[Journal Name], [Volumn Number], 1–20 ([Volumn Year])

2 1/2 D visual servoing with respect to unknown objects through a new estimation scheme of camera displacement

EZIO MALIS AND FRANÇOIS CHAUMETTE

IRISA / INRIA Rennes
Campus de Beaulieu
35042 Rennes-cedex, France

;

Abstract. Classical visual servoing techniques need a strong a priori knowledge of the shape and the dimensions of the observed objects. In this paper, we present how the 2 1/2 D visual servoing scheme we have recently developed, can be used with unknown objects characterized by a set of points. Our scheme is based on the estimation of the camera displacement from two views, given by the current and desired images. Since vision-based robotics tasks generally necessitate to be performed at video rate, we focus only on linear algorithms. Classical linear methods are based on the computation of the essential matrix. In this paper, we propose a different method, based on the estimation of the homography matrix related to a virtual plane attached to the object. We show that our method provides a more stable estimation when the epipolar geometry degenerates. This is particularly important in visual servoing to obtain a stable control law, especially near the convergence of the system. Finally, experimental results confirm the improvement in the stability, robustness, and behaviour of our scheme with respect to classical methods.

Keywords: visual servoing, projective geometry, homography

1. Introduction

Standard eye-in-hand visual servoing approaches, that is position-based and image-based visual servoings, need a strong a priori knowledge of the 3D model of the observed object [29, 14, 17]. On one hand, in position-based visual servoing, the features used as inputs of the control scheme are expressed in the 3D Cartesian space [30]. To compute such features, the pose of the object with respect to the camera is estimated at each iteration of the control law. Numerous methods exist to recover the pose of an object (see [6] for example), but they are all based on the knowledge of a perfect geometric 3D model of the object. On the other hand, in image-based visual servoing, the visual features used as inputs of the control scheme are directly expressed in the 2D image space [9]. However, the internal part of the control scheme relies on an estimation or an approximation of the interaction matrix (also called image Jacobian). This matrix describes the relationship between the motion of the visual features in the image and the 3D motion of the camera mounted on the end-effector of the robot. If translational motions have to be controlled (which is generally the case), it thus depends on the depth from the camera to each considered geometrical feature. Once again, a pose estimation algorithm is generally used to estimate the 3D parameters involved in the interaction matrix. In some cases [9], a coarse approximation, corresponding to the value

of the interaction matrix computed at the desired robot position, is sufficient. However, an a priori knowledge on the 3D shape and dimensions of the observed object is still necessary to determine the desired value of the same 3D parameters. Another method in image-based visual servoing consists in numerically estimating the coefficients of the interaction matrix, without taking into account its analytical form [15, 19]. Contrarily to the previous ones, this method does not need any 3D a priori knowledge. However, it is unfortunately impossible to demonstrate and to ensure its stability.

In this paper, we present how the 2 1/2 D visual servoing scheme we have recently developed [24, 25], can be used with unmodeled objects. As will be detailed later, this scheme does not necessitate any 3D knowledge of the considered object, which increases the versatility and the application area of visual servoing. Furthermore, this scheme combines the advantages of classical visual servoings techniques and avoids their respective drawbacks. More precisely, the first drawback in position-based visual servoing is that none control is performed in the image, which implies that the object may get out of the camera field of view during the servoing (leading of course to its failure), especially if the initial robot position is far away from its desired one. The second drawback is that strong hypotheses have to be stated in order to demonstrate the stability of the system [3]. Image-based visual servoing also suffers from several drawbacks [3]: first, the interaction matrix may become singular during the servoing, which of course leads to an unstable behaviour. Second, local minima may be reached, which means that the final robot position does not correspond to the desired one. If another control strategy is used to avoid potential local minima, the motion in the image becomes unpredictable, which means that it is impossible to ensure that the object will always remain in the camera field of view. Furthermore, the robot trajectory may not be satisfactory because of the strong coupling in the coefficients of the interaction matrix. Finally, even if image-based visual servoing is known to be very robust in practice with respect to camera and robot calibration errors [8], it is in general impossible to exhibit exploitable analytical stability conditions.

As already described in [25] which was devoted to the automatic control part of our scheme, 2 1/2 D visual servoing consists in combining visual features obtained directly from the image, and estimated 3D information. As will be recalled in Section 2, we thus obtain a block-triangular interaction matrix that provides interesting decoupling properties. As detailed in [24, 25], it is also possible to be sure that the convergence will be ensured and that the object will remain in the camera field of view whatever the initial robot position. Analytical conditions to ensure the global stability of the system even in the presence of calibration errors have also been determined. In this paper, we focus on the estimation of the 3D parameters involved in our control scheme. If a 3D CAD model of the object is available, it is of course possible to obtain these parameters using a classical pose estimation algorithm. However, we will see that all these parameters can be determined from an Euclidean reconstruction up to a scalar factor. Such a reconstruction can be obtained from two images of an unknown object characterized by a set of points (assumed to be matched) [20, 10]. In our case, the first image is the desired one (acquired at the desired robot position during an off-line learning step), while the second image is the current one (acquired at each iteration of the control law).

The same idea of using an unknown object in visual servoing has been recently presented in [1]. However, the control scheme described in that paper corresponds to a classical position-based visual servoing, which means that it is subject to the drawbacks of this approach we have recalled above. Furthermore, the Euclidean reconstruction is obtained from the essential matrix, and we will show in this paper that it implies an unstable behaviour near the convergence of the system.

The Euclidean reconstruction from two views is well known to be the motion and structure from motion problem. It is, by its own nature, non-linear. Therefore, the classical approach to solve this problem is composed of two steps: using first a linear algorithm to provide an initialisation to a non-linear algorithm [20]. In this paper, we point out our attention only on the first linear stage, since the time processing of non linear algorithms are generally not compatible with the rate of vi-

sual servoing schemes (that have to be as close as possible to the video rate). Several methods were proposed to linearly solve the motion and structure from motion problem. They are generally based on the computation of the fundamental matrix [23] if pixel image points coordinates are used, or of the essential matrix [21, 13] if normalized image points coordinates are used. However, the epipolar geometry degenerates in some cases (for example if the motion is a pure rotation or if the considered object is planar [22]). If such degenerate configurations are not detected, the estimation of motion and structure will be completely unstable in their neighbourhood, which will induce an unstable and thus unsatisfactory behaviour of the control scheme. Unfortunately, in visual servoing, the displacement that the robot has to realize is of course unknown, and it may be possible to encounter a degenerate case even for the initial robot position. Moreover, a positioning task is achieved when the two considered images of the object are the same (image noise measurement excepted), which of course corresponds to a degenerate case for the epipolar geometry. Dealing with these degenerate configurations is thus particularly important in visual servoing.

