix $\mathbf{C}$ thus depends on the depth Z of each target point used in the visual servoing. An estimation of the depth can be obtained using, as in 3D visual servoing, a pose determination algorithm (if a 3D target model is available), or using a structure from known motion algorithm (if the camera motion can be measured). However, using this choice for $\mathbf{C}$ may lead the system to near, or even reach, a singularity of the interaction matrix. Furthermore, the convergence may also not be attained due to local minima reached because of the computation by the control law of unrealizable motions in the image [4].

Another choice is to consider $\mathbf{C}$ as a constant matrix equal to $\mathbf{L}(\mathbf{s}^*, Z^*)^+$, the pseudo-inverse of the interaction matrix computed for $\mathbf{s} = \mathbf{s}^*$ and $Z = Z^*$, where Z^* is an approximate value of Z at the desired camera position. In this simple case, the condition for convergence is satisfied only in the neighborhood of the desired position, which means that the convergence may not be ensured

if the initial camera position is too far away from the desired one. If an exponential convergence is desired ($\dot{\mathbf{e}} = -\lambda \mathbf{e}$ where λ is a positive scalar) and the target is known to be motionless (see [2, 1, 27] otherwise), a simple control law is given by [10]:

$$\mathbf{T} = -\lambda \mathbf{e} = -\lambda \mathbf{C}(\mathbf{s} - \mathbf{s}^*) \tag{49}$$

The block diagram of the 2D visual servoing approach is given in Figure 7.

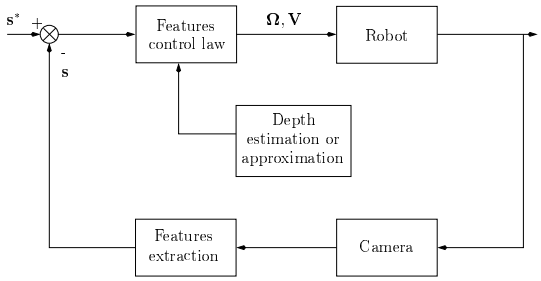


Figure 7: Block diagram of the 2D visual servoing

3.3 2D 1/2 visual servoing

In the following subsections, we discuss the choice of the control vector which may be designed to realize a positioning task controlling the six camera d.o.f.

3.3.1 Orientation control vector

In order to control the orientation of the camera, we use of course the 3D estimated rotation $\mathbf{R}$ between $\mathcal{F}$ and $\mathcal{F}^*$ (which has to reach the identity matrix). Let $\mathbf{u}$ be the rotation axis and θ the rotation angle obtained from $\mathbf{R}$. The angular velocity $\dot{\theta}$ around the axis of rotation $\mathbf{u}$ is simply expressed as a function of the camera velocity screw $\mathbf{T}$ as:

$$\mathbf{u}\,\dot{\theta} = \begin{bmatrix} \mathbf{0} & \mathbf{I}_3 \end{bmatrix} \mathbf{T} \tag{50}$$

The control of the camera orientation is thus decoupled from the control of its position since the former is directly available from the obtained partial pose. Let us now discuss the choice of the position control vector.

3.3.2 Position control vector

3D position control It is possible to design a control law directly in the Cartesian space, such that $\mathbf{t}_{d^*}$ has to reach $\left[\begin{array}{ccc} 0 & 0 & 0 \end{array}\right]^T$ (which thus implies the achievement of the positioning task). A scheme very similar to the classical 3D visual servoing can hence be performed without knowing the 3D structure of the target. The relationship between the time variation of $\mathbf{t}_{d^*}$ and the velocity of the camera is:

$$\dot{\mathbf{t}}_{d^*} = \left[\begin{array}{cc} -\frac{1}{d^*}\mathbf{I}_3 & [\mathbf{t}_{d^*}]_\times \end{array}\right] \mathbf{T} \tag{51}$$

where $[\mathbf{t}_{d^*}]_\times$ is the skew-symmetric matrix associated to vector $\mathbf{t}_{d^*}$. Using such a control vector, it is quite easy to demonstrate the stability of the obtained system in all the task space (i.e. the half space in front of the reference plane) if perfect measurements and estimations are performed. However, as for 3D visual servoing, such a control vector does not ensure that the considered object will always remain in the camera field of view. Getting out of the image may thus occur, particularly in the presence of important camera or robot calibration errors.

2D position control It is also possible to control the camera position directly in the image space (as it is done for the 2D visual servoing, the main difference being that the orientation is controlled using the result of the motion estimation). For example, if the chosen features are the coordinates $\mathbf{s} = \left[\begin{array}{cc} x & y \end{array}\right]^T = \left[\begin{array}{cc} X/Z & Y/Z \end{array}\right]^T$ in the image of a 3D point $\mathbf{P}$ of coordinates $\left[\begin{array}{ccc} X & Y & Z \end{array}\right]^T$ in the camera frame, the time variation of $\mathbf{s}$ is given by equation (46). The interaction matrix related to a 2D point is function of the unknown Z coordinate of the corresponding 3D point. In the present case, if the image coordinates of the three 3D points belonging to the reference plane π are used, their depth $Z_i = d^*\rho_i$ can be easily computed up to the scale factor d^* using equation (38). Then the image Jacobian matrix can be written ($i = 1, 2, 3$):

$$\mathbf{L}_i(\mathbf{s}_i, \rho_i, d^*) = \left[\begin{array}{cccccc} -1/(d^*\rho_i) & 0 & x_i/(d^*\rho_i) & x_i y_i & -1 - x_i^2 & y_i \\ 0 & -1/(d^*\rho_i) & y_i/(d^*\rho_i) & 1 + y_i^2 & -x_i y_i & -x_i \end{array}\right] \tag{52}$$

Contrarily to the 2D visual servoing, in the present case, the decoupled control of the camera orientation allows the system to avoid local minima. However, the stability analysis is analytically quite as difficult as for the 2D visual servoing. Furthermore, the camera position is totally controlled in the image space, and the Cartesian trajectory of the camera is thus unpredictable. Experimental results show that, using this approach when the camera displacement is too large, the robot may unfortunately reach its joints limits, or the target may become so little in the image that the visual servoing has to be stopped.

2D 1/2 position control The previous control schemes have some advantages but similar drawbacks to the 3D and 2D visual servoings. We have therefore preferred another more satisfactory solution. The position of the camera can be controlled in the image space and in the Cartesian space at the same time. In order to maintain the target in the camera field of view, we introduced in [25] the use of three independent visual features, such as the two image coordinates of a target point and the ratio $r = \frac{d}{d^*}$, computed from equation (37), which controls the depth between the camera and the target. However, the stability analysis is again analytically very difficult in presence of calibration errors. In order to simplify the problem, we describe now a more satisfactory solution.

Consider a point $\mathbf{P}$ (called the reference point) lying in the reference plane π. The time derivative of its coordinates, expressed in the current camera frame, can be written as:

$$\dot{\mathbf{P}} = \begin{bmatrix} -\mathbf{I}_3 & [\mathbf{P}]_\times \end{bmatrix} \mathbf{T} \tag{53}$$

Let us define the *extended image coordinates* $\mathbf{p}_e$ as follows:

$$\mathbf{p}_e = \begin{bmatrix} x & y & z \end{bmatrix}^T = \begin{bmatrix} \frac{X}{Z} & \frac{Y}{Z} & \log(\rho) \end{bmatrix}^T \tag{54}$$

where $z = \log(\rho) = \log(Z/d^*)$ is a supplementary normalized coordinate which can be computed from equation (38). The time derivative of the extended

image coordinates can be written as:

$$\dot{\mathbf{p}}_e = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} \frac{\dot{X}}{Z} - \frac{X}{Z}\frac{\dot{Z}}{Z} \\ \frac{\dot{Y}}{Z} - \frac{X}{Z}\frac{Z}{Z} \\ \frac{\dot{Z}}{Z} \end{bmatrix} = \frac{1}{Z}\begin{bmatrix} 1 & 0 & -\frac{X}{Z} \\ 0 & 1 & -\frac{Y}{Z} \\ 0 & 0 & 1 \end{bmatrix}\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = -\frac{1}{d^*\rho}\mathbf{L}_v(\mathbf{p})\dot{\mathbf{P}} \tag{55}$$

where $\mathbf{L}_v(\mathbf{p})$ is an upper triangular matrix given by:

$$\mathbf{L}_v(\mathbf{p}) = \begin{bmatrix} -1 & 0 & x \\ 0 & -1 & y \\ 0 & 0 & -1 \end{bmatrix} \tag{56}$$

Then, using equation (53) and equation (55), we finally obtain (since $\mathbf{P} = Z\mathbf{p}$):

$$\dot{\mathbf{p}}_e = \begin{bmatrix} \frac{1}{d^*\rho}\mathbf{L}_v(\mathbf{p}) & \mathbf{L}_\omega(\mathbf{p}) \end{bmatrix}\mathbf{T} = \mathbf{L}_e(\mathbf{p}_e, d^*)\mathbf{T} \tag{57}$$

where $\mathbf{L}_e$ is the interaction matrix related to the extended image coordinates, and:

$$\mathbf{L}_\omega(\mathbf{p}) = \mathbf{L}_v(\mathbf{p})[\mathbf{p}]_\times = \begin{bmatrix} xy & -(1+x^2) & y \\ (1+y^2) & -xy & -x \\ -y & x & 0 \end{bmatrix} \tag{58}$$

3.3.3 Control law

A general positioning task controlling the 6 camera d.o.f. can be described as the regulation to zero of a task function [28]. Let us remark that, as described previously, building a task function from $\mathbf{t}_{d^*}$ or $\mathbf{s}_i$ $(i = 1, 2, 3)$, the camera can be controlled in the Cartesian space or in the image space respectively. In the case of 2D 1/2 visual servoing, the positioning task can be described as the regulation to zero of the following task function:

$$\mathbf{e} = \begin{bmatrix} \mathbf{p}_e - \mathbf{p}_e^* & \mathbf{u}^T\theta \end{bmatrix}^T \tag{59}$$

where $\mathbf{p}_e^*$ can be estimated from the desired image. The exponential convergence of $\mathbf{p}_e$ toward $\mathbf{p}_e^*$ and $\mathbf{u}\theta$ toward 0 can be obtained by imposing $\dot{\mathbf{e}} = -\lambda\mathbf{e}$

(where λ tunes the convergence rate). The time derivative of the task function is related to the camera velocity screw by:

$$\dot{\mathbf{e}} = \mathbf{L}\mathbf{T} \tag{60}$$

where $\mathbf{L}$ is an upper triangular matrix given by:

$$\begin{aligned} \mathbf{L} &= \begin{bmatrix} \frac{1}{d^*\rho}\mathbf{L}_v & \mathbf{L}_\omega \\ \mathbf{0} & \mathbf{I}_3 \end{bmatrix} = \\ &= \begin{bmatrix} -1/(d^*\rho) & 0 & x/(d^*\rho) & xy & -(1+x^2) & y \\ 0 & -1/(d^*\rho) & y/(d^*\rho) & (1+y^2) & -xy & -x \\ 0 & 0 & -1/(d^*\rho) & -y & x & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned} \tag{61}$$

Then, if the target is assumed to be motionless, the corresponding control law is given by:

$$\mathbf{T} = -\lambda \widehat{\mathbf{L}}^{-1} \widehat{\mathbf{e}} \tag{62}$$

where $\mathbf{T}$ is the camera velocity sent to the robot controller, $\widehat{\mathbf{e}}$ is the measured task function and $\widehat{\mathbf{L}}$ is an approximation of the interaction matrix $\mathbf{L}$. Since $\mathbf{L}$ depends on the unknown distance d^*, an approximate value $\widehat{d^*}$ has thus to be chosen during the off-line learning stage and introduced in $\widehat{\mathbf{L}}^{-1}$. However, this value has not to be precisely determined (by hand in the following experiments) since it has a small influence on the stability of the system. More precisely, it influences the time-to-convergence of the translational velocity and the amplitude of the possible tracking error due to a wrong compensation of the rotational motion. As far as the tracking error is concerned, it is proportional to the rotational velocity and thus disappears when the camera is correctly oriented. Finally, the control law is given by:

$$\begin{bmatrix} \mathbf{V} \\ \Omega \end{bmatrix} = -\lambda \begin{bmatrix} \widehat{d^*}\widehat{\rho}\widehat{\mathbf{L}}_v^{-1} & -\widehat{d^*}\widehat{\rho}\widehat{\mathbf{L}}_v^{-1}\widehat{\mathbf{L}}_\omega \\ \mathbf{0} & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} \widehat{\mathbf{p}}_e - \widehat{\mathbf{p}}_e^* \\ \widehat{\mathbf{u}}^T\widehat{\theta} \end{bmatrix} \tag{63}$$

Let us remark that the rotational control loop is decoupled from the translational one (see Figure 8). A such decoupled system allows us to obtain the

convergence in all the task space if exact model and perfect measurements are assumed. Furthermore, and contrarily to 2D and 3D visual servoings, we will show in the following section that it is possible to obtain the necessary and sufficient conditions for local asymptotic stability, and sufficient conditions for global asymptotic stability in presence of camera calibration errors.

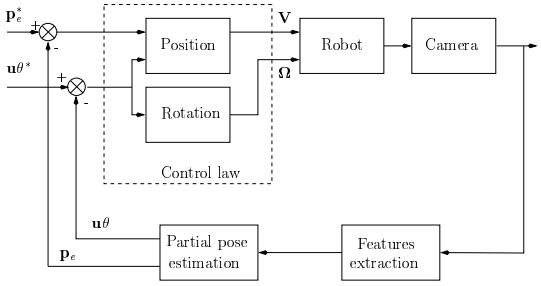


Figure 8: Block diagram of the 2D 1/2 visual servoing

4 Stability analysis of the 2D 1/2 visual servoing

Let us assume that the only possible errors are on the intrinsic camera parameters and on d^*. The task function can be reconstructed as:

$$\widehat{\mathbf{e}} = \begin{bmatrix} \mathbf{E}_\tau & 0 \\ 0 & \mathbf{E}_\theta \end{bmatrix} \mathbf{e} \tag{64}$$

with (see equation (43)):

$$\mathbf{E}_\tau = \begin{bmatrix} \delta\mathbf{A}_{11} & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{E}_\theta = \mu \begin{bmatrix} \delta\mathbf{A}_{11} & \delta\mathbf{p}_0 \\ 0 & 1 \end{bmatrix} \tag{65}$$

where $\delta\mathbf{A}_{11}$ is the (2×2) sub-matrix of $\delta\mathbf{A}$ containing the pixel lengths (see (8)), $\delta\mathbf{p}_0$ is the (2×1) sub-vector containing the error on the image center and $\mu = \frac{1}{\|\delta\mathbf{A}\mathbf{u}\|}$. The closed-loop system can be written:

$$\dot{\mathbf{e}} = \mathbf{f}(\mathbf{e}) = -\lambda \mathbf{Q}(\mathbf{e})\mathbf{e} \tag{66}$$

where, denoting $\nu = \frac{\widehat{d^*}\widehat{\rho}}{d^*\rho} = \frac{\widehat{d^*}}{d^*}\|\delta\mathbf{A}^{-T}\mathbf{n}^*\|$, matrix $\mathbf{Q}(\mathbf{e})$ is given by:

$$\mathbf{Q}(\mathbf{e}) = \begin{bmatrix} \nu\mathbf{L}_v\widehat{\mathbf{L}}_v^{-1}\mathbf{E}_\tau & \mathbf{L}_v([\mathbf{p}]_\times - \nu[\widehat{\mathbf{p}}]_\times)\mathbf{E}_\theta \\ \mathbf{0} & \mathbf{E}_\theta \end{bmatrix} \tag{67}$$

Function $\mathbf{f}$ is a C^∞ vector field defined on an open subset S of SE_3. It is easy to show the existence and uniqueness of the equilibrium point:

Proposition 1 *The only point of equilibrium for* $\mathbf{f}$*, i.e a point* $\mathbf{e}^\circ \in S$ *such that* $\mathbf{f}(\mathbf{e}^\circ) = 0$*, is* $\mathbf{e}^\circ = 0$*.*

Proof of proposition 1 *The existence of the equilibrium point is evident since if* $\mathbf{e} = 0$*, then* $\mathbf{f}(\mathbf{e}) = \mathbf{Q}(\mathbf{e})\mathbf{e} = 0$*. This equilibrium point is unique if and only if* $\det(\mathbf{Q}(\mathbf{e})) \neq 0, \quad \forall \mathbf{e} \in S$*. Since matrix* $\mathbf{Q}$ *is upper triangular, its determinant can be easily calculated:*

$$\begin{aligned} \det(\mathbf{Q}) &= \nu^3 \det(\mathbf{L}_v) \det(\widehat{\mathbf{L}}_v^{-1}) \det(\mathbf{E}_\tau) \det(\mathbf{E}_\theta) = \\ &= \nu^3\mu^3 \frac{\alpha_u^2}{\widehat{\alpha}_u^2}\frac{\alpha_v^2}{\widehat{\alpha}_v^2} \end{aligned} \tag{68}$$

then $\det(\mathbf{Q}) \neq 0, \forall \mathbf{e} \in S$ *since* $\nu \neq 0$ *and* $\mu \neq 0$*.*

Therefore, the task has no singularity and, if the task function decreases, it decreases towards 0.

4.1 Local asymptotic stability

Theorem 1 (Local asymptotic stability) *The differential system (66) is locally asymptotically stable around the equilibrium point* $\mathbf{e}^\circ$ *if and only if:*

$$\frac{\widehat{\alpha}_u}{\alpha_u} > 0, \quad \frac{\widehat{\alpha}_v}{\alpha_v} > 0 \quad \textit{and} \quad \frac{\widehat{d}^*}{d^*} > 0 \tag{69}$$

In practice, these conditions are of course easily verified.

Proof of theorem 1 *Consider the first order Taylor expansion of the non-linear differential system (66) around the equilibrium point:*

$$\dot{\mathbf{e}} \approx -\lambda \mathbf{Q}(\mathbf{e}^\circ)(\mathbf{e} - \mathbf{e}^\circ) \tag{70}$$

The linearized system (70) is asymptotically stable if and only if the eigenvalues of $\mathbf{Q}(\mathbf{e}^\circ)$ *are positive. They are given by:*

$$\begin{array}{lll} \lambda_1 = \nu\alpha_u/\widehat{\alpha}_u \ , & \lambda_3 = \nu \ , & \lambda_5 = \mu\alpha_u/\widehat{\alpha}_u \ , \\ \lambda_2 = \nu\alpha_v/\widehat{\alpha}_v \ , & \lambda_4 = \mu \ , & \lambda_6 = \mu\alpha_v/\widehat{\alpha}_v \end{array} \tag{71}$$

These values are positive if and only if conditions (69) are verified. Since the linearized system is asymptotically stable, the local asymptotic stability, around the equilibrium point, of the non-linear system is proven.

Let us remark that these conditions are also necessary and sufficient for global asymptotic stability of the orientation control subsystem. However, it is well known that the error may considerably increase before convergence towards zero. The target may thus get out of the camera field of view during the servoing. Indeed, it must be emphasized that the differential equation (66) is constrained by $\mathbf{e} \in \mathcal{S}$. Therefore, it is more interesting to find sufficient conditions to ensure the decreasing of the error at each iteration, which ensures the global asymptotic stability of the system.

4.2 Global asymptotic stability

It is well known that a sufficient condition to ensure the decreasing of the error at each iteration is the positiveness of matrix $\mathbf{Q}(\mathbf{e}) > 0, \ \forall \mathbf{e}$. Indeed, multiplying equation (66) by vector $\mathbf{e}^T$ we obtain:

$$\mathbf{e}^T\dot{\mathbf{e}} = -\lambda \mathbf{e}^T\mathbf{Q}(\mathbf{e})\mathbf{e} \tag{72}$$

Then, if $\mathbf{Q}(\mathbf{e}) > 0$, the time derivative of the task function norm is negative:

$$\frac{1}{2}\frac{d\|\mathbf{e}\|}{dt} = -\lambda \mathbf{e}^T \mathbf{Q}(\mathbf{e})\mathbf{e} < 0 \tag{73}$$

and the error decreases at each iteration. Since the only point of equilibrium of the system is $\mathbf{e}^\circ = 0$, the task function decreases towards zero.

Theorem 2 (Global asymptotic stability) *The differential system (66) is globally asymptotically stable, only if conditions (69) are verified, and if:*

$$\sigma = \frac{\alpha_u}{\widehat{\alpha}_u} + \frac{\alpha_v}{\widehat{\alpha}_v} - \sqrt{\left(\frac{\alpha_u}{\widehat{\alpha}_u} - \frac{\alpha_v}{\widehat{\alpha}_v}\right)^2 + \left(\frac{\alpha_{uv}}{\widehat{\alpha}_u} - \frac{\widehat{\alpha}_{uv}}{\widehat{\alpha}_u}\frac{\alpha_v}{\widehat{\alpha}_v}\right)^2} > 0 \tag{74}$$

$$\sigma_1 = \sigma + 1 - \sqrt{(\sigma-1)^2 + \|\mathbf{I} - \delta\mathbf{A}\|^2(1+\overline{\gamma}^2)} > 0 \tag{75}$$

$$\sigma_2 = \sigma + 1 - \sqrt{(\sigma-1)^2 + \|\delta\mathbf{p}_0\|^2} > 0 \tag{76}$$

$$\mu(\nu\|\mathbf{I} - \delta\mathbf{A}\| + |1-\nu|)^2\|\delta\mathbf{A}\|^2 g^2(\overline{\gamma}) < \nu\sigma_1\sigma_2 \tag{77}$$

where $\overline{\gamma} = \tan(\psi)$ is the tangent of the vision angle ψ ($\overline{\gamma}$ is the maximum value of $\gamma = \sqrt{x^2+y^2}$) and:

$$g^2(\overline{\gamma}) = \frac{2 + \overline{\gamma}^2 + \sqrt{\overline{\gamma}^2(\overline{\gamma}^2+4)}}{2}(1+\overline{\gamma}^2) \tag{78}$$

The proof is given in Appendix B. The sufficient conditions can be analyzed in order to have an approximative idea of the convergence domain. However, they depend on $\delta\mathbf{A}$ and μ (which depends on the rotation axis $\mathbf{u}$). In order to simplify the analysis, we now present the following corollary whose proof is given in Appendix C.

Corollary (Global asymptotic stability) *The differential system (66) is globally asymptotically stable, only if conditions (69) are verified, and if:*

$$\sigma > 0 \tag{79}$$

$$\sigma_3 = \sigma + 1 - \sqrt{(\sigma-1)^2 + (\|\mathbf{I} - \delta\mathbf{A}_{11}\| + \|\delta\mathbf{p}_0\|)^2(1+\overline{\gamma}^2)} > 0 \tag{80}$$

$$\sqrt{2}\frac{(\nu(\|\mathbf{I} - \delta\mathbf{A}_{11}\| + \|\delta\mathbf{p}_0\|) + |1-\nu|)^2(\|\delta\mathbf{A}_{11}\| + \sqrt{1+\|\delta\mathbf{p}_0\|^2})^2 g^2(\overline{\gamma})}{\sqrt{\sigma' + 1 - \sqrt{(\sigma'-1)^2 + 4\|\delta\mathbf{A}_{11}\|^2\|\delta\mathbf{p}_0\|^2}}} < \nu\sigma_2\sigma_3 \tag{81}$$

where:

$$\sigma' = \frac{1}{2}\left(\frac{\alpha_u^2}{\widehat{\alpha}_u^2} + \frac{\alpha_v^2}{\widehat{\alpha}_v^2} + \left(\frac{\alpha_{uv}}{\widehat{\alpha}_u} - \frac{\widehat{\alpha}_{uv}}{\widehat{\alpha}_u}\frac{\alpha_v}{\widehat{\alpha}_v}\right)^2 - \sqrt{\left(\frac{\alpha_u^2}{\widehat{\alpha}_u^2} + \frac{\alpha_v^2}{\widehat{\alpha}_v^2} + \left(\frac{\alpha_{uv}}{\widehat{\alpha}_u} - \frac{\widehat{\alpha}_{uv}}{\widehat{\alpha}_u}\frac{\alpha_v}{\widehat{\alpha}_v}\right)^2\right)^2 - 4\frac{\alpha_u^2}{\widehat{\alpha}_u^2}\frac{\alpha_v^2}{\widehat{\alpha}_v^2}}\right) \tag{82}$$

These sufficient conditions are more restrictive than the conditions imposed by Theorem 2. However, these new conditions do not depend on the axis of rotation $\mathbf{u}$, and the error $\delta\mathbf{p}_0$ on the principal point is separated from the error $\delta\mathbf{A}_{11}$ on the pixel lengths. The analysis is thus simplified even if the conditions seem more complex.

Let us first analyze condition (79) in more details. This condition is equivalent to the following one:

$$\left(\frac{\alpha_{uv}}{\alpha_u}\frac{\alpha_u}{\widehat{\alpha}_u} - \frac{\widehat{\alpha}_{uv}}{\widehat{\alpha}_u}\frac{\alpha_v}{\widehat{\alpha}_v}\right)^2 < 4\frac{\alpha_u}{\widehat{\alpha}_u}\frac{\alpha_v}{\widehat{\alpha}_v} \tag{83}$$

Since $\cot(\phi) = \frac{\alpha_{uv}}{\alpha_u}$ (see equation (8)), we have:

$$\left(\cot(\phi)\frac{\alpha_u}{\widehat{\alpha}_u} - \cot(\widehat{\phi})\frac{\alpha_v}{\widehat{\alpha}_v}\right)^2 < 4\frac{\alpha_u}{\widehat{\alpha}_u}\frac{\alpha_v}{\widehat{\alpha}_v} \tag{84}$$

Let us note $\delta\phi = \widehat{\phi} - \phi$ the error made on the estimation of the angle ϕ. In general, the estimated angle $\widehat{\phi}$ is set to $\pi/2$. We show now that this choice do not have a great influence on the stability of the system. Indeed, we thus have $\cot(\widehat{\phi}) = 0$, $\cot(\phi) = \cot(\widehat{\phi} - \delta\phi) = \cot(\pi/2 - \delta\phi) = \tan(\delta\phi)$ and condition (84) can be written as:

$$\tan^2(\delta\phi) < 4\frac{\widehat{\alpha}_u}{\widehat{\alpha}_v}\frac{\alpha_v}{\alpha_u} \qquad \Rightarrow \qquad |\delta\phi| < \arctan\left(2\sqrt{\frac{\alpha_v}{\widehat{\alpha}_v}/\frac{\alpha_u}{\widehat{\alpha}_u}}\right) \tag{85}$$

In Figure 9, the bounds for $|\delta\phi|$ are plotted versus the ratio $\alpha_u/\widehat{\alpha}_u$ on the x axis and versus the ratio $\alpha_v/\widehat{\alpha}_v$ on the y axis.

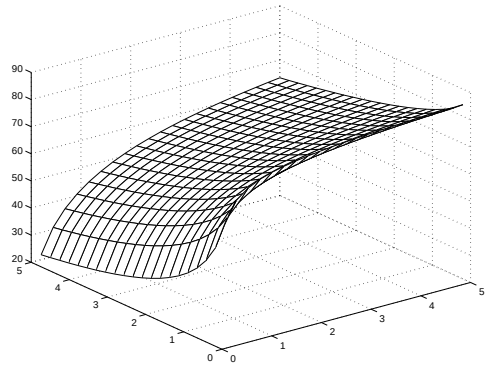


Figure 9: Stability bounds for $|\delta\phi|$

If the ratios are well estimated then $|\delta\phi| < 63$ dg. In the worst considered case, when the estimated ratio is $\widehat{\alpha}_u/\widehat{\alpha}_v = 1/5$ and the real ratio is $\alpha_u/\alpha_v = 5$ (which corresponds to a rectangular pixel with the x-length five times the y-length), we have $|\delta\phi| < 22$ dg. For a more common camera with $\alpha_u/\alpha_v = 3/4$, if $\widehat{\alpha}_u/\widehat{\alpha}_v = 4/3$ then $|\delta\phi| < 56$ dg, which seems impossible to not satisfy.

Let us now analyze condition (80). This condition can be written as:

$$\|(\mathbf{I} - \delta\mathbf{A}_{11})\| + \|\delta\mathbf{p}_0\| < 2\sqrt{\sigma}/\sqrt{1+\overline{\gamma}^2} \tag{86}$$

which implies:

$$\|\delta\mathbf{p}_0\| < 2\sqrt{\sigma}/\sqrt{1+\overline{\gamma}^2} - \|(\mathbf{I} - \delta\mathbf{A}_{11})\| \tag{87}$$

In Figure 10, the bounds for $\|\delta\mathbf{p}_0\|$ are plotted versus the ratio $\alpha_u/\widehat{\alpha}_u$ on the x axis and versus the ratio $\alpha_v/\widehat{\alpha}_v$ on the y axis (in order to obtain these results, we have set $\overline{\gamma} = 0.364$, which corresponds to a camera with a 20 dg vision angle, and $\delta\phi = 20$ dg). In the worst considered case, when $\alpha_u/\widehat{\alpha}_u = 0.5$ and $\alpha_v/\widehat{\alpha}_v = 2$ then $\|\delta\mathbf{p}_0\| < 0.44$. Since $\|\delta\mathbf{p}_0\| = \sqrt{\widehat{x}_0^2 + \widehat{y}_0^2}$, this means that the 3D vector associated to the estimated principal point must be in a cone of angle $\psi_0 < 24$ dg.