The motion and structure can also be estimated from an homography matrix related to a virtual plane attached the object [11, 31]. The homography matrix may be estimated jointly to the epipole using, for example, the "virtual parallax algorithm" (VP) [2]. However, we will see that the epipole estimation is unnecessary for the homography estimation. The number of unknowns using the VP algorithm is thus not minimal if we are only interested in the estimation of the motion and structure (which is the case in our visual servoing problem). Furthermore, there are three supplementary epipolar configurations where it is impossible to extract the homography matrix with the VP algorithm.

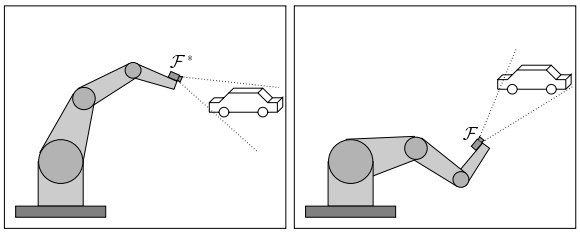
For these reasons, we propose a new method, again based on virtual parallax, for the direct estimation of the homography matrix relative to a virtual plane. With an adequate choice of the three points defining the virtual plane, we will see that it provides more stable results than the classical methods in the degenerate configurations for the epipolar geometry, as soon as image noise measurements are taken into account. Indeed, even if the degenerate cases are common to any reconstruction method, numerical stability of the estimation depends of the chosen method, and we explain in this paper why the one we propose gives satisfactory results. We have however to note that the problem of features matching has not been considered. Our method, in its current form, is thus unable to take into account potential outliers.

The use of planes and parallax for motion estimation has also been studied in [18] and [5], but using the hypothesis that four coplanar points can be extracted in both images. We will see that the method we propose does not need any hypothesis. Furthermore, the issue of handling degenerate situations has been recently addressed in [27], switching from epipole to homography estimation when degeneracies occur. However, in presence of noisy measurements, detecting such degeneracies is very complex. Moreover, even if the detection is perfectly realized, a discontinuity of the estimation will be obtained at each change of the used method if image noise and calibration errors exist. Since we use the same estimation method in all cases, our visual servoing scheme does not present such discontinuities.

The paper is organised as follows. In Section 2, we describe the 2 1/2 D visual servoing scheme and show which information provided by an Euclidean reconstruction is needed to design it. In Section 3, we review the classical linear methods to compute the fundamental matrix and then to extract the motion from the camera intrinsic parameters and the essential matrix. In Section 4, we propose an algorithm for the estimation of a collineation relative to a virtual plane attached to an unknown three-dimensional object characterized by a set of points. Knowing the camera internal parameters, the displacement of the camera can be extracted from the corresponding homography matrix. In Section 5, we compare our approach with the classical algorithms, especially in the particular case when the epipolar geometry is close to be degenerate. Finally, experimental results obtained using an eye-in-hand system are presented in Section 6.

2. The 2 1/2 D visual servoing

One of the typical applications of visual servoing consists in positioning an eye-in-hand system relative to an object, for a grasping task for instance. Generally, the positioning task is divided into two steps. In a first off-line learning step (see Figure 1), the camera is moved to its desired position with respect to the object (which corresponds to camera pose $\mathcal{F}^*$). The corresponding image is acquired and the extracted visual features are stored. In the second on-line step, after the camera and/or the object have been moved, the camera motion is controlled so that the current visual features (corresponding to camera pose $\mathcal{F}$) reach their desired position in the image. In other words, the rotation matrix $\mathbf{R}$ and the translation $\mathbf{t}$ between $\mathcal{F}$ and $\mathcal{F}^*$ have to reach the identity matrix and 0 respectively.



(a) final position (b) initial position

Fig. 1. Visual servoing with an eye-in-hand system

The 2D 1/2 visual servoing scheme consists in combining 2D image features and 3D information. More precisely, the feature vector used as input of the control law is selected as:

$$\mathbf{s} = [x, y, z, \theta\mathbf{u}^T]^T \tag{1}$$

where:

- x and y are the normalized metric coordinates of an image point, computed from the coordinates of this point measured in pixels and an estimation (generally coarse) of the camera intrinsic parameters ;
- $z = \log Z$, Z being the depth of the considered point;
- θ and $\mathbf{u}$ are respectively the angle and axis of rotation extracted from $\mathbf{R}$.

The task function $\mathbf{e}$, that has to be regulated to 0 [26], is directly obtained from the error $(\mathbf{s} - \mathbf{s}^*)$, where $\mathbf{s}^*$ is the desired value for $\mathbf{s}$. More precisely, $\mathbf{e}$ is given by:

$$\mathbf{e} = [x - x^*, y - y^*, \log \rho, \theta\mathbf{u}^T]^T \tag{2}$$

where the first two components of $\mathbf{e}$ are directly computed from the current and desired images, and the last four components of $\mathbf{e}$ are composed of 3D information that have to be estimated, ρ being defined as the ratio Z/Z^* between the current and desired depths of the selected point.

It is shown in [24, 25] that the corresponding interaction matrix, defined such that $\dot{\mathbf{e}} = \mathbf{L}\ \mathbf{v}$ where $\mathbf{v}$ is the camera velocity screw, is an upper block-triangular matrix given by:

$$\mathbf{L} = \begin{bmatrix} \frac{1}{Z}\mathbf{L_v} & \mathbf{L_{v\omega}} \\ \mathbf{0_3} & \mathbf{L}_\omega \end{bmatrix} \tag{3}$$

where:

$$\mathbf{L_v} = \begin{bmatrix} -1 & 0 & x \\ 0 & -1 & y \\ 0 & 0 & -1 \end{bmatrix}$$

$$\mathbf{L_{v\omega}} = \begin{bmatrix} xy & -(1+x^2) & y \\ (1+y^2) & -xy & -x \\ -y & x & 0 \end{bmatrix}$$

and:

$$\mathbf{L}_\omega = \mathbf{I}_3 - \frac{\theta}{2}[\mathbf{u}]_\times + \left(1 - \frac{\mathrm{sinc}(\theta)}{\mathrm{sinc}^2(\frac{\theta}{2})}\right)[\mathbf{u}]_\times^2 \tag{4}$$

with $\mathrm{sinc}(\theta) = \sin(\theta)/\theta$, $[\mathbf{u}]_\times$ being the antisymmetric matrix associated to $\mathbf{u}$.