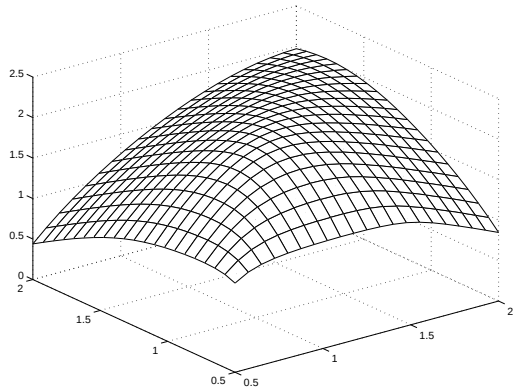


Figure 10: Stability bounds for $\|\delta \mathbf{p}_0\|$

The previous conditions are quite easy to verify. However, the stability domain is not so large since condition (81) must also be verified. This last condition can be written as a second degree inequality (where $\nu = \|\delta \mathbf{A}^{-T}\mathbf{n}^*\| d^*/\widehat{d}^*$):

$$c_2\nu^2 + c_1\nu + c_0 < 0 \tag{88}$$

where ($s_\nu = sign(1-\nu)$):

$$c_2 = \sqrt{2}\frac{(\|\mathbf{I} - \delta\mathbf{A}_{11}\| + \|\delta\mathbf{p}_0\| - s_\nu)^2(\|\delta\mathbf{A}_{11}\| + \sqrt{1+\|\delta\mathbf{p}_0\|^2})^2 g^2(\overline{\gamma})}{\sqrt{\sigma' + 1 - \sqrt{(\sigma'-1)^2 + 4\|\delta\mathbf{A}_{11}\|^2\|\delta\mathbf{p}_0\|^2}}} \tag{89}$$

$$c_1 = 4s_\nu\frac{(\|\mathbf{I} - \delta\mathbf{A}_{11}\| + \|\delta\mathbf{p}_0\| - s_\nu(\|\delta\mathbf{A}_{11}\| + \sqrt{1+\|\delta\mathbf{p}_0\|^2})^2 g^2(\overline{\gamma})}{\sqrt{2}\sqrt{\sigma' + 1 - \sqrt{(\sigma'-1)^2 + 4\|\delta\mathbf{A}_{11}\|^2\|\delta\mathbf{p}_0\|^2}}} - \sigma_2\sigma_3 \tag{90}$$

$$c_0 = \sqrt{2}\frac{(\|\delta\mathbf{A}_{11}\| + \sqrt{1+\|\delta\mathbf{p}_0\|^2})^2 g^2(\overline{\gamma})}{\sqrt{\sigma' + 1 - \sqrt{(\sigma'-1)^2 + 4\|\delta\mathbf{A}_{11}\|^2\|\delta\mathbf{p}_0\|^2}}} \tag{91}$$

The coefficients of the inequality (88) depend on the sign of $1-\nu$ which can be verified after resolution. The solution of the second degree inequality is:

$$\nu_{min} = \frac{-c_1 - \sqrt{c_1^2 - 4c_0c_2}}{2c_0} < \nu < \frac{-c_1 + \sqrt{c_1^2 - 4c_0c_2}}{2c_0} = \nu_{max} \tag{92}$$

As a first example, we obtain a sufficient condition when the hypothesis of exact measurements cannot be applied to the distance d^* since it is estimated by hand. In that case, where perfect camera calibration is assumed ($\delta\mathbf{A}_{11} = \mathbf{I}_2$, $\delta\mathbf{p}_0 = \mathbf{0}$, $\sigma = \sigma_1 = \sigma_2 = 2$, $\sigma' = 1$ and $\nu = \frac{\widehat{d}^*}{d^*}$), condition (81) can be written:

$$(1 - \frac{\widehat{d}^*}{d^*})^2 g^2(\overline{\gamma}) < 4\frac{\widehat{d}^*}{d^*} \tag{93}$$

The corresponding solution of the inequality (92) is:

$$1 + 2\frac{1 - \sqrt{g^2(\overline{\gamma}) + 1}}{g^2(\overline{\gamma})} < \frac{\widehat{d}^*}{d^*} < 1 + 2\frac{1 + \sqrt{g^2(\overline{\gamma}) + 1}}{g^2(\overline{\gamma})} \tag{94}$$

The two bounds are plotted in Figure 11 versus $\overline{\gamma}$. This means that, if we

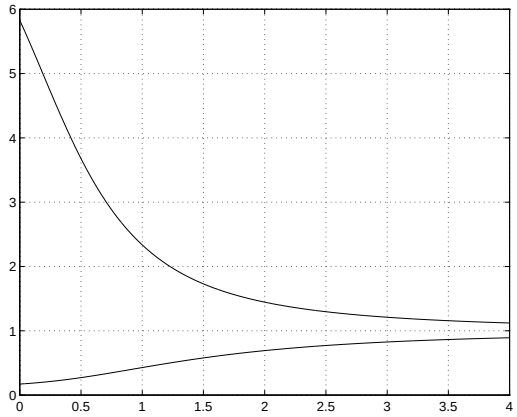


Figure 11: Stability bounds for relative depth $d^*/\widehat{d}^*$

consider for example a camera with a 20° vision angle (then $\overline{\gamma} = 0.364$), the stability condition is verified if $0.24 < \widehat{d}^*/d^* < 4.22$. If the real distance d^* is 50 cm, the system will asymptotically converge for any initial position in the task space if $\widehat{d}^*$ is chosen between 12 and 211 cm. This result definitively validates the robustness of our control scheme in absence of camera calibration errors.

Moreover, similar results can be obtained by considering camera calibration errors. Condition (81) depends on the five camera intrinsic parameters, then

we first study the stability with a fixed $\delta\mathbf{A}_{11}$ and a variable $\delta\mathbf{p}_0$ and after with a variable $\delta\mathbf{A}_{11}$ and a fixed $\delta\mathbf{p}_0$. It must be noted that, if $\mathbf{n}^* \approx \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$, then $\nu \approx d^*/\widehat{d}^*$, $\forall\delta\mathbf{A}$.

We first consider $\alpha_u/\widehat{\alpha}_u = \alpha_v/\widehat{\alpha}_v = 1.5$ (which means 50% error on each pixel length) and $\delta\phi = 5$ dg. The two bounds are plotted in Figure 12. This

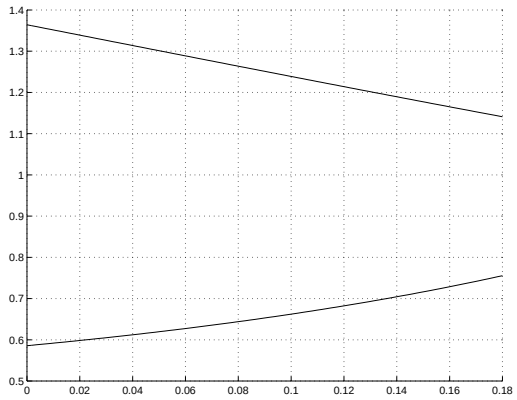


Figure 12: Stability bounds for ν versus $\|\delta\mathbf{p}_0\|$

means that, if $\|\delta\mathbf{p}_0\| = 0.0875$ (which corresponds to a cone with a 5 dg angle), then $0.60 < \nu < 1.26$. In order to obtain a simpler interpretation of this condition, we suppose now that $\nu \approx d^*/\widehat{d}^*$ (which means that the normal to the reference plane is $\mathbf{n}^* \approx \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}^T$). If the real distance d^* is again 50 cm, the system will asymptotically converge for any initial position in the task space if $\widehat{d}^*$ is chosen between 30 and 63 cm. If this hypothesis on $\mathbf{n}^*$ is not verified, the stability domain is smaller. Indeed, $\nu = \|\delta\mathbf{A}^{-T}\mathbf{n}^*\|d^*/\widehat{d}^*$ then $\sqrt{\rho^{-1}(\delta\mathbf{A}^T\delta\mathbf{A})}\|\mathbf{n}^*\| \leq d^*/\widehat{d}^* \leq \sqrt{\rho(\delta\mathbf{A}^{-T}\delta\mathbf{A}^{-1})}\|\mathbf{n}^*\|$ (where $\rho^{-1}(\mathbf{M}^{-1}) = \min_{1\leq i\leq n}\lambda_i$ and $\rho(\mathbf{M}) = \max_{1\leq i\leq n}\lambda_i$) . Since $\|\mathbf{n}^*\| = 1$, the stability of the system will be ensured if (see equation (92)):

$$\nu_{min}\sqrt{\rho(\delta\mathbf{A}^T\delta\mathbf{A})} < d^*/\widehat{d}^* < \sqrt{\rho^{-1}(\delta\mathbf{A}^{-T}\delta\mathbf{A}^{-1})}\nu_{max} \tag{95}$$

In the considered case, we have $\sqrt{\rho^{-1}(\delta\mathbf{A}^{-T}\delta\mathbf{A}^{-1})} = 0.99$ and $\sqrt{\rho(\delta\mathbf{A}^T\delta\mathbf{A})} = 1.57$ then $0.94 < d^*/\widehat{d}^* < 1.25$ and $\widehat{d}^*$ has to be chosen between 47 and 62 cm.

We fix now $\|\delta\mathbf{p}_0\| = 0.1763$ (which corresponds to a cone with a 10 dg angle) and again $\delta\phi = 5$ dg. The upper and lower bounds for ν are plotted

in Figure 13 versus the ratio $\alpha_u/\widehat{\alpha}_u$ on the x axis and versus the ratio $\alpha_v/\widehat{\alpha}_v$ on the y axis. For a common camera with $\alpha_u/\alpha_v = 3/4$, if $\widehat{\alpha}_u/\widehat{\alpha}_v = 4/3$ then

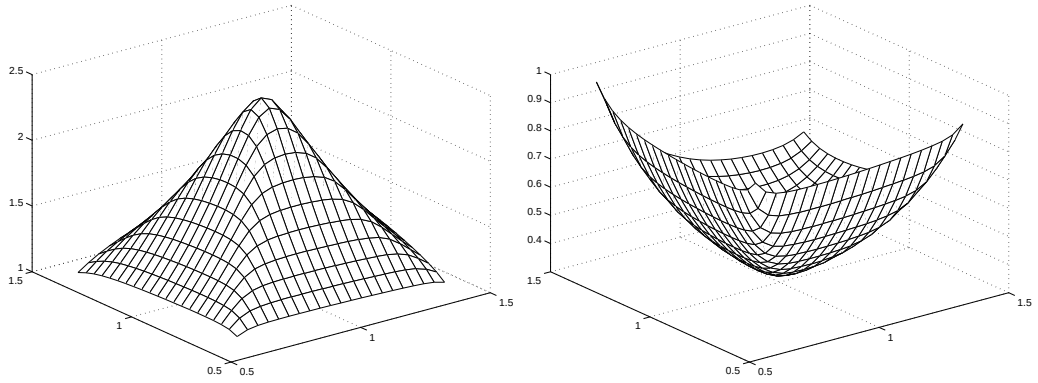


Figure 13: Stability bounds for ν versus $\frac{\alpha_u}{\widehat{\alpha}_u}$ and $\frac{\alpha_v}{\widehat{\alpha}_v}$

$0.62 < \nu \approx d^*/\widehat{d}^* < 1.32$. If the real distance d^* is again 50 cm, the system will asymptotically converge for any initial position in the task space if $\widehat{d}^*$ is chosen between 31 and 66 cm.

Let us finally emphasize that conditions (79-81) are more restrictive than conditions (74-77). When they are ensured, the system will asymptotically converge for any initial camera position in the half space in front of the target. If the task space is smaller (i.e. if the initial camera position is always in a region of the half space), the stability analysis can be made from conditions (74-77) taking into account the restriction on the task space, and thus a larger stability domain will be obtained. More generally, all these conditions are only sufficient, then the convergence can be realized even for larger errors. In the next section, we will see experimentally that the 2D 1/2 visual servoing is robust also in presence of errors in the hand-eye calibration.

5 Simulations results

We present in this section simulation results to compare the motion estimation method we have proposed in Section 2 with the classical ones. The target is composed of 12 points lying either on a plane, or in a volume of $20cm^3$ (in that case, the first 4 points are coplanar). Of course, none knowledge on the structure of the object is used in the motion estimation. For each camera displacement, 100 samples with additive noise on image coordinates (1 pixel standard deviation) where generated. The following errors were computed (where the hat refers to the estimated values):

- Rotational error: The distance between the two rotations $\mathbf{R}$ and $\widehat{\mathbf{R}}$ which is the shortest length of the geodesic starting at $\mathbf{R}$ and ending at $\widehat{\mathbf{R}}$. The shortest length of this geodesic is the rotation angle θ_r of the matrix $\mathbf{R}\widehat{\mathbf{R}}^{-1}$.
- Translational error: The angle θ_t between the normalized vectors $\mathbf{t}/\|\mathbf{t}\|$ and $\widehat{\mathbf{t}}/\|\widehat{\mathbf{t}}\|$.

The median and average deviation result of the 100 samples, and the maximum of the absolute value of each error was then computed. The results are plotted respectively with:

- a triangle for the eight point algorithm using normalized data (EL) (see Section 2.2.3).
- a square for the homography estimation using the virtual parallax algorithm (VP) (see Section 2.2.4).
- a circle for the linear homography matrix estimation algorithm (HL) (see Section 2.2.5).
- a x-mark for the non-linear algorithm (NL) described in [7] and initialized with the EL algorithm results.

The EL and NL algorithms have been tested using the software developed by Z. Zhang[1].

[1] software Fmatrix is available on http://www.inria.fr/robotvis/personnel/zzhang/zzhang-eng.html

5.1 Planar target

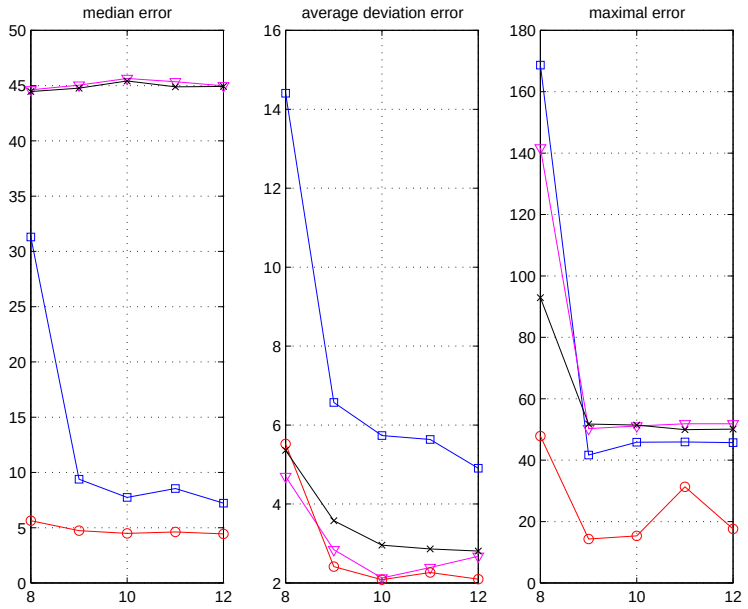
The results depicted in Figure 14 have been obtained using a planar target. The camera displacement was selected as : $\mathbf{t} = \begin{bmatrix} -38 & 38 & 45 \end{bmatrix}^T$ cm and $\mathbf{u}\theta = \begin{bmatrix} 64 & -16 & 138 \end{bmatrix}^T$ dg. As expected, considering a planar target is unfavorable for the algorithms EL and NL based on the fundamental matrix estimation since the epipole is undefined in that case. Important median errors (40 and 55 dg for the rotational and translational errors) are obtained using these algorithms whatever the number of points. Results using our HL algorithm are satisfactory (median error is equal to 5 dg) since the most accurate and stable.