The determinant of $\mathbf{L}_\omega$ is

$$\det(\mathbf{L}_\omega) = 1/\mathrm{sinc}^2(\frac{\theta}{2}) \tag{5}$$

and it is thus singular only for $\theta = 2k\pi$, $\forall k \in \mathbb{Z}^*$ (i.e. out of the possible workspace). We have also the following nice property:

$$\mathbf{L}_\omega^{-1}\ \theta\ \mathbf{u} = \theta\ \mathbf{u} \tag{6}$$

We can note that $\mathbf{L}$ is singular only in degenerate cases (such as $Z = 0$ and $1/Z = 0$). Finally, if the the object is known to be motionless and if a simple exponential decrease of each component of $\mathbf{e}$ is specified, we obtain the following control law:

$$\mathbf{v} = -\lambda\ \mathbf{L}^{-1}\ \mathbf{e} \tag{7}$$

where λ tunes the convergence rate. More precisely, we have:

$$\mathbf{v} = -\lambda \begin{bmatrix} Z\,\mathbf{L}_v^{-1} & -Z\,\mathbf{L}_v^{-1}\mathbf{L}_{v\omega} \\ \mathbf{0} & \mathbf{I}_3 \end{bmatrix} \begin{bmatrix} x - x^* \\ y - y^* \\ \log\rho \\ \theta\mathbf{u} \end{bmatrix} \quad (8)$$

If the CAD model of the object is known, a classical pose estimation algorithm can be used, and all the values involved in (8) are available at each iteration. Otherwise, if we deal with an unknown object, we can use an Euclidean reconstruction between the current and desired views, as we are going to see in the following sections. In that case, $\rho = Z/Z^*$ and $\mathbf{u}\theta$ can be computed, and the only unknown parameter is the depth Z. However, Z can be written $Z = \rho Z^*$ and the only unknown parameter of our control scheme becomes the constant scalar value Z^*. Furthermore, this value has not to be precisely determined (by hand in the experiments) since, as demonstrated in [25], it has a small influence on the stability of the system. In practice, an approximate value is chosen during the off-line learning stage.

Finally, if we consider possible calibration and measurement errors, the control law is given by:

$$\mathbf{v} = -\lambda\,\widehat{\mathbf{L}}^{-1}\,\widehat{\mathbf{e}} \quad (9)$$

where $\widehat{\mathbf{e}}$ is the measured value of $\mathbf{e}$ and $\widehat{\mathbf{L}}^{-1}$ is an approximation of $\mathbf{L}^{-1}$:

$$\widehat{\mathbf{L}}^{-1} = \begin{bmatrix} \widehat{Z}^*\widehat{\rho}\,\widehat{\mathbf{L}}_v^{-1} & -\widehat{Z}^*\widehat{\rho}\,\widehat{\mathbf{L}}_v^{-1}\widehat{\mathbf{L}}_{v\omega} \\ \mathbf{0} & \mathbf{I}_3 \end{bmatrix} \quad (10)$$

Let us emphasise that $\widehat{\mathbf{L}}^{-1}$ is an upper triangular square matrix without any singularity in the whole task space. The stability and convergence of the control law can thus be obtained for any initial camera position such that the considered object is in the camera field of view. Furthermore, such a decoupled system provides a satisfactory camera trajectory in the Cartesian space. Indeed, the rotational control loop is decoupled from the translational one (see Figure 2), and the chosen reference point is controlled by the translational camera d.o.f. such that its trajectory is a straight line in the state space, and thus in the image. If a correct calibration is available, this point will thus always remain in the camera field of view whatever the initial camera position. Of course, this property does not ensure that all the object will remain visible. In practice, it is possible to change the point during servoing, and we can select as reference point the nearest the bounds of the image plane. However, this solution leads to a discontinuity in the translational components of the camera velocity at each change of point. Another strategy is to select the reference point as the nearest of the center of gravity of the object in the image. This would increase the probability that the object remains in the camera field of view, but without any complete assurance. In [25], an adaptive control law is proposed to deal with this problem.

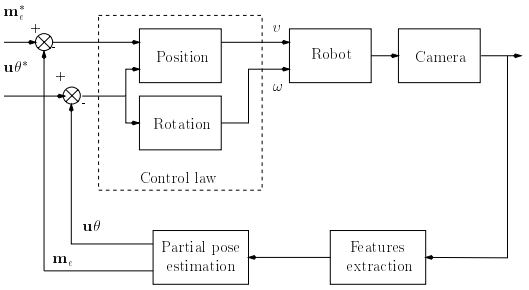


Fig. 2. Block diagram of the 2 1/2 D visual servoing

Furthermore, it is well known that the local asymptotic stability of the closed-loop system is ensured if all the eigenvalues of $\mathbf{L}\widehat{\mathbf{L}}^{-1}$ are positive. Similarly, the global asymptotic stability is ensured (which implies the decreasing of $\|\mathbf{e}\|$ at each iteration) if the sufficient condition $\mathbf{L}\widehat{\mathbf{L}}^{-1} > 0$ is satisfied. Determining analytical and practical conditions for the stability of image-based and position-based visual servoings is in general impossible (or under very strong hypotheses [3]). On the other hand, thanks to the nice form of $\mathbf{L}$ and $\widehat{\mathbf{L}}^{-1}$, it is possible to determine, when an Euclidean reconstruction is performed, the necessary and sufficient conditions for local asymptotic stability, and sufficient conditions for global asymptotic stability in the presence of camera calibration errors (see [24, 25] for more details). For example, it is possible to determine bounds on $\widehat{Z^*}$ in function of calibration errors such that the global stability of the system is ensured whatever the initial camera position.

We now describe how the 3D parameters involved in our control law can be estimated from a set of matched points in the current and desired images.