5.2 Non-planar target

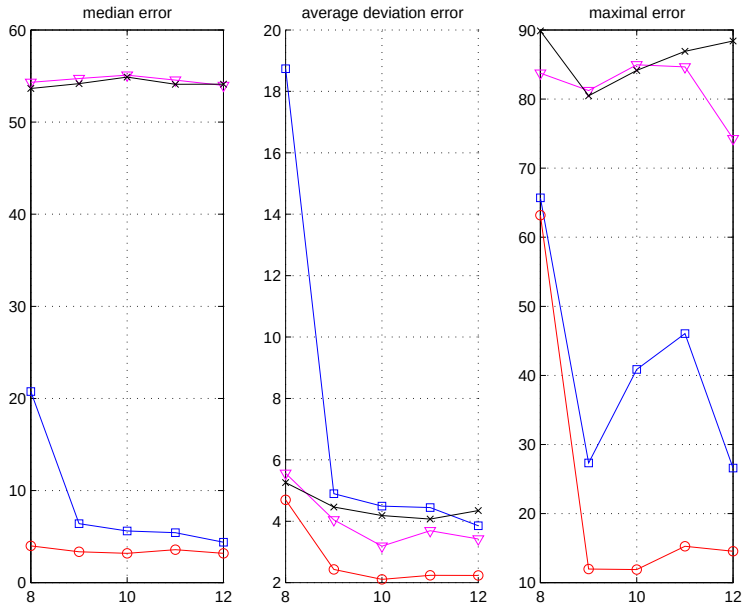
We now use a non-planar target composed by 12 points in a 20 cm^3 volume.

5.2.1 Pure rotation

We here test the robustness of the algorithms to pure rotations. The results for a null rotation $\mathbf{u}\theta = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^T$ dg and a rotation $\mathbf{u}\theta = \begin{bmatrix} 6 & 6 & 6 \end{bmatrix}^T$ dg are shown in Figure 15 and Figure 16 respectively. In this degenerate case for epipolar geometry (since the epipole is undefined in the image), the HL algorithm produces more accurate and robust results than the VP algorithm, since the homography is not estimated jointly to the epipole. As expected, the EL and NL algorithms are less robust than the HL and VP algorithms. This results confirm that, in the singular cases, the use of an homography matrix is preferable to obtain the motion parameters. Since the first 4 points are planar, all algorithms produce worst results using only 8 points (theoretically 5 points, not lying on π, are needed). This explains the large change, especially for the EL and NL algorithms, on the error value from 8 to 9 points.



θ_r (dg) versus number of points



θ_t (dg) versus number of points

Figure 14: Comparison of the motion estimation methods for a planar object

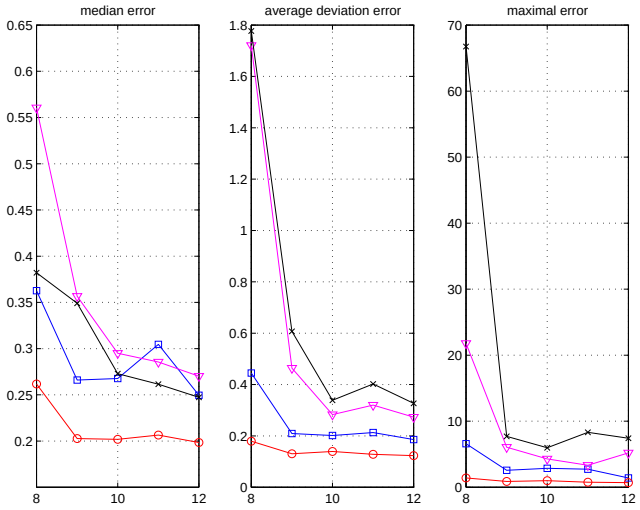


Figure 15: Comparison of the motion estimation methods for a null rotation

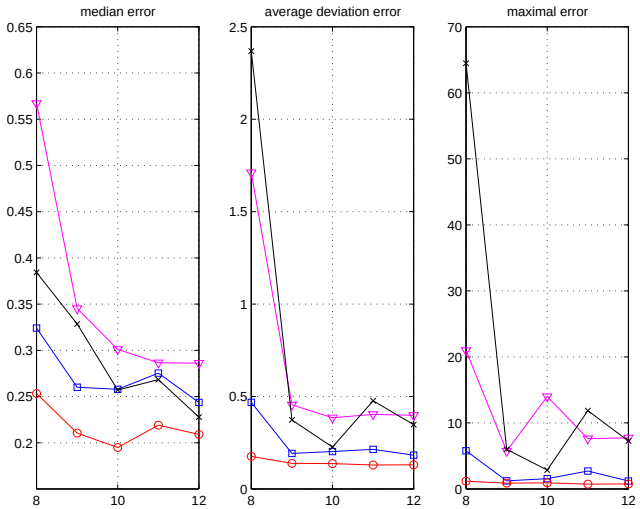


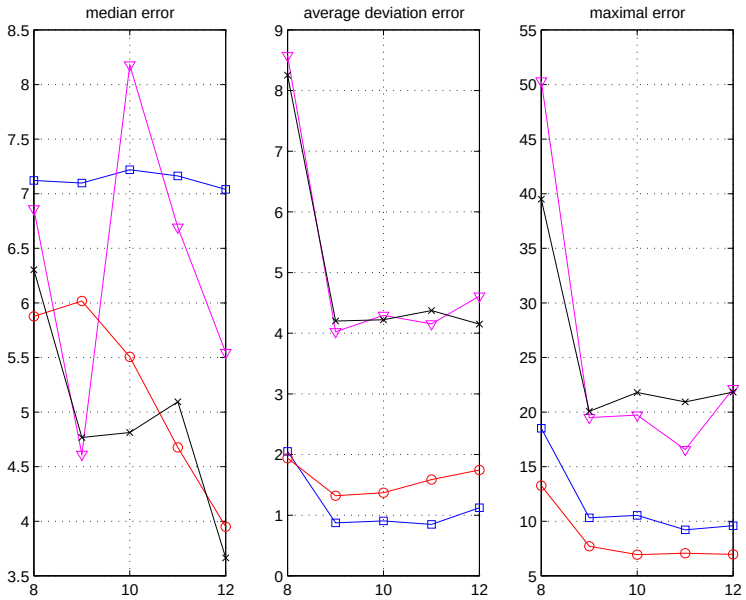
Figure 16: Comparison of the motion estimation methods for a pure rotation

5.2.2 Pure translation

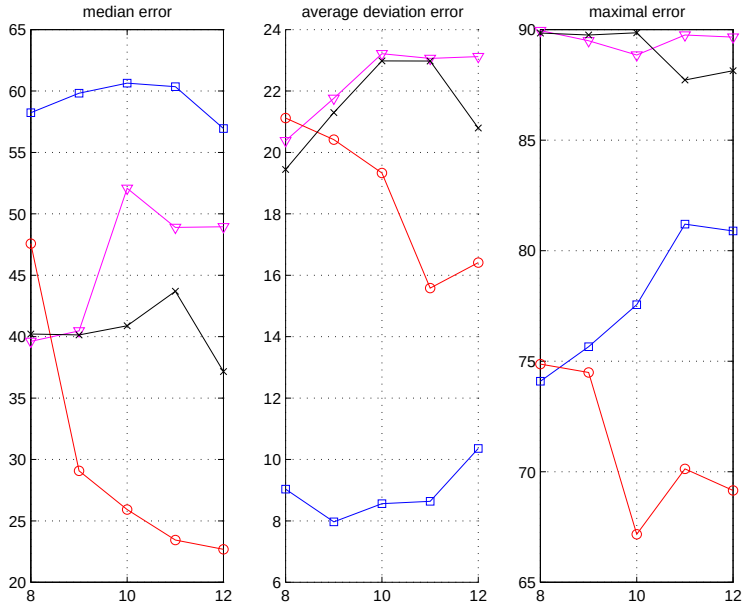
In the following simulation, the displacement was a pure translation $\mathbf{t} = \begin{bmatrix} 6 & 6 & 6 \end{bmatrix}^T$ cm. It is well known that the fundamental matrix estimation is not robust for pure translations [23]. Indeed, as can be seen in Figure 17, the EL and NL algorithms can be unstable providing a maximal translational error of 90 dg. The HL algorithm can reach a 75 dg maximal translational error (which is not satisfactory), but its median and average deviation errors are less than 30 dg. As far as the rotational error is concerned (see Figure 17), even if the median error is similar for all algorithms, the average deviation and the maximal error are divided by factor 2 using the HL and VP algorithms. However, for translational and rotational errors, the bias of the VP algorithm is more important than the HL one.

5.2.3 Generic displacement

Finally, the results obtained with a generic displacement ($\mathbf{t} = \begin{bmatrix} -38 & 38 & 45 \end{bmatrix}^T$ cm and $\mathbf{u}\theta = \begin{bmatrix} 64 & -16 & 138 \end{bmatrix}^T$ dg) are depicted in Figure 18. In that case, the NL algorithm produces, as expected, the best results, but we can note that those obtained using the HL algorithm are really satisfactory in regard to those obtained using the EL method. Finally, the VP algorithm gives here the worst results since the joint estimation of the epipole and of the homography matrix implies important errors on the camera displacement.

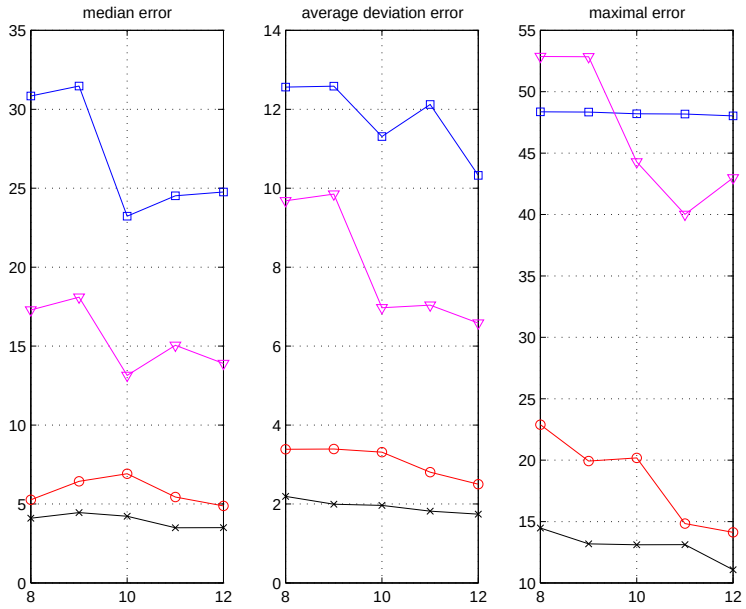


θ_r (dg) versus number of points

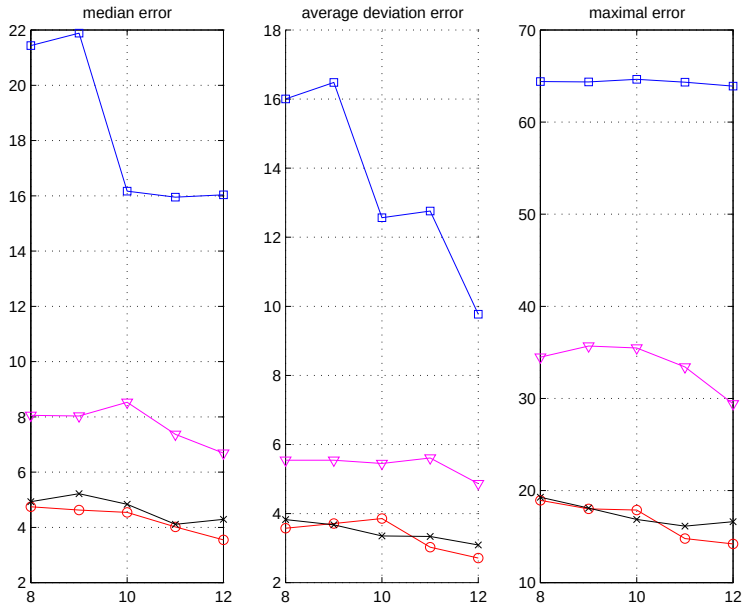


θ_t (dg) versus number of points

Figure 17: Comparison of the motion estimation methods for a translational displacement



θ_r (dg) versus number of points



θ_t (dg) versus number of points

Figure 18: Comparison of the motion estimation methods for a generic displacement

6 Experimental results

6.1 Experiment with a real scene

We now consider a real scene[2] (see Figure 19) and a calibrated eye-in-hand system. In the reported experiment, the camera displacement has been set to $\mathbf{t} = \begin{bmatrix} 14 & 6 & -18 \end{bmatrix}^T$ cm and $\mathbf{u}\theta = \begin{bmatrix} 2.1 & -3.1 & -0.7 \end{bmatrix}^T$ dg. The points (matched using the software Image Matching [3]) in the two images were numbered from 1 to 28 (see Figure 19a and Figure 19b). The first 3 points were chosen by hand as reference points for the change of projective coordinates.

The errors θ_r and θ_t versus number of points, are depicted in Figure 19c and Figure 19d respectively. On the whole, the NL algorithm gives better results than the EL algorithm (surprisingly except for 13 points). According to the simulation results, the HL algorithm produces more accurate results than the EL and the VP algorithms. However, in this experiment the VP algorithm results are not so far from the EL results. It is quite surprising that the HL method gives more accurate results than the NL method. This is probably due to the fact that the camera displacement was not so important in regard of the dimension of the scene. The choice of the three reference points is also important for the HL method, but it can be done automatically by selecting the three points which maximize the surface of the corresponding triangle.

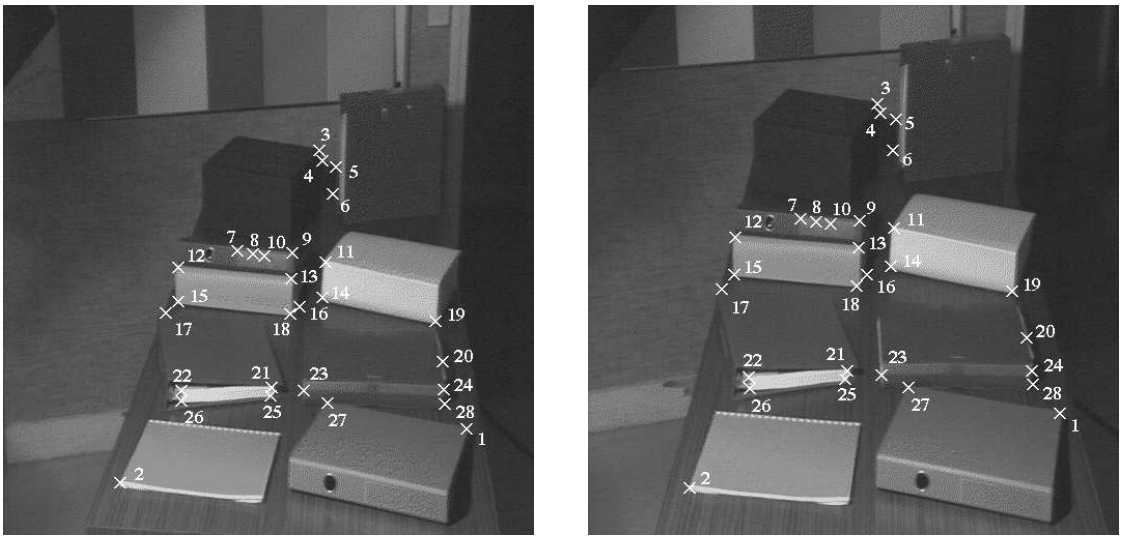
6.2 Experiments with an eye-in-hand robotic system

The HL method has been integrated in the 2D 1/2 visual servoing scheme described in Section 3 and tested on a seven d.o.f. industrial robot MITSUBISHI PA10 (at EDF DER Chatou) and a six d.o.f. Cartesian robot AFMA (at IRISA).