3. Camera displacement from the essential matrix

In this section, we review the classical approach to recover the displacement of a camera from two views of an unknown object. In our case, the first image corresponds to the desired one (acquired during the off-line learning step), and the second image to the current one (acquired at each iteration of the control law). The desired position of the camera optical centre is denoted C^*, while its current position is denoted C (see Figure 3). The perspective projection of a point $P \in \mathbb{P}^3$ in the first image is denoted $\mathbf{p}^*$ (with homogeneous coordinates $\mathbf{p}^* = \left[\begin{array}{ccc} u^* & v^* & 1 \end{array}\right]^T$). Similarly, the projection of P in the second image is denoted p (with homogeneous coordinates $\mathbf{p} = \left[\begin{array}{ccc} u & v & 1 \end{array}\right]^T$). p and p^* are measured in pixels and are assumed to be matched.

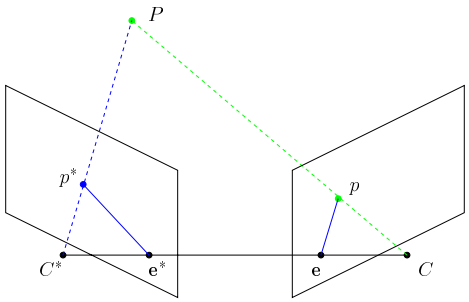


Fig. 3. Epipolar geometry

3.1. The epipolar geometry

It is well known that the plane defined by the three points C, C^* and P intersects the image planes in two epipolar lines. The first one is defined by $(\mathbf{p}^*, \mathbf{e}^*)$, and the second one, denoted $\mathbf{l}$, is defined by $(\mathbf{p}, \mathbf{e})$, where $\mathbf{e}^*$ and $\mathbf{e}$ are the epipoles (i.e., the projection of $\mathcal{C}$ and $\mathcal{C}^*$ in the image planes). Using projective coordinates, the epipolar line $\mathbf{l}$ can be written:

$$\mathbf{l} = \mathbf{p} \wedge \mathbf{G}_\infty \mathbf{p}^* \tag{11}$$

where $\mathbf{G}_\infty$ is the collineation relative to the plane at infinity [10]. Since the epipole $\mathbf{e}$ lies on line $\mathbf{l}$, we have $\mathbf{l}^T\mathbf{e} = 0$, which can be written, using equation (11), as:

$$\mathbf{p}^T \mathbf{F} \mathbf{p}^* = 0 \tag{12}$$

where $\mathbf{F} = [\mathbf{e}]_\times \mathbf{G}_\infty$ is the fundamental matrix ($[\mathbf{e}]_\times$ is the crossproduct matrix associated to vector $\mathbf{e}$). In the general case, $\mathbf{F}$ is rank 2, which implies a non-linear constraint on the nine entries of $\mathbf{F}$ [23].

3.2. Fundamental matrix estimation

We now review two linear algorithms to estimate the fundamental matrix. We remind that we only consider linear algorithms because of time processing constraints imposed by visual servoing.

3.2.1. The eight points algorithm. The classical approach to compute the epipolar geometry is the eight points algorithm [21, 13]. Since equation (12) is true for each pair of points $(\mathbf{p}_j, \mathbf{p}_j^*)$, it is possible to obtain a linear system if n pairs are available:

$$\mathbf{C}_f \mathbf{f} = 0 \tag{13}$$

where:

$$\mathbf{f} = \left[\begin{array}{ccccccccc} f_{11} & f_{12} & f_{13} & f_{21} & f_{22} & f_{23} & f_{31} & f_{32} & f_{33} \end{array}\right]^T$$

are the 9 unknown entries of $\mathbf{F}$ and $\mathbf{C}_f$ is a $(n \times 9)$ measurement matrix. System (13) is homogeneous and, since $\mathbf{F}$ is defined up to a scale factor, a minimum of 8 pairs of points are necessary to solve (13). In presence of noise, the linearized estimation problem can be written:

$$\min_{\mathbf{f}} \|\mathbf{C}_f \mathbf{f}\| \quad \text{subject to} \quad \|\mathbf{f}\| = 1 \tag{14}$$

The solution of this problem is obtained by performing the Singular Values Decomposition (SVD) of the measurement matrix $\mathbf{C}_f = \mathbf{U}\mathbf{S}\mathbf{V}^T$. 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 object is planar [22]), the fundamental matrix is also undefined, which implies an unstable estimation near this particular case. We will see in Section 4 that the method we propose is able to adequately deal with this problem.

Furthermore, we can note that this method does not take into account the rank 2 constraint on the fundamental matrix. This constraint is generally

introduced a posteriori using a non-linear algorithm [7, 23]. Since the aim of this paper is to focus on linear algorithms, the non-linear criteria are not detailed here.

3.2.2. The virtual parallax algorithm. To simplify the computation of matrix $\mathbf{F}$, Boufama et al. [2] perform a change of projective coordinates using 4 matched points in each image. These points are chosen such that not any three of them are collinear in the images. Let $\mathbf{M}$ and $\mathbf{M}^*$ be the matrices of change of coordinates, of dimension (3×3), respectively calculated as a function of $\mathbf{p}_1, \mathbf{p}_2, \mathbf{p}_3, \mathbf{p}_4$ and $\mathbf{p}_1^*, \mathbf{p}_2^*, \mathbf{p}_3^*, \mathbf{p}_4^*$. The image points $\tilde{\mathbf{p}}_j = \left[\begin{array}{ccc} \tilde{u}_j & \tilde{v}_j & \tilde{w}_j \end{array}\right]^T$ and $\tilde{\mathbf{p}}_j^* = \left[\begin{array}{ccc} \tilde{u}_j^* & \tilde{v}_j^* & \tilde{w}_j^* \end{array}\right]^T$ in the new coordinate system are given by $\tilde{\mathbf{p}}_j = \mathbf{M}^{-1}\mathbf{p}_j$ and $\tilde{\mathbf{p}}_j^* = \mathbf{M}^{*-1}\mathbf{p}_j^*$. Choosing $\left[\begin{array}{ccc} \tilde{\mathbf{p}}_1 & \tilde{\mathbf{p}}_2 & \tilde{\mathbf{p}}_3 \end{array}\right] = \left[\begin{array}{ccc} \tilde{\mathbf{p}}_1^* & \tilde{\mathbf{p}}_2^* & \tilde{\mathbf{p}}_3^* \end{array}\right] = \mathbf{I}_3$ for the first three points, the collineation matrix $\tilde{\mathbf{G}}$, related to the plane π defined by these three points, is diagonal when expressed in the new coordinate system:

$$\tilde{\mathbf{G}} = \mathbf{M}^{-1}\mathbf{G}\mathbf{M}^* = diag(\tilde{h}_u, \tilde{h}_v, \tilde{h}_w) \tag{15}$$