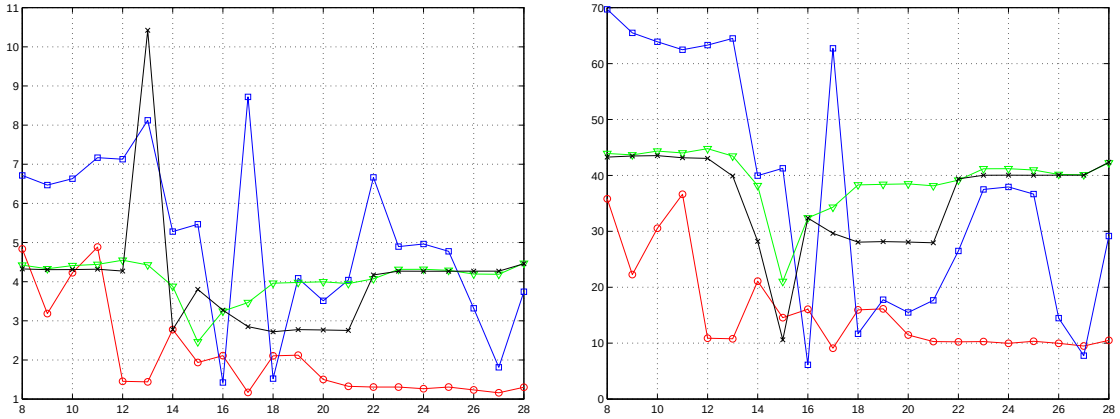
As far as camera calibration is concerned, the pixel and focal lengths given by the camera constructorare used. The image center has been used for the principal point. As far as hand-eye calibration is concerned, the transformation matrix between the camera and the robot end-effector frames is set with an

[2] the images are provided by the Syntim project and they are available on http://www-syntim.inria.fr/syntim/analyse/paires-eng.html at INRIA

[3] the software Image Matching was developed by Zhengyou Zhang and it is available on http://www.inria.fr/robotvis/personnel/zzhang/zzhang-eng.html



(a) first image (b) second image



(c) rotational error (dg) (d) direction of translation error (dg)

Figure 19: Comparison of the motion estimation methods for a real scene

accuracy to within 1 cm for translation and 5 dg for rotation. The target was a black board with twelve white marks on three parallel planes (see Figure 20). The extracted visual features are the image coordinates of the center of gravity of each mark. With such simple images, the control loop can easily be carried out at video rate.

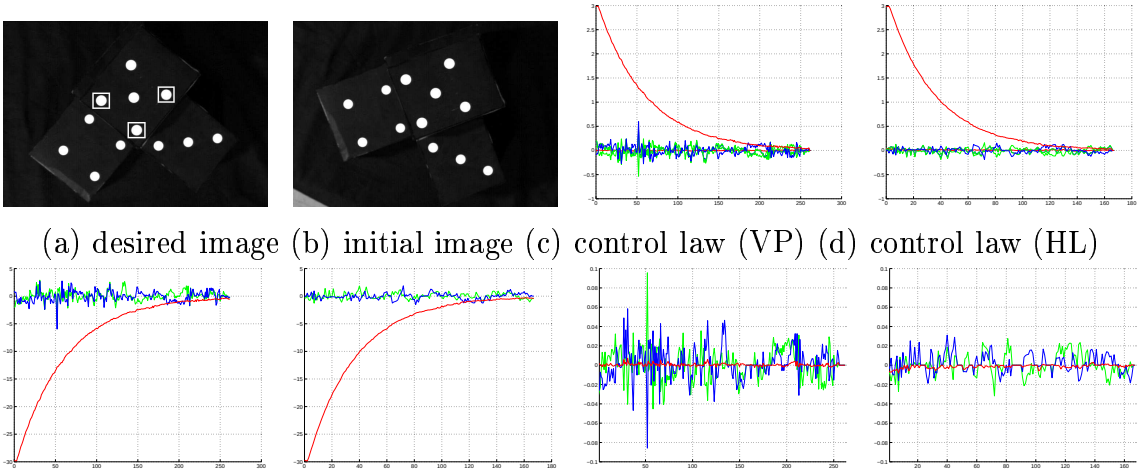
For large camera displacements, as the considered one, points matching between current and desired images is an important computer vision problem which here is supposed to be already solved. This problem also occurs for image-based visual servoing, and similarly for position-based visual servoing since points matching between the image and the 3D model of the target is needed in that case.

In order to prove the validity of the homography estimation, even in non optimal conditions, the three reference points were not taken spread in the image (see Figure 20a where a square has been superimposed around each reference point).

In the following experiments, the NL method has not been tested since it is not able to provide results at video-rate. The EL method has also not been implemented. From the simulation results described in the previous section, very unstable results can be expected for camera motion such that the epipole is undefined, which unfortunately occurs when the camera reaches it desired position. For this reason, only the VP and HL methods were tested. However, except near the desired position, the EL method would provide intermediary results between the VP and the HL methods.

6.2.1 Pure Rotation

The results of the 2D 1/2 visual servoing, obtained performing a rotation of -30 dg around the z axis, are illustrated in Figure 20. The HL and VP algorithms produce good results even if the epipole is undefined. However, it can be observed that the rotation (Figure 20f) and the scaled translation (Figure 20h) estimated using the HL algorithm are more robust than the ones estimated using the VP algorithm (see Figure 20e and Figure 20g). This implies a more stable control law (see Figure 20c and Figure 20d).



(a) desired image (b) initial image (c) control law (VP) (d) control law (HL)

(e) rotation (VP) (f) rotation (HL) (g) translation (VP) (h) translation (HL)

Figure 20: Results of the 2D 1/2 visual servoing for a rotational camera displacement

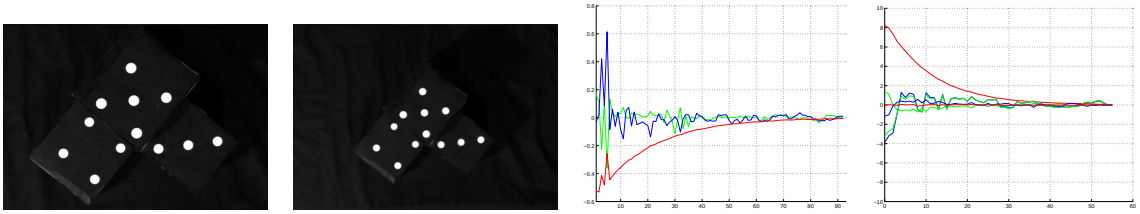
6.2.2 Pure Translation

In this second experiment, the camera displacement was a pure translation such that the epipole coincides with a reference point in the image ($\mathbf{e} = \mathbf{m}_1$) . The obtained results are depicted in Figure 21 (organized as Figure 20). Let us remark that even if the HL algorithm is more robust and stable, the 2D 1/2 visual servoing converges also if the VP algorithm is used, thanks to the used closed-loop control scheme. As can be seen on the plots, from iteration 0 to 5 the VP algorithm is very unstable since it is near its singularity. As can be seen in Figure 21c (VP) and Figure 21d (HL), the estimation of the homography matrix reflects on the computed control law which is thus more stable using the HL method.

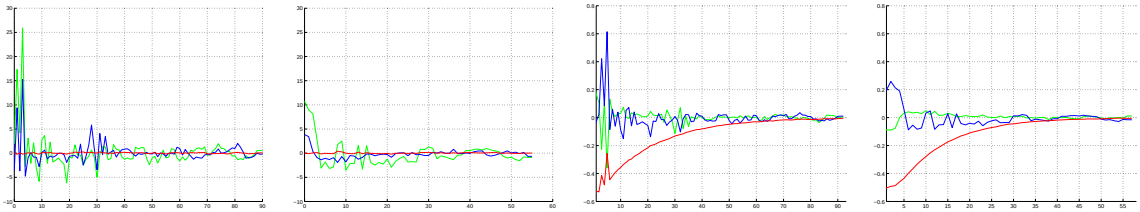
6.2.3 Generic camera displacement

In this last experiment (see Figure 22), a generic camera displacement is performed: $\mathbf{t} = \begin{bmatrix} -1.3 & 55.2 & 4.1 \end{bmatrix}^T$ cm and $\mathbf{u}\theta = \begin{bmatrix} 36.2 & -17.2 & 48.4 \end{bmatrix}^T$ dg. Once again and according to the simulation results, the HL algorithm produces more

robust results than the VP algorithm (see the output control law in Figure 22d and in Figure 22c respectively).

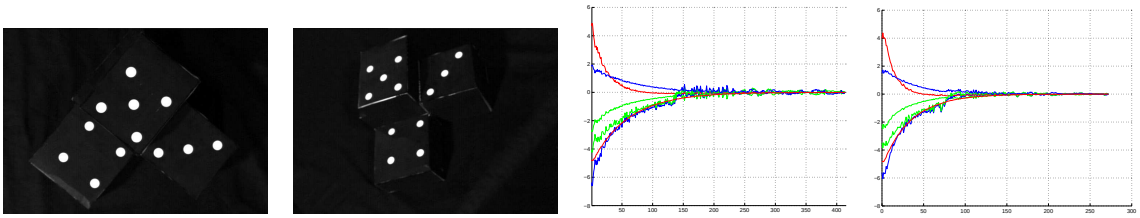


(a) desired image (b) initial image (c) control law (VP) (d) control law (HL)



(e) rotation (VP) (f) rotation (HL) (g) translation (VP) (h) translation (HL)

Figure 21: Results of the 2D 1/2 visual servoing for a translational camera displacement



(a) desired image (b) initial image (c) control law (VP) (d) control law (HL)

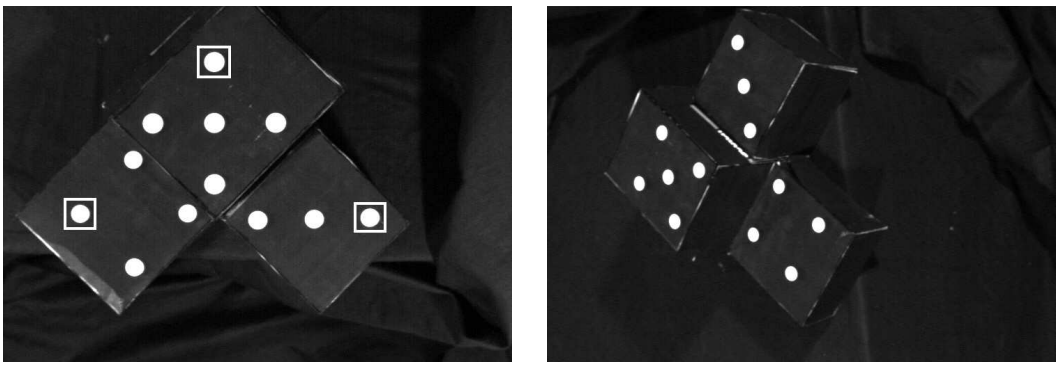
Figure 22: Results of the 2D 1/2 visual servoing for a generic displacement

For example, using the HL algorithm, the estimated rotational displacement, from the initial to desired camera poses, was $\overline{\mathbf{r}} = \begin{bmatrix} 34.8 & -14.9 & 48.3 \end{bmatrix}^T$ dg. Similarly, the estimated direction of translation was $\overline{\mathbf{t}}/\|\overline{\mathbf{t}}\| = \begin{bmatrix} -0.04 & 0.99 & 0.04 \end{bmatrix}^T$ (while the real direction of translation was $\mathbf{t}/\|\mathbf{t}\| = \begin{bmatrix} -0.02 & 0.99 & 0.07 \end{bmatrix}^T$).

The algorithm is thus quite precise (maximal rotational error is around 2 dg, as well as the angle error on the direction of translation) despite the coarse calibration which has been used.

6.3 Comparison between visual servoing schemes

We now present the results obtained using the different control vectors described in Section 3. For these experiments, the HL algorithm has been used since it provides more accurate and stable results. The images corresponding to the desired and initial camera position are given in Figure 23a and 23b respectively. Contrarily to the previous experiments, the three reference points were taken spread in the image (see Figure 23a where a square has been superimposed around each reference point) in order to improve the motion estimation. Furthermore, d^* has been set to 50 cm while its real value is equal to 60 cm.



(a) (b)

Figure 23: Images of the target for the desired (a) and the initial (b) position

6.3.1 3D visual servoing

In this experiment, the camera position is controlled in the Cartesian space. The estimated rotation and translation are given in Figure 24a and Figure 24b respectively. As can be seen on the plots, the estimated rotation and translation do not diverge. However, as can be seen in Figure 24d, the target gets out of the camera field of view (for security reasons, the control scheme is stop-

ped when at least one of the twelve points is no more visible in the image). This failure is encountered because, using this scheme, there is absolutely no control in the image. The probability of failure increases considerably when a bad camera calibration is used or in presence of robot calibration errors.