Then, the fundamental matrix can be written in the new coordinate system as $\tilde{\mathbf{F}} = [\tilde{\mathbf{e}}]_\times \tilde{\mathbf{G}}$ where $\tilde{\mathbf{e}} = \mathbf{M}^{-1}\mathbf{e}$ is the epipole in the new coordinate system. Using equation (12), we obtain:

$$\tilde{\mathbf{p}}^T [\tilde{\mathbf{e}}]_\times \tilde{\mathbf{G}}\ \tilde{\mathbf{p}}^* = 0 \tag{16}$$

which is polynomial of degree two in four unknowns (i.e., two unknowns for the epipole and two unknowns for the diagonal collineation matrix since they are defined up to a scale factor). After few developments, equation (16) can be written as [2]:

$$\mathbf{C}_{\tilde{f}}\ \tilde{\mathbf{f}} = 0 \tag{17}$$

where $\tilde{\mathbf{f}} = \left[\begin{array}{cccccc} \tilde{e}_x\tilde{g}_w & \tilde{e}_x\tilde{g}_v & \tilde{e}_y\tilde{g}_u & \tilde{e}_y\tilde{g}_w & \tilde{e}_z\tilde{g}_v & \tilde{e}_z\tilde{g}_u \end{array}\right]^T$. This new equation is linear homogeneous 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 $\mathbf{C}_{\tilde{f}}$ is of dimension $(m \times 6)$, and the system can be solved by performing the SVD of $\mathbf{C}_{\tilde{f}} = \mathbf{U}\mathbf{S}\mathbf{V}^T$. Once again, the solution $\overline{\tilde{\mathbf{f}}}$ is the column of $\mathbf{V}$ corresponding to the minimal singular value of $\mathbf{S}$ (0 in absence of noise). After the vector $\overline{\tilde{\mathbf{f}}}$ is obtained, the original unknowns can easily be determined.

As in the previous case, this method is inadequate when the epipole is undefined in the image. Furthermore, there are three supplementary singular cases where the collineation matrix $\tilde{\mathbf{G}}$ cannot be estimated. Indeed, if $\tilde{\mathbf{e}} = \left[\begin{array}{ccc} 1 & 0 & 0 \end{array}\right]^T$, only $\tilde{g}_2/\tilde{g}_3$ is known; if $\tilde{\mathbf{e}} = \left[\begin{array}{ccc} 0 & 1 & 0 \end{array}\right]^T$, only $\tilde{g}_1/\tilde{g}_3$ is known; and, if $\tilde{\mathbf{e}} = \left[\begin{array}{ccc} 0 & 0 & 1 \end{array}\right]^T$, only $\tilde{g}_1/\tilde{g}_2$ is known. If these particular cases can be detected, another algorithm can be used. However, in presence of noise, the detection of such particular cases is quite difficult and, if the detection fails, the results will not be accurate since zero values estimation is very sensitive to numerical errors.

The main advantage of the virtual parallax algorithm with respect to the eight points algorithm is that, even degenerating in the above singular cases, it can provide the collineation matrix, which is always defined contrarily to the fundamental matrix. However, in this algorithm, the collineation matrix estimation depends on the epipole estimation, and the number of unknowns is not minimal. For these reasons, we propose in Section 4 a method that determines directly the collineation matrix without estimating the epipole.

3.3. The Essential matrix

The fundamental matrix $\mathbf{F}$ is estimated using pixel image coordinates. From $\mathbf{F}$, the essential matrix $\mathbf{E}$ can be computed as follows:

$$\mathbf{E} = \mathbf{A}^T\mathbf{F}\mathbf{A} \tag{18}$$

$\mathbf{A}$ being a non-singular (3×3) matrix containing the intrinsic parameters of the camera:

$$\mathbf{A} = \begin{bmatrix} fk_u & -fk_u\cot(\theta) & u_0 \\ 0 & fk_v/\sin(\theta) & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{19}$$

where u_0 and v_0 are the coordinates of the principal point (in pixels), f is the focal length (in meters), k_u et k_v are the magnifications respectively in the $\overrightarrow{u}$ and $\overrightarrow{v}$ direction (in pixels/meters), and θ is the angle between these axes.

Matrix $\mathbf{E}$ must satisfy the Huang-Faugeras constraints [16]: $\sigma_1 = \sigma_2$ 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:

$$\mathbf{E} = [\mathbf{t}]_\times \mathbf{R} \tag{20}$$

where rotation matrix $\mathbf{R}$ and translation $\mathbf{t}$ represent the camera displacement between $\mathcal{F}$ and $\mathcal{F}^*$. The Huang-Faugeras constraints can be imposed a posteriori by using the algorithm of Tsai and Huang [28] to estimate the motion parameters. This method has been proved to be optimal by Hartley [12]. From $\mathbf{E}$, the rotation matrix $\mathbf{R}$ and the direction of translation $\mathbf{t}/\|\mathbf{t}\|$ can thus be directly calculated. All the values involved in our visual servoing scheme (except Z^* of course) are thus available: axis $\mathbf{u}$ and angle θ directly from $\mathbf{R}$, while ρ is given by $\frac{\|[\mathbf{t}]_\times \mathbf{R}\mathbf{m}^*\|}{\|[\mathbf{t}]_\times \mathbf{m}\|}$.

If the camera is coarsely calibrated (which is generally the case in visual servoing), it is clear that, even if $\mathbf{F}$ is perfectly estimated, $\mathbf{E}$ will be biased, which will induce errors on the estimation of the motion parameters. The closed-loop control used in visual servoing is generally able to overcome such problems. In fact, as already explained, the main problem encountered with the above methods occurs when the epipolar geometry is undefined, which is unfortunately the case when the camera comes near its desired position. Near convergence, unstable estimations will cause an unstable control law, which leads of course to an unsatisfactory behaviour. In the following section, we propose a different method to estimate the parameters involved in our visual servoing scheme. We will see in Section 5 that it provides more stable results near convergence, and is thus more adequate in visual servoing.

4. Camera displacement from the homography matrix

We now propose a linear algorithm to directly estimate the homography matrix relative to a virtual plane attached to the object.

4.1. The virtual parallax

Let us consider three 3D points P_i of the object ($i = 1, 2, 3$). We will see at the beginning of the next subsection how these points have to be chosen in practice. We here only consider that they are not collinear in both images, and thus define a virtual plane, denoted π (see Figure 4). It is well known that each image point with projective coordinates $\mathbf{p}_i$ in $\mathcal{F}$, is related to the corresponding image point with projective coordinates $\mathbf{p}_i^*$ in $\mathcal{F}^*$, by a collineation $\mathbf{G}$ such that [11]:

$$\mathbf{p}_i \propto \mathbf{G}\mathbf{p}_i^* \quad \{i = 1, 2, 3\} \tag{21}$$

where $\mathbf{G}$ is a homogeneous full rank (3×3) matrix. Let us remark that $\mathbf{G}$ is defined up to a scalar factor, therefore one of the entries of $\mathbf{G}$ can be set to 1 without loss of generality. Equation (21) is valid for all points lying on π. Therefore, if the considered object is known to be planar and if more than three points are available, the 8 unknown entries of $\mathbf{G}$ can be estimated by solving a simple linear homogeneous system obtained from $\mathbf{p}_i \wedge \mathbf{G}\mathbf{p}_i^* = 0$.

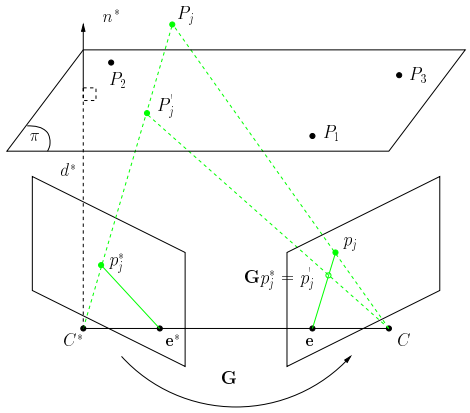


Fig. 4. Virtual parallax

Now, let us suppose that the structure of the object is not planar. If a point P_j does not belong to π, the line (C^*P_j) and plane π intersect in a virtual 3D point P_j' (see Figure 4). P_j' and P_j project on the same point $\mathbf{p}_j^*$ in the first image and on two different points ($\mathbf{p}_j$ and the virtual point $\mathbf{p}_j' = \mathbf{G}\mathbf{p}_j^*$) in the second image (parallax effect). The equation of the epipolar line $\mathbf{l}_j$ can be now written as follow:

$$\mathbf{l}_j = \mathbf{p}_j \wedge \mathbf{G}\mathbf{p}_j^* \tag{22}$$

4.2. Collineation estimation

Our approach, similar to the one proposed in [4], is based on the constraint that all the epipolar

lines meet in the epipole. Hence, for each set of three epipolar lines (22), we have:

$$\left| \mathbf{l}_j \;\; \mathbf{l}_k \;\; \mathbf{l}_l \right| = 0 \tag{23}$$

which means:

$$\left| \mathbf{p}_j \wedge \mathbf{G}\mathbf{p}_j^* \;\; \mathbf{p}_k \wedge \mathbf{G}\mathbf{p}_k^* \;\; \mathbf{p}_l \wedge \mathbf{G}\mathbf{p}_l^* \right| = 0 \tag{24}$$

However, equation (24) is non-linear with respect to the entries of the collineation matrix. In order to simplify the computation of $\mathbf{G}$, a change of projective coordinates is performed. In contrast with Boufama [2] and Couapel [4], the change of coordinates matrices $\mathbf{M}$ and $\mathbf{M}^*$ are constructed using only the three reference points chosen to define π. The transformation matrices are given by $\mathbf{M} = \left[\mathbf{p}_1 \;\; \mathbf{p}_2 \;\; \mathbf{p}_3 \right]$ and $\mathbf{M}^* = \left[\mathbf{p}_1^* \;\; \mathbf{p}_2^* \;\; \mathbf{p}_3^* \right]$. Choosing again $\left[\tilde{\mathbf{p}}_1 \;\; \tilde{\mathbf{p}}_2 \;\; \tilde{\mathbf{p}}_3 \right] = \left[\tilde{\mathbf{p}}_1^* \;\; \tilde{\mathbf{p}}_2^* \;\; \tilde{\mathbf{p}}_3^* \right] = \mathbf{I}_3$, the collineation matrix $\tilde{\mathbf{G}}$ in the new coordinates system is diagonal: $\tilde{\mathbf{G}} = \mathbf{M}^{-1}\mathbf{G}\mathbf{M}^* = diag(\tilde{g}_u, \tilde{g}_v, \tilde{g}_w)$. It is clear that the choice of the three reference points is important in our method. In order to obtain an accurate and robust estimation, this choice is done automatically by selecting the three points which maximize the surface of the corresponding triangle in both images. Furthermore, we can note that the change of coordinates normalizes the data, which is very important to obtain an accurate estimation in the projective domain [13].

Equation (24) can be written in the new coordinate system as:

$$\left| \tilde{\mathbf{p}}_j \wedge \tilde{\mathbf{G}}\tilde{\mathbf{p}}_j^* \;\; \tilde{\mathbf{p}}_k \wedge \tilde{\mathbf{G}}\tilde{\mathbf{p}}_k^* \;\; \tilde{\mathbf{p}}_l \wedge \tilde{\mathbf{G}}\tilde{\mathbf{p}}_l^* \right| = 0 \tag{25}$$

This equation based on virtual parallax is homogeneous and polynomial of degree three. Contrarily to equation (16) used in the virtual parallax algorithm, equation (25) does not depend on the epipole and contains only three unknowns. This is particularly important since the three singular cases of the virtual parallax method ($\tilde{\mathbf{e}} = \left[1 \;\; 0 \;\; 0 \right]^{\mathbf{T}}$, etc.) are not degenerate in our method. Furthermore, since the estimation of the epipole is unnecessary in our visual servoing scheme, we have no interest in introducing its components as supplementary unknowns. In other words, we benefit by the well known numerical analysis property that a more robust solution with respect to noise is obtained when the number of unknowns is minimal.