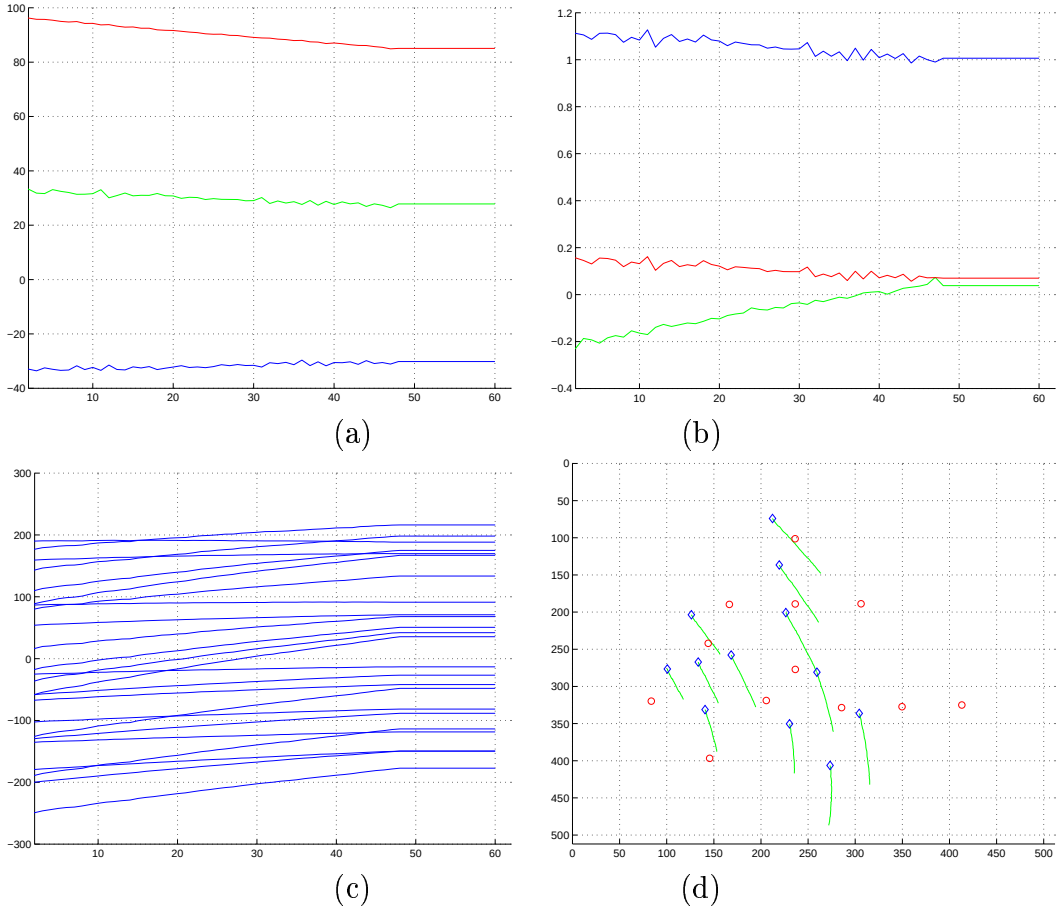
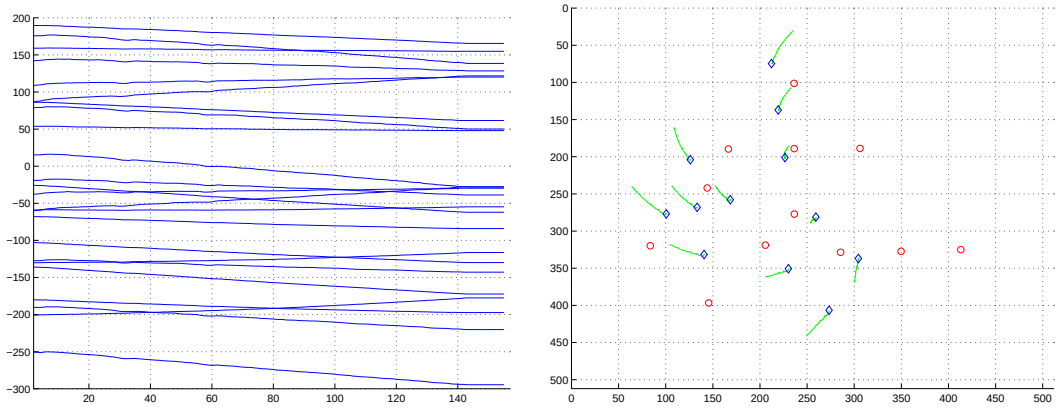


Figure 24: Results of the 3D visual servoing: rotation $\mathbf{u}\theta$ (dg) (a) and scaled translation $\mathbf{t}/d^*$ (b) versus iteration number; error on the coordinates of the image features (pixels) versus iteration number (c) and trajectory of target points in the image (d)

6.3.2 2D visual servoing

In this experiment, the camera is fully controlled using classical image-based visual servoing. A great robustness with respect to calibration errors can thus be expected. However, the camera trajectory in the Cartesian space is no more controlled which, once again, causes the failure of the servoing, since, as can be seen in Figure 25, one target point gets out of the image. This is due to the too important camera displacement from initial to desired poses.



(a) (b)

Figure 25: Results of the 2D visual servoing: error on the coordinates of the image features (pixels) versus iteration number (a) and trajectory of target points in the image (b)

6.3.3 2D 1/2 visual servoing

We now present the results obtained using the 2D 1/2 visual servoing scheme presented in Section 3.3.2. The error on $\mathbf{p}_e - \mathbf{p}_e^*$ is plotted in Figure 26a, the estimated rotation is plotted in Figure 26b and the computed control law is given in Figure 26c and Figure 26d. We can now observe the convergence of the task function towards 0. Finally, the error on the image coordinates of each target point is given in Figure 26e and the corresponding trajectory in the image is given in Figure 26f. The reference point trajectory can be easily identified since it looks like a straight line in the image. We can note the

convergence of the coordinates to their desired values, which demonstrates the correct realization of the task.

6.4 Robustness with camera and hand-eye coarse calibration

In order to test the robustness of our approach with respect to bad calibration, a supplementary error on the camera intrinsic parameters (20%) is added, as well as on the translation (5 cm on each axis) and on the rotation (5 dg on each axis) of the transformation matrix between the camera and the robot end-effector. The obtained results are given in Figure 27. As can be seen in Figure 27a, the convergence of the error is no more perfectly exponential. This is due to the bad calibration of the camera and the rough approximation of d^* (which has no influence using a coarse calibration). However, even in this worse case, we can note the stability and the robustness of the control law. Contrarily to the previous experiment, the trajectory of the reference point in the image is no more a straight line since the camera is bad calibrated as well as the homogeneous transformation matrix between the camera and the robot end-effector frame. However, the convergence of the coordinates to their desired values demonstrates the correct realization of the task.

7 Conclusion

Simulation and experimental results confirm that recovering the motion parameters from the homography matrix gives better results than recovering the motion from the fundamental matrix. This is due to the fact that the complexity of the problem is reduced since the homography is estimated separately from the epipole. Simulations show that good results can be obtained with few points (12), and adding more points does not increase considerably the precision of the algorithm. The homography matrix estimation can be useful in all applications where scaled 3D reconstruction is useful. In our case, the motion parameters extracted from the homography matrix have been used to design a new approach to vision-based robot control which presents many advantages with respect to the classical position-based and image-based visual

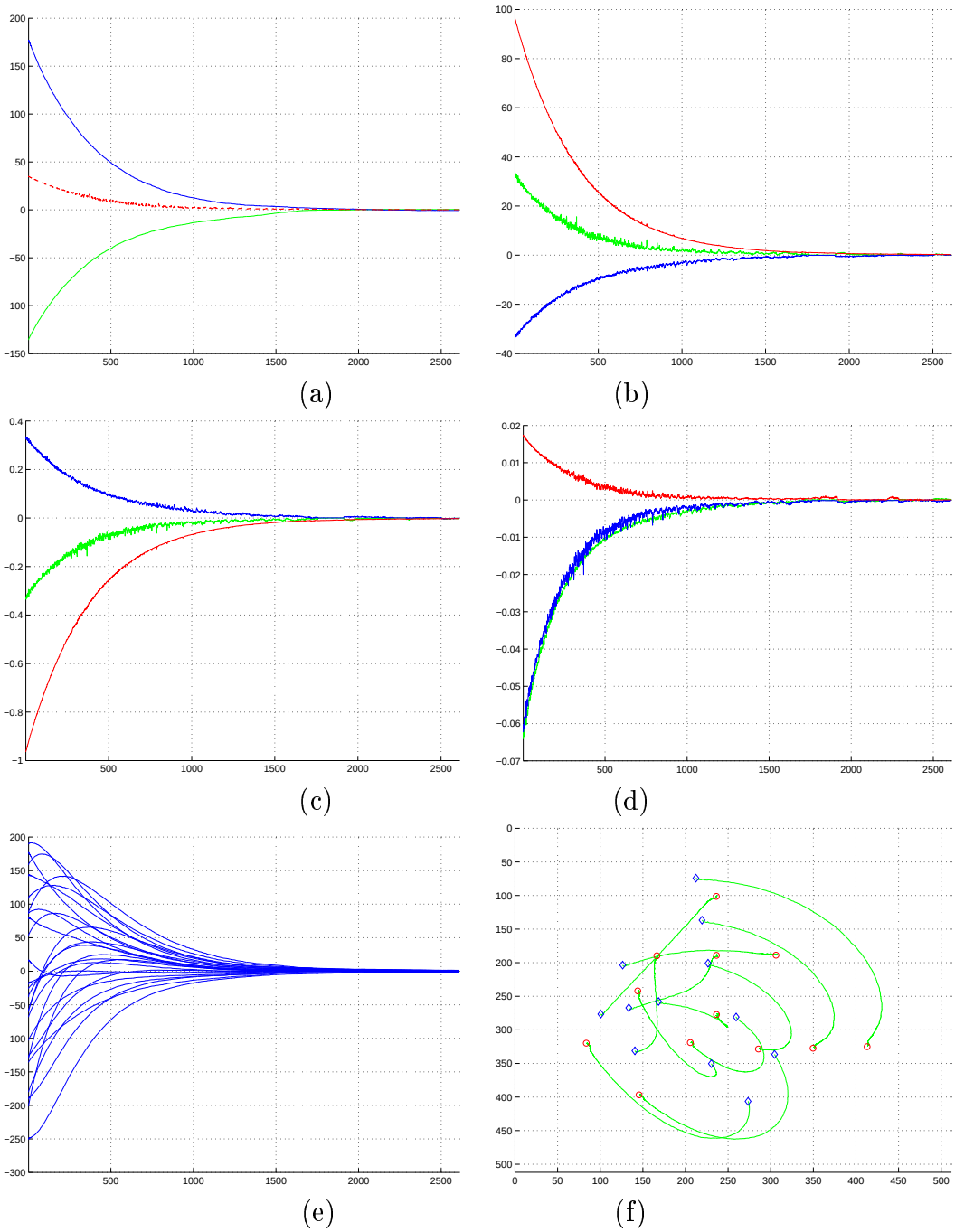


Figure 26: Results of the 2D 1/2 visual servoing: error on the image (pixels) and z (%) coordinates (a), rotation $\mathbf{u}\theta$ (dg) (b), rotational velocity Ω (dg/s) (c) and translational velocity $\mathbf{V}$ (cm/s) (d) versus iteration number; error on the coordinates of the image features (pixels) versus iteration number (e) and trajectory of target points in the image (f)

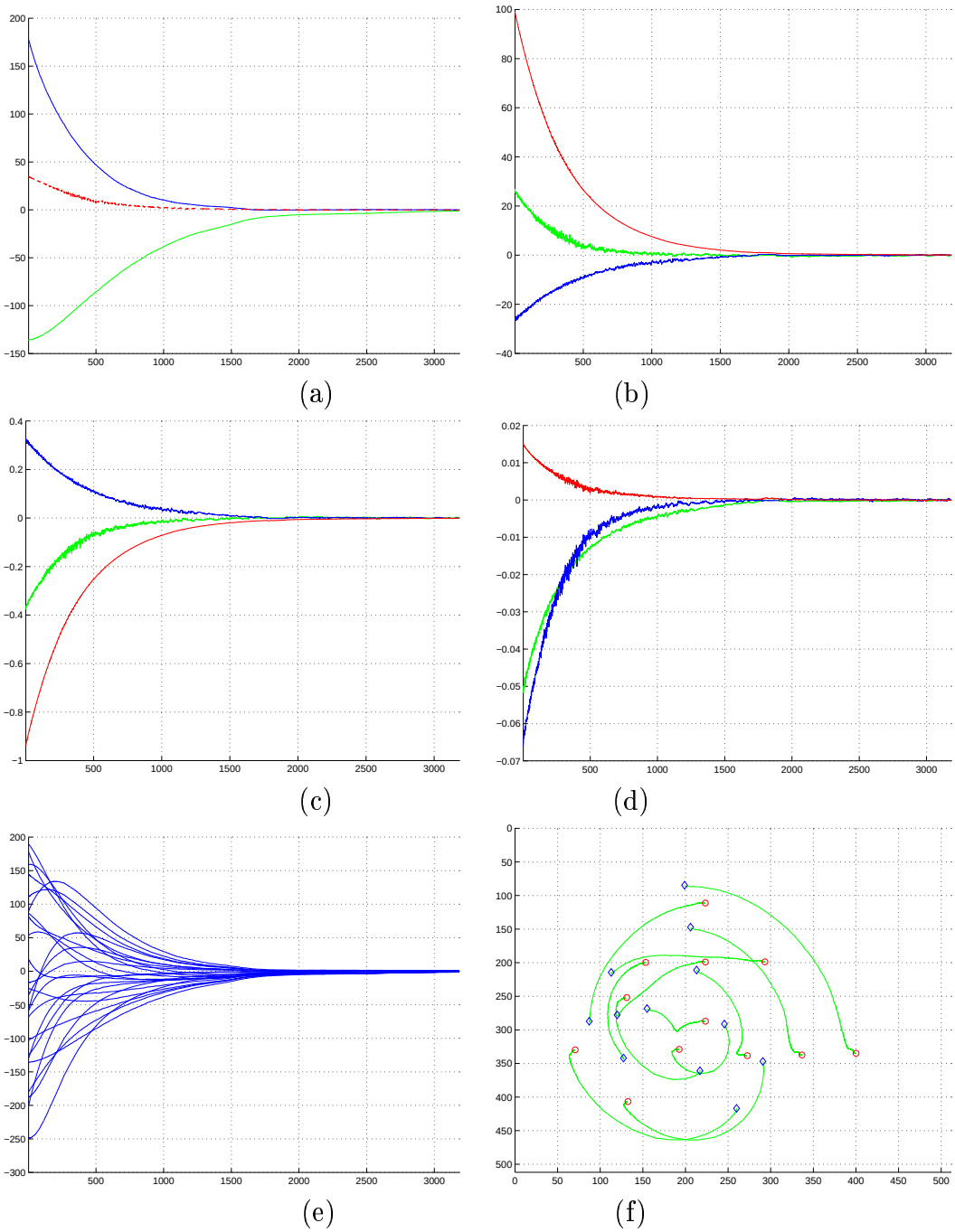


Figure 27: Results of the 2D 1/2 visual servoing with camera and hand-eye coarse calibration: error on image coordinates (pixels) and z (%) (a), $\mathbf{u}\theta$ (dg) (b), rotational velocity Ω (dg/s) (c) and translational velocity $\mathbf{V}$(cm/s) (d) versus iteration number; error on pixels coordinates (e) and trajectory in the image (f) of the target points.

servoings. This method does not need any 3D target model, nor a precise camera calibration and presents very interesting decoupling and stability properties. Thanks to its simple structure, analytical results on its robustness can be obtained. The necessary and sufficient conditions for local asymptotic stability and sufficient conditions for global asymptotic stability in presence of camera calibration errors have been obtained. Experimental results show the validity of our approach and its robustness not only with respect to camera calibration but also to hand-eye calibration errors. Future work will be devoted to the coupling of 2D 1/2 visual servoing with real objects and complex images.

Acknowledgements

This work was supported by the national French Company of Electricity Power: EDF. We are grateful to the team manager and the researchers of the Teleoperation/Robotics group, at DER Chatou, for their participation and help. We are also particularly grateful to Radu Horaud and Gabriella Csurka for their interest in this work, comments and discussions which have allowed us to improve the quality of Section 2.