After computation, (25) can be written under the form:

$$\mathbf{C}_{\tilde{g}}\mathbf{x} = 0 \tag{26}$$

where the entries of the measurement matrix $\mathbf{C}_{\tilde{g}}$ are given in Appendix, and:

$$\mathbf{x}^T = \left[\tilde{g}_u^2\tilde{g}_v \;\; \tilde{g}_v^2\tilde{g}_u \;\; \tilde{g}_u^2\tilde{g}_w \;\; \tilde{g}_v^2\tilde{g}_w \;\; \tilde{g}_w^2\tilde{g}_u \;\; \tilde{g}_w^2\tilde{g}_v \;\; \tilde{g}_u\tilde{g}_v\tilde{g}_w \right]$$

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. Once again, the problem can be solved by performing the SVD of $\mathbf{C}_{\tilde{g}} = \mathbf{U}\mathbf{S}\mathbf{V}^T$ and by selecting as solution the column of $\mathbf{V}$ corresponding to the minimal singular value (0 in absence of noise). However, $\mathbf{C}_{\tilde{g}}$ is of dimension $(m \times 7)$ with $m >> 7$. In practice, we prefer to obtain the same solution from the SVD of $\mathbf{C}_{\tilde{g}}^T\mathbf{C}_{\tilde{g}} = \mathbf{V}\mathbf{S}^T\mathbf{S}\mathbf{V}^T$, which is of dimension (7×7). Memory space and time processing are thus minimized. Finally, the original unknowns can be computed by solving the following linear homogeneous system:

$$\begin{bmatrix} -\overline{x}_2 & \overline{x}_1 & 0 \\ \overline{x}_5 & 0 & -\overline{x}_3 \\ -\overline{x}_7 & \overline{x}_3 & 0 \\ \overline{x}_7 & 0 & -\overline{x}_1 \\ -\overline{x}_4 & \overline{x}_7 & 0 \\ \overline{x}_4 & 0 & -\overline{x}_2 \\ \overline{x}_6 & 0 & -\overline{x}_7 \\ 0 & -\overline{x}_6 & \overline{x}_3 \end{bmatrix} \begin{bmatrix} \tilde{g}_u \\ \tilde{g}_v \\ \tilde{g}_w \end{bmatrix} = 0 \tag{27}$$

Contrarily to the algorithms described in the previous section, the collineation matrix can be better estimated because the dimension of the problem is reduced and the epipole estimation is avoided. Furthermore, our method does not seem to introduce any new degenerate case. We explain now why this method provides indeed more accurate results when the epipolar geometry degenerates (in Sections 5 and 6 are given the experiments which confirm the following theoretical results).

The epipolar geometry degenerates when the projections of corresponding points are related by a collineation. This happens when all 3D points lie on a plane or when the camera performs a pure rotation. In this case, the columns of the determinant in equation (24) become null. However, they are not null for any $\tilde{g}_u, \tilde{g}_v, \tilde{g}_w$ since the collineation matrix is always defined and unique. Indeed, $\tilde{g}_u$, $\tilde{g}_v$ and $\tilde{g}_w$ have to verify equation (26), that is:

$$\begin{aligned} c_1\tilde{g}_u^2\tilde{g}_v + c_2\tilde{g}_v^2\tilde{g}_u + c_3\tilde{g}_u^2\tilde{g}_w + c_4\tilde{g}_v^2\tilde{g}_w \; + \\ c_5\tilde{g}_w^2\tilde{g}_u + c_6\tilde{g}_w^2\tilde{g}_v + c_7\tilde{g}_u\tilde{g}_v\tilde{g}_w \; = \; 0 \end{aligned} \tag{28}$$

If the matched points are related by a collineation, we have:

$$\begin{bmatrix} \tilde{u} \\ \tilde{v} \\ \tilde{w} \end{bmatrix} = \begin{bmatrix} \overline{g}_u & 0 & 0 \\ 0 & \overline{g}_v & 0 \\ 0 & 0 & \overline{g}_w \end{bmatrix} \begin{bmatrix} \tilde{u}^* \\ \tilde{v}^* \\ \tilde{w}^* \end{bmatrix} \tag{29}$$

We show in Appendix that the coefficients of equation (28) become in that case:

$$c_1 = \frac{\alpha}{\overline{g}_u^2\overline{g}_v}, \;\; c_2 = -\frac{\alpha}{\overline{g}_v^2\overline{g}_u}, \;\; c_3 = -\frac{\alpha}{\overline{g}_u^2\overline{g}_w},$$

$$c_4 = \frac{\alpha}{\overline{g}_v^2\overline{g}_w}, \;\; c_5 = \frac{\alpha}{\overline{g}_w^2\overline{g}_u}, \quad c_6 = -\frac{\alpha}{\overline{g}_w^2\overline{g}_v},$$

$$c_7 = 0$$

where $\alpha \neq 0$ except if the three considered points are collinear. Setting $\tilde{g}_w = \overline{g}_w = 1$ without loss of generality, equation (28) can be factorized as follows:

$$\alpha(\tilde{g}_u - \overline{g}_u)(\tilde{g}_v - \overline{g}_v)(\tilde{g}_u\overline{g}_v - \tilde{g}_v\overline{g}_u) = 0 \tag{30}$$

We thus obtain three different sets of solutions:

$$\begin{aligned} &\{\tilde{g}_u = \overline{g}_u, \forall\tilde{g}_v\} \; , \\ &\{\tilde{g}_v = \overline{g}_v, \forall\tilde{g}_u\} \; , \\ &\{\tilde{g}_u = \overline{g}_u\tilde{g}_v/\overline{g}_v, \forall\tilde{g}_v\} \end{aligned} \tag{31}$$

It is worth noting that all these solutions meet in a single solution, that is the expected one: $\tilde{g}_u = \overline{g}_u, \tilde{g}_v = \overline{g}_v$. In absence of noise, we could easily detect that a degenerate case occurs (in that case, the rank of $C_{\tilde{g}}$ is 1), and obtain the exact solution as $\tilde{g}_u = \overline{g}_u = -c_4/c_2 = c_5/c_3$ and $\tilde{g}_v = \overline{g}_v = -c_3/c_1 = c_6/c_4$. In the presence of noise, even if we consider that it is impossible to detect that we are in a degenerate case, the nice property that all sets of solutions have a unique common solution ensures that the solution obtained from (26) will be near this common solution, that is the real one. Of course, the error between the obtained result and the real value is directly related to the level of noise.

On the other hand, estimating the epipolar geometry through the fundamental matrix in the degenerate cases leads to very unstable results. Consider for example the case of a planar object. In that case, any point in the image can be chosen as epipole. Then, an infinity of vectors $\mathbf{f}$ are solutions of system (13). In presence of noise, if it is impossible to detect that a degenerate case occurs, any solution may be chosen as the good solution, which implies that the estimation of the motion parameters is generally completely wrong. On the other hand, as explained above, there exists only one collineation matrix, and its estimation is possible through systems (26) and (27).