Appendix A

The j-row of the coefficient matrix $\mathbf{C}_{\tilde{h}}$ (see equation (30)) can be written in function of the image points coordinates as follows:

$$
\begin{aligned}
c_{1j} &= w_i w_j v_k u_k^* (u_j^* v_i^* - u_i^* v_j^*) + w_i w_k v_j u_j^* (u_i^* v_k^* - u_k^* v_i^*) + w_j w_k v_i u_i^* (u_k^* v_j^* - u_j^* v_k^*) \\
c_{2j} &= w_i w_j u_k v_k^* (u_i^* v_j^* - u_j^* v_i^*) + w_i w_k u_j v_j^* (u_k^* v_i^* - u_i^* v_k^*) + w_j w_k u_i v_i^* (u_j^* v_k^* - u_k^* v_j^*) \\
c_{3j} &= v_i v_k w_j u_j^* (u_i^* w_k^* - u_k^* w_i^*) + v_i v_j w_k u_k^* (u_j^* w_i^* - u_i^* w_j^*) + v_j v_k w_i u_i^* (u_k^* w_j^* - u_j^* w_k^*) \\
c_{4j} &= u_i u_k w_j v_j^* (v_i^* w_k^* - v_k^* w_i^*) + u_i u_j w_k v_k^* (v_j^* w_i^* - v_i^* w_j^*) + u_j u_k w_i v_i^* (v_k^* w_j^* - v_j^* w_k^*) \\
c_{5j} &= v_j v_k u_i w_i^* (u_j^* w_k^* - u_k^* w_j^*) + v_i v_k u_j w_j^* (u_k^* w_i^* - u_i^* w_k^*) + v_i v_j u_k w_k^* (u_i^* w_j^* - u_j^* w_i^*) \\
c_{6j} &= u_j u_k v_i w_i^* (v_j^* w_k^* - v_k^* w_j^*) + u_i u_k v_j w_j^* (v_k^* w_i^* - v_i^* w_k^*) + u_i u_j v_k w_k^* (v_i^* w_j^* - v_j^* w_i^*) \\
c_{7j} &= u_i v_k w_j (u_k^* v_j^* w_i^* - u_j^* v_i^* w_k^*) + u_k v_i w_j (u_j^* v_k^* w_i^* - u_i^* v_j^* w_k^*) + u_i v_j w_k (u_k^* v_i^* w_j^* - u_j^* v_k^* w_i^*) + \\
&+ u_j v_i w_k (u_i^* v_k^* w_j^* - u_k^* v_j^* w_i^*) + u_k v_j w_i (u_j^* v_i^* w_k^* - u_i^* v_k^* w_j^*) + u_j v_k w_i (u_i^* v_j^* w_k^* - u_k^* v_i^* w_j^*)
\end{aligned}
$$

Appendix B

Proof of theorem 2 *Let us suppose that conditions (69) are verified. As already stated, this is not a constraining assumption in practice, but has to be done for the following demonstration since such conditions are necessary for the local asymptotic stability of the system, and thus are also necessary (but not sufficient) for the global asymptotic stability of the system. It is well known that a sufficient condition for global asymptotic stability of the system (66) is the positiveness of matrix* $\mathbf{Q}$*. This matrix is positive if and only if its symmetric part* $\mathbf{S} = (\mathbf{Q} + \mathbf{Q}^T)/2$ *is positive. Denoting an element of* $\mathcal{S}$ *as* $\mathbf{e} = \left[\begin{array}{cc} \mathbf{e}_1^T & \mathbf{e}_2^T \end{array} \right]^T$*, matrix* $\mathbf{S}$ *is positive if and only if:*

$$\frac{1}{2}\mathbf{e}_1^T\mathbf{S}_{11}\mathbf{e}_1 + \frac{1}{2}\mathbf{e}_2^T\mathbf{S}_{22}\mathbf{e}_2 + \mathbf{e}_1^T\mathbf{S}_{12}\mathbf{e}_2 > 0 \ , \qquad \forall \mathbf{e} \neq 0 \tag{96}$$

The matrices $\mathbf{S}_{11}$ *and* $\mathbf{S}_{22}$ *have to be positive since if* $\mathbf{e}_1 = 0$ *the inequality becomes* $\mathbf{e}_2^T\mathbf{S}_{22}\mathbf{e}_2 > 0$ $(\forall \mathbf{e}_2 \neq 0)$*, and if* $\mathbf{e}_2 = 0$ *the inequality becomes* $\mathbf{e}_1^T\mathbf{S}_{11}\mathbf{e}_1 > 0$ $(\forall \mathbf{e}_1 \neq 0)$*. We now determine the conditions for their positiveness and, in order to ensure inequality (96), a lower bound for* $\mathbf{e}_1^T\mathbf{S}_{11}\mathbf{e}_1$ *and* $\mathbf{e}_2^T\mathbf{S}_{22}\mathbf{e}_2$*, and an upper bound for* $\mathbf{e}_1^T\mathbf{S}_{12}\mathbf{e}_2$ *will be also found.*

- *Consider first matrix* $\mathbf{S}_{11}$*. We have:*

$$\mathbf{S}_{11} = \nu \left[\begin{array}{cc} \delta\mathbf{A}_{11} + \delta\mathbf{A}_{11}^T & \delta\mathbf{p} \\ \delta\mathbf{p}^T & 2 \end{array} \right] \tag{97}$$

where $\delta\mathbf{p} = \left[\begin{array}{cc} \widehat{x} - x & \widehat{y} - y \end{array} \right]^T$*. A necessary condition for definite positiveness of matrix* $\mathbf{S}_{11}$ *is* $\nu = \frac{\widehat{d}^*}{d^*}\|\delta\mathbf{A}^{-T}\mathbf{n}^*\| > 0$ *which is obviously verified. Furthermore, matrix* $\mathbf{S}_{11}$ *is positive only if matrix* $\delta\mathbf{A}_{11} + \delta\mathbf{A}_{11}^T$ *is positive. This symmetric matrix is positive if and only if its eigenvalues are positive, and thus if and only if its minimum eigenvalue, denoted* σ*, is positive:*

$$\sigma = \frac{\alpha_u}{\widehat{\alpha}_u} + \frac{\alpha_v}{\widehat{\alpha}_v} - \sqrt{\left(\frac{\alpha_u}{\widehat{\alpha}_u} - \frac{\alpha_v}{\widehat{\alpha}_v}\right)^2 + \left(\frac{\alpha_{uv}}{\widehat{\alpha}_u} - \frac{\widehat{\alpha}_{uv}}{\widehat{\alpha}_u}\frac{\alpha_v}{\widehat{\alpha}_v}\right)^2} > 0 \tag{98}$$

We here find condition (74). Let us now suppose that this condition is ensured. Matrix $\mathbf{S}_{11}$ *can thus be bounded as follows:*

$$\mathbf{e}_1^T\mathbf{S}_{11}\mathbf{e}_1 \geq \mathbf{e}_1^T\mathbf{S}_{11}'\mathbf{e}_1 \ , \qquad \forall \mathbf{e}_1 \tag{99}$$

where:

$$\mathbf{S}_{11}^{'} = \nu \begin{bmatrix} \sigma \mathbf{I}_2 & \delta \mathbf{p} \\ \delta \mathbf{p}^T & 2 \end{bmatrix} \tag{100}$$

If matrix $\mathbf{S}_{11}^{'}$ *is positive then matrix* $\mathbf{S}_{11}$ *is also positive. Remembering that* $\|\delta \mathbf{p}\| = \|(\mathbf{I} - \delta \mathbf{A})\mathbf{p}\|$*, the minimum eigenvalue of matrix* $\mathbf{S}_{11}^{'}$ *is given by:*

$$\sigma_1^{'} = \nu(\sigma + 1 - \sqrt{(\sigma - 1)^2 + \|\delta \mathbf{p}\|^2}) = \nu(\sigma + 1 - \sqrt{(\sigma - 1)^2 + \|(\mathbf{I} - \delta \mathbf{A})\mathbf{p}\|^2}) \tag{101}$$

Once again, matrix $\mathbf{S}_{11}^{'}$ *is positive if its minimum eigenvalue is positive. Since* $\|(\mathbf{I} - \delta \mathbf{A})\mathbf{p}\|^2 \leq \|(\mathbf{I} - \delta \mathbf{A})\|^2 \|\mathbf{p}\|^2$ *and* $\|\mathbf{p}\|^2 = (1 + \gamma^2) \leq (1 + \overline{\gamma}^2)$ *(where* $\gamma = \sqrt{x^2 + y^2}$ *and* $\overline{\gamma}$ *is its maximum value),* $\sigma_1^{'}$ *can be bounded as follow:*

$$\sigma_1^{'} \geq \nu(\sigma + 1 - \sqrt{(\sigma - 1)^2 + \|(\mathbf{I} - \delta \mathbf{A})\|^2 (1 + \overline{\gamma}^2)}) \tag{102}$$

Then, since $\nu > 0$*,* $\sigma_1^{'}$ *is positive if:*

$$\sigma_1 = \sigma + 1 - \sqrt{(\sigma - 1)^2 + \|(\mathbf{I} - \delta \mathbf{A})\|^2 (1 + \overline{\gamma}^2)} > 0 \tag{103}$$

which is nothing but condition (75). This means that if $\sigma > 0$ *and* $\sigma_1 > 0$ *then* $\mathbf{S}_{11} > 0$ *and* $\mathbf{e}_1^T \mathbf{S}_{11} \mathbf{e}_1$ *can be bounded as follows:*

$$\mathbf{e}_1^T \mathbf{S}_{11} \mathbf{e}_1 \geq \mathbf{e}_1^T \mathbf{S}_{11}^{'} \mathbf{e}_1 \geq \sigma_1^{'} \|\mathbf{e}_1\|^2 \geq \nu \sigma_1 \|\mathbf{e}_1\|^2 \tag{104}$$

- *Consider now matrix* $\mathbf{S}_{22}$*. We have:*

$$\mathbf{S}_{22} = \mu \begin{bmatrix} \delta \mathbf{A}_{11} + \delta \mathbf{A}_{11}^T & \delta \mathbf{p}_0 \\ \delta \mathbf{p}_0^T & 2 \end{bmatrix} \tag{105}$$

A necessary condition for definite positiveness of matrix $\mathbf{S}_{22}$ *is* $\mu = \frac{1}{\|\delta \mathbf{A} \mathbf{u}\|} > 0$ *which is obviously satisfied. Let us remark that this matrix has the same structure of matrix* $\mathbf{S}_{11}$*, then* $\mathbf{S}_{22}$ *can be bounded as follow:*

$$\mathbf{e}_2^T \mathbf{S}_{22} \mathbf{e}_2 \geq \mathbf{e}_2^T \mathbf{S}_{22}^{'} \mathbf{e}_2 \ , \qquad \forall \mathbf{e}_2 \tag{106}$$

where:

$$\mathbf{S}'_{22} = \mu \begin{bmatrix} \sigma \mathbf{I}_2 & \delta \mathbf{p}_0 \\ \delta \mathbf{p}_0^T & 2 \end{bmatrix} \tag{107}$$

Once again, matrix $\mathbf{S}'_{22}$ *is positive if its minimum eigenvalue is positive. The minimum eigenvalue of matrix* $\mathbf{S}'_{22}$ *is:*

$$\sigma'_2 = \mu(\sigma + 1 - \sqrt{(\sigma-1)^2 + \|\delta \mathbf{p}_0\|^2}) \tag{108}$$

Then, since $\mu > 0$, σ'_2 *is positive if:*

$$\sigma_2 = \sigma + 1 - \sqrt{(\sigma-1)^2 + \|\delta \mathbf{p}_0\|^2} > 0 \tag{109}$$

which gives condition (76). Therefore, if $\sigma > 0$ *and* $\sigma_2 > 0$ *then* $\mathbf{S}_{22} > 0$ *and* $\mathbf{e}_2^T \mathbf{S}_{22} \mathbf{e}_2$ *can be bounded as follows:*

$$\mathbf{e}_2^T \mathbf{S}_{22} \mathbf{e}_2 \geq \mathbf{e}_2^T \mathbf{S}'_{22} \mathbf{e}_2 \geq \sigma'_2 \|\mathbf{e}_2\|^2 \geq \mu \sigma_2 \|\mathbf{e}_2\|^2 \tag{110}$$

- *Consider now matrix* $\mathbf{S}_{12}$*. We have:*

$$\mathbf{S}_{12} = \mu \mathbf{L}_v ([\mathbf{p} - \nu \widehat{\mathbf{p}}]_\times) \delta \mathbf{A} \tag{111}$$

The l_2 *norm of* $\mathbf{S}_{12}$ *can be bounded as follow:*

$$\|\mathbf{S}_{12}\| \leq \mu \|\mathbf{L}_v\| \, \|[\mathbf{p} - \nu \widehat{\mathbf{p}}]_\times\| \|\delta \mathbf{A}\| \tag{112}$$

and, since the norm of a skew-symmetric matrix associated to a vector is equal to the vector norm, we obtain:

$$\begin{array}{rcl} \|\mathbf{S}_{12}\| & \leq & \mu \|\mathbf{L}_v\| \, \|\delta \mathbf{A}\| \, \|\mathbf{p} - \nu \widehat{\mathbf{p}}\| \\ & \leq & \mu \|\mathbf{L}_v\| \, \|\delta \mathbf{A}\| \, \|(1-\nu)\mathbf{p} + \nu(\mathbf{p} - \widehat{\mathbf{p}})\| \\ & \leq & \mu \|\mathbf{L}_v\| \, \|\delta \mathbf{A}\| \, \|(1-\nu)\mathbf{p} + \nu(\mathbf{I} - \delta \mathbf{A})\mathbf{p})\| \\ & \leq & \mu \|\mathbf{L}_v\| \, \|\delta \mathbf{A}\| \, \|\mathbf{p}\| (\nu \|\mathbf{I} - \delta \mathbf{A}\| + |1-\nu|) \end{array} \tag{113}$$

By noting:

$$g(\gamma) = \|\mathbf{L}_v\| \|\mathbf{p}\| = \sqrt{\frac{2 + \gamma^2 + \sqrt{\gamma^2(\gamma^2+4)}}{2}(1+\gamma^2)} \tag{114}$$

we can note that function $g(\gamma)$ is monotonic. Therefore, if $\gamma \leq \overline{\gamma}$ then $g(\gamma) \leq g(\overline{\gamma})$. We thus obtain:

$$\|\mathbf{S}_{12}\| \leq \mu(\nu\|\mathbf{I} - \delta\mathbf{A}\| + |1-\nu|)\|\delta\mathbf{A}\|g(\overline{\gamma}) \tag{115}$$

We can now find a last condition for positiveness of matrix $\mathbf{S}$. The inequality (96) holds if:

$$\frac{1}{2}\mathbf{e}_1^T\mathbf{S}_{11}\mathbf{e}_1 + \frac{1}{2}\mathbf{e}_2^T\mathbf{S}_{22}\mathbf{e}_2 > |\mathbf{e}_1^T\mathbf{S}_{12}\mathbf{e}_2| \tag{116}$$

From (104) and (110), we have:

$$\frac{1}{2}\mathbf{e}_1^T\mathbf{S}_{11}\mathbf{e}_1 + \frac{1}{2}\mathbf{e}_2^T\mathbf{S}_{22}\mathbf{e}_2 \geq \frac{1}{2}\nu\sigma_1\|\mathbf{e}_1\|^2 + \frac{1}{2}\mu\sigma_2\|\mathbf{e}_2\|^2 \tag{117}$$

and, from (115), we have:

$$|\mathbf{e}_1^T\mathbf{S}_{12}\mathbf{e}_2| \leq \mu(\nu\|\mathbf{I} - \delta\mathbf{A}\| + |1-\nu|)\|\delta\mathbf{A}\|g(\overline{\gamma})\|\mathbf{e}_1\|\|\mathbf{e}_2\| \tag{118}$$

Then, the inequality (116) holds if:

$$\frac{1}{2}\nu\sigma_1\|\mathbf{e}_1\|^2 + \frac{1}{2}\mu\sigma_2\|\mathbf{e}_2\|^2 > \mu(\nu\|\mathbf{I} - \delta\mathbf{A}\| + |1-\nu|)\|\delta\mathbf{A}\|g(\overline{\gamma})\|\mathbf{e}_1\|\|\mathbf{e}_2\| \tag{119}$$

This inequality has the following solution $\forall\mathbf{e}_1$ and $\forall\mathbf{e}_2$ (and $\mathbf{e} \neq 0$):

$$(\mu(\nu\|\mathbf{I} - \delta\mathbf{A}\| + |1-\nu|)\|\delta\mathbf{A}\|g(\overline{\gamma}))^2 < 4(\frac{1}{2}\nu\sigma_1)(\frac{1}{2}\mu\sigma_2) \tag{120}$$

that finally, can be simply written as:

$$\mu(\nu\|\mathbf{I} - \delta\mathbf{A}\| + |1-\nu|)^2\|\delta\mathbf{A}\|^2 g^2(\overline{\gamma}) < \nu\sigma_1\sigma_2 \tag{121}$$

which is condition (77).

Appendix C

Proof of corollary *Let us first find an upper bound of* $\mu = \frac{1}{\|\delta\mathbf{A}\mathbf{u}\|}$. *The square of the norm of vector* $\delta\mathbf{A}\mathbf{u}$ *is given by:*

$$\|\delta\mathbf{A}\mathbf{u}\|^2 = \mathbf{u}^T\delta\mathbf{A}^T\delta\mathbf{A}\mathbf{u} = \mathbf{u}^T \begin{bmatrix} \delta\mathbf{A}_{11}^T\delta\mathbf{A}_{11} & \delta\mathbf{A}_{11}^T\delta\mathbf{p}_0 \\ \delta\mathbf{p}_0^T\delta\mathbf{A}_{11} & 1 \end{bmatrix} \mathbf{u} \tag{122}$$

Let σ' *be the minimum eigenvalue of* $\delta\mathbf{A}_{11}^T\delta\mathbf{A}_{11}$. *We have:*

$$\sigma' = \frac{1}{2}\left(\frac{\alpha_u^2}{\widehat{\alpha}_u^2} + \frac{\alpha_v^2}{\widehat{\alpha}_v^2} + \left(\frac{\alpha_{uv}}{\widehat{\alpha}_u} - \frac{\widehat{\alpha}_{uv}}{\widehat{\alpha}_u}\frac{\alpha_v}{\widehat{\alpha}_v}\right)^2 - \sqrt{\left(\frac{\alpha_u^2}{\widehat{\alpha}_u^2} + \frac{\alpha_v^2}{\widehat{\alpha}_v^2} + \left(\frac{\alpha_{uv}}{\widehat{\alpha}_u} - \frac{\widehat{\alpha}_{uv}}{\widehat{\alpha}_u}\frac{\alpha_v}{\widehat{\alpha}_v}\right)^2\right)^2 - 4\frac{\alpha_u^2}{\widehat{\alpha}_u^2}\frac{\alpha_v^2}{\widehat{\alpha}_v^2}}\right) \tag{123}$$

We know from the form of $\delta\mathbf{A}_{11}^T\delta\mathbf{A}_{11}$ *that* σ' *is positive. The square of the norm of vector* $\delta\mathbf{A}\mathbf{u}$ *can thus be bounded as follows:*

$$\|\delta\mathbf{A}\mathbf{u}\|^2 \geq \mathbf{u}^T\delta\mathbf{A}'^T\delta\mathbf{A}'\mathbf{u} \tag{124}$$

where:

$$\delta\mathbf{A}'^T\delta\mathbf{A}' = \begin{bmatrix} \sigma'\mathbf{I} & \delta\mathbf{A}_{11}^T\delta\mathbf{p}_0 \\ \delta\mathbf{p}_0^T\delta\mathbf{A}_{11} & 1 \end{bmatrix} \tag{125}$$

The minimum eigenvalue of $\delta\mathbf{A}'^T\delta\mathbf{A}'$ *is:*

$$\frac{1}{2}\left(\sigma' + 1 - \sqrt{(\sigma'-1)^2 + 4\|\delta\mathbf{A}_{11}^T\delta\mathbf{p}_0\|^2}\right) \tag{126}$$

and can be bounded as follows:

$$\frac{1}{2}\left(\sigma' + 1 - \sqrt{(\sigma'-1)^2 + 4\|\delta\mathbf{A}_{11}\delta\mathbf{p}_0\|^2}\right) \geq \frac{1}{2}\left(\sigma' + 1 - \sqrt{(\sigma'-1)^2 + 4\|\delta\mathbf{A}_{11}\|^2\|\delta\mathbf{p}_0\|^2}\right) \tag{127}$$

Consequently, the square of the norm of vector $\delta\mathbf{A}\mathbf{u}$ *can be bounded as follows:*

$$\|\delta\mathbf{A}\mathbf{u}\|^2 \geq \frac{1}{2}\left(\sigma' + 1 - \sqrt{(\sigma'-1)^2 + 4\|\delta\mathbf{A}_{11}\|^2\|\delta\mathbf{p}_0\|^2}\right)\|\mathbf{u}\|^2 \tag{128}$$

Finally, since $\|\mathbf{u}\|^2 = 1$, μ *can be bounded as follow:*

$$\mu = \frac{1}{\|\delta\mathbf{A}\mathbf{u}\|} \leq \frac{\sqrt{2}}{\sqrt{\sigma' + 1 - \sqrt{(\sigma'-1)^2 + 4\|\delta\mathbf{A}_{11}\|^2\|\delta\mathbf{p}_0\|^2}}} \tag{129}$$

Remembering that $\|\delta\mathbf{A}\| \leq \|\delta\mathbf{A}_{11}\| + \sqrt{1 + \|\delta\mathbf{p}_0\|^2}$ *and* $\|\mathbf{I}_3 - \delta\mathbf{A}\| \leq \|\mathbf{I}_2 - \delta\mathbf{A}_{11}\| + \|\delta\mathbf{p}_0\|$, *the left size of condition (77) can be bounded as follows:*

$$\begin{gathered}\mu(\nu\|\mathbf{I} - \delta\mathbf{A}\| + |1-\nu|)^2\|\delta\mathbf{A}\|^2 g^2(\overline{\gamma}) \leq \\ \sqrt{2}\frac{(\nu(\|\mathbf{I} - \delta\mathbf{A}_{11}\| + \|\delta\mathbf{p}_0\|) + |1-\nu|)^2(\|\delta\mathbf{A}_{11}\| + \sqrt{1 + \|\delta\mathbf{p}_0\|^2})^2 g^2(\overline{\gamma})}{\sqrt{\sigma' + 1 - \sqrt{(\sigma'-1)^2 + 4\|\delta\mathbf{A}_{11}\|^2\|\delta\mathbf{p}_0\|^2}}}\end{gathered} \tag{130}$$

Furthermore, condition (75) can be written:

$$\sigma_1 \geq \sigma_3 = (\sigma + 1 - \sqrt{(\sigma-1)^2 + (\|\mathbf{I}_2 - \delta\mathbf{A}_{11}\| + \|\delta\mathbf{p}_0\|)^2(1 + \overline{\gamma}^2)}) > 0 \tag{131}$$

which is condition (80). We can note that, if this inequality is verified then (76) is verified too. Then, condition (76) is no more necessary. The rigth size of condition (77) can be bounded as follows:

$$\nu\sigma_1\sigma_2 \geq \nu\sigma_2\sigma_3 \tag{132}$$

Finally, condition (77) will be verified if:

$$\sqrt{2}\frac{(\nu(\|\mathbf{I} - \delta\mathbf{A}_{11}\| + \|\delta\mathbf{p}_0\|) + |1-\nu|)^2(\|\delta\mathbf{A}_{11}\| + \sqrt{1 + \|\delta\mathbf{p}_0\|^2})^2 g^2(\overline{\gamma})}{\sqrt{\sigma' + 1 - \sqrt{(\sigma'-1)^2 + 4\|\delta\mathbf{A}_{11}\|^2\|\delta\mathbf{p}_0\|^2}}} < \nu\sigma_2\sigma_3 \tag{133}$$

which is condition (81).

References

[1] P. K. Allen, A. Timcenko, Yoshimi B., and P Michelman. Automated tracking and grasping of a moving object with a robotic hand-eye system. *IEEE Trans. on Robotics and Automation*, 9(2):152–165, April 1993.

[2] F. Bensalah and F. Chaumette. Compensation of abrupt motion changes in target tracking by visual servoing. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, IROS'95*, volume 1, pages 181–187, Pittsburgh, Pennsylvania, August 1995.

[3] B. Boufama and R. Mohr. Epipole and fundamental matrix estimation using the virtual parallax property. In *IEEE International Conference on Computer Vision*, pages 1030–1036, Cambridge, USA, 1995.

[4] F. Chaumette. Potential problems of stability and convergence in image-based and position-based visual servoing. In *Workshop on Vision and Control*, Block Island, USA, June 1997.

[5] B. Couapel and K. Bainian. Stereo vision with the use of a virtual plane in the space. *Chinese Journal of Electronics*, 4(2):32–39, April 1995.

[6] 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.

[7] R. Deriche, Z. Zhang, Q.-T. Luong, and O. Faugeras. Robust recovery of the epipolar geometry for an uncalibrated stereo rig. In *European Conference on Computer Vision*, Stockholm, Sweden, 1994.

[8] E. Dombre and W. Khalil. *Modélisation et commande des robots*. Hermes, 1988.

[9] B. Espiau. Effect of camera calibration errors on visual servoing in robotics. In *3rd International Symposium on Experimental Robotics*, Kyoto, Japan, October 1993.

[10] 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.

[11] O. Faugeras. *Three-dimensionnal computer vision: a geometric viewpoint.* MIT Press, Cambridge, Massachusetts, 1993.

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

[13] R. I. Hartley. Estimation of relative camera positions for uncalibrated cameras. In G. Sandini, editor, *Computer Vision - ECCV '92*, volume 588 of *Lecture Notes in Computer Science*, pages 579–587, Santa Margherita Ligure, Italia, May 1992. Springer-Verlag.

[14] R. I. Hartley. In defense of the eight-point algorithm. *IEEE Trans. on PAMI*, 19(6):580–593, June 1997.

[15] K. Hashimoto, editor. *Visual Servoing: Real Time Control of Robot manipulators based on visual sensory feedback*, volume 7 of *World Scientific Series in Robotics and Automated Systems.* World Scientific Press, Singapore, 1993.

[16] D. J. Heeger and A. D. Jepson. Subspace methods for recovering rigid motion i: Algorithm and implementation. *International Journal of Computer Vision*, 7(2):95–117, 1992.

[17] T. S. Huang and O. Faugeras. Some properties of the E matrix in two-view motion estimation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 11(12):1310–1312, December 1989.

[18] 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.

[19] M. Irani and P. Anandan. Parallax geometry of pairs of points for 3d scene analysis. In *European Conference on Computer Vision*, pages 17–30, 1996.

[20] C.P. Jerian and R. Jain. Structure from motion - A critical analysis of methods. *IEEE Trans. on Systems, Man, and Cybernetics*, 21(3):572–588, May/June 1991.

[21] H. C. Longuet-Higgins. A computer algorithm for reconstructing a scene from two projections. *Nature*, 293:133–135, September 1981.

[22] H.C. Longuet-Higgins. The reconstruction of a scene from two projections: configurations that defeat the 8-point algorithm. In *Proceedings of the 1st Conference on Artificial intelligence applications*, pages 395–397, Denver, 1984.

[23] Q.-T. Luong. *Matrice fondamentale et calibration visuelle sur l'environnement: vers une plus grande autonomie des systemes robotiques.* PhD thesis, Univerisité de Paris-sud, centre d'Orsay, 1992.

[24] Q.-T. Luong and O. Faugeras. The fundamental matrix: Theory, algorithms, and stability analysis. *International Journal of Computer Vision*, 17(1):43–75, January 1996.

[25] E. Malis, F. Chaumette, and S. Boudet. Positioning a coarse-calibrated camera with respect to an unknown planar object by 2D 1/2 visual servoing. In *5th IFAC Symposium on Robot Control (SYROCO'97)*, volume 2, pages 517–523, Nantes, France, September 1997.

[26] R. Mohr and B. Triggs. Projective geometry for image analysis. A tutorial given at the International Symposium of Photogrammetry and Remote Sensing, July 1996.

[27] N. P. Papanikolopoulos, P. K. Kosla, and T. Kanade. Visual tracking of a moving target by a camera mounted on a robot: a combination of control and vision. *IEEE Trans. on Robotics and Automation*, 9(1):14–35, February 1993.

[28] C. Samson, M. Le Borgne, and B. Espiau. *Robot Control: the Task Function Approach*, volume 22 of *Oxford Engineering Science Series*. Clarendon Press, Oxford, Royaume Uni, 1991.

[29] A. Shashua and N. Navab. Relative affine structure: canonical model for 3d from 2d geometry and applications. *IEEE Trans. on PAMI*, 18(9):873–883, September 1996.

[30] R. Y. Tsai and T. S. Huang. Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *IEEE Trans. on PAMI*, 6(1):13–27, January 1984.

[31] T. Vieville, C. Zeller, and L. Robert. Using collineations to compute motion and structure in an uncalibrated image sequence. *IJCV*, 20(3):213–242, 1996.

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

[33] C. Wiles and M. Brady. On the appropriateness of camera models. In *Computer Vision - ECCV '96*, volume 2, pages 228–237, April 1996.

[34] 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.

[35] Z. Zhang and A. R. Hanson. Scaled euclidean 3D reconstruction based on externally uncalibrated cameras. In *IEEE Symposium on Computer Vision*, Coral Gables, FL, 1995.

[36] Z. Zhang and A. R. Hanson. 3D reconstruction based on homography mapping. In *ARPA Image Understanding Workshop*, Palm Springs, CA, 1996.

Unit´e de recherche INRIA Lorraine, Technopôle de Nancy-Brabois, Campus scientifique,
615 rue du Jardin Botanique, BP 101, 54600 VILLERS LÈS NANCY
Unit´e de recherche INRIA Rennes, Irisa, Campus universitaire de Beaulieu, 35042 RENNES Cedex
Unit´e de recherche INRIA Rhône-Alpes, 655, avenue de l'Europe, 38330 MONTBONNOT ST MARTIN
Unit´e de recherche INRIA Rocquencourt, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex
Unit´e de recherche INRIA Sophia-Antipolis, 2004 route des Lucioles, BP 93, 06902 SOPHIA-ANTIPOLIS Cedex

Éditeur
INRIA, Domaine de Voluceau, Rocquencourt, BP 105, 78153 LE CHESNAY Cedex (France)
http://www.inria.fr
ISSN 0249-6399