Consider now the case of a pure rotation. The solution of system (13) should be $\mathbf{f} = 0$. However, the fundamental matrix is estimated by imposing the constraint $\|\mathbf{f}\| = 1$ since $\mathbf{f}$ is computed as a column of an orthonormal matrix. It is thus impossible to obtain an estimation near the right solution, that is $\|\mathbf{f}\| = 0$. On the contrary, the solutions of system (26) and (27) always satisfy the constraints $\|\mathbf{x}\| = 1$ and $\|\tilde{\mathbf{h}}\| = 1$ respectively. These constraints, imposed when performing the SVD of the measurement matrix, are ensured even in the degenerate cases. Then, the estimation of the camera displacement around these singular configurations will be more accurate when performed from the collineation matrix than from the fundamental matrix. As already stated, 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.

4.3. The homography matrix

The corresponding matrix of $\mathbf{G}$ in the calibrated domain is the homography matrix $\mathbf{H}$. The transformation between the pixel coordinates $\mathbf{p} = \begin{bmatrix} u & v & 1 \end{bmatrix}^T$ and the normalized coordinates $\mathbf{m} = \begin{bmatrix} x & y & 1 \end{bmatrix}^T$ of an image point is known to be $\mathbf{p} = \mathbf{A}\mathbf{m}$ where $\mathbf{A}$ is given in (19). The homography matrix can be written as a function of the calibration parameters and of the collineation ma-

trix as follows:

$$\mathbf{H} = \mathbf{A}^{-1}\mathbf{G}\mathbf{A} \tag{32}$$

Furthermore, the homography matrix can be written as a function of the camera displacement [11]:

$$\mathbf{H} = \mathbf{R} + \frac{\mathbf{t}}{d^*}\,\mathbf{n}^{*T} \tag{33}$$

where $\mathbf{n}^*$ is the normal to the virtual plane π expressed in $\mathcal{F}^*$, and d^* is the distance from C^* to π (see Figure 5). 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 compute these parameters, one of the algorithms proposed in [11] or [31] can be used. Unfortunately, in the most general case, we have two different solutions. If the object is known to be planar, the indetermination can be eliminated if an additional information is available (for example from the normal vector to the virtual plane π). Otherwise, the indetermination is eliminated by considering another reference plane and by choosing the common solution between the two pairs [11]. In visual servoing, this has to be done only once, at the first iteration of the control law, since the solution the nearest of the previous one can be chosen for the next iterations.

Finally, the ratio ρ involved in our control scheme can be directly computed from $\mathbf{R}$, $\mathbf{t}_{d^*}$, and $\mathbf{n}^*$. Indeed, we have:

$$\begin{cases} \rho = \dfrac{\mathbf{n}^T\mathbf{m}}{\mathbf{n}^{*T}\mathbf{m}^*} r & \text{if } \mathbf{m} \in \pi \\ \rho = \dfrac{\| [\mathbf{t}_{d^*}]_\times \mathbf{R}\mathbf{m}^* \|}{\| [\mathbf{t}_{d^*}]_\times \mathbf{m} \|} & \text{if } \mathbf{m} \notin \pi \end{cases} \tag{34}$$

where the ratio r between distances d and d^* (see Figure 5) is given by:

$$r = \frac{d}{d^*} = \det(\mathbf{H}) = 1 + \mathbf{n}^{*T}\mathbf{R}\mathbf{t}_{d^*} \tag{35}$$

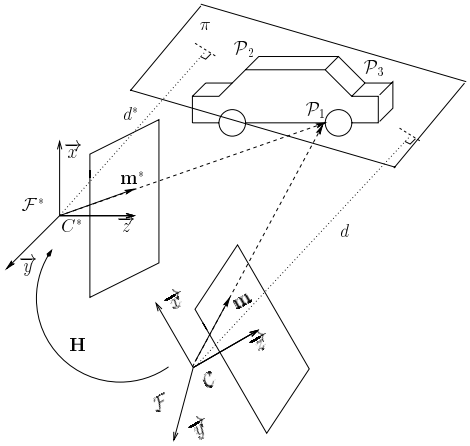


Fig. 5. Motion and structure parameters.

If the camera is not perfectly calibrated and $\widehat{\mathbf{A}}$ is used instead of $\mathbf{A}$, it is again possible to express the parameters involved in our control scheme. More precisely, if we consider that the homography at infinity can be estimated, and restrict the computation to the case where the point $\mathbf{m}$ used in the control scheme belongs to π, we obtain:

$$\widehat{\theta} = \theta\ ,\ \widehat{\mathbf{u}} = \frac{\delta\mathbf{A}\mathbf{u}}{\|\delta\mathbf{A}\mathbf{u}\|} \text{ and } \widehat{\rho} = \rho \tag{36}$$

where $\delta\mathbf{A} = \widehat{\mathbf{A}}^{-1}\mathbf{A}$ describes the (unknown) error on the intrinsic parameters. It must be emphasized that rotation angle θ and ratio ρ are computed without error. Our control scheme is thus particularly robust with respect to calibration errors. As already explained in Section 2, thanks to the simple above relations, we have been able in [24, 25] to determine analytical conditions to ensure the local and global asymptotic stability of our system in presence of calibration errors.

5. Simulations results

In this section, we compare the accuracy of our method to standard ones. The simulated objects are composed of a cloud of 16 points, and, for each experiment, several objects are randomly built. The camera displacement is also randomly chosen, and, for each camera displacement, several random additive noise on image coordinates (with 1 pixel standard deviation) was generated. As already explained, the three points automatically selected to define the reference plane π are such that they maximize the surface of the correspond-

ing triangles in both images. The mean, the standard deviation and the maximum of the absolute value of the following errors was then computed (where the hat refers to the estimated value):

- 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 $\widehat{\mathbf{t}}/\|\widehat{\mathbf{t}}\|$.

As already said, we focused in this paper on linear estimations since they are the only ones able to give results at video rate. Since time processing is not critical in simulation, we consider also in this section the results obtained with the non-linear method described in [7, 32]. The results of the different methods are plotted in the figures respectively with:

- a triangle for the eight point algorithm using normalized data (EL) (see Section 3.2.1).
- a square for the motion estimation using the virtual parallax algorithm (VP) (see Section 3.2.2).
- a circle for the linear homography matrix estimation algorithm (HL) (see Section 4.2).
- a diamond for the non-linear algorithm (NL) described in [7, 32] and initialized with the EL algorithm results.

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

5.1. Accuracy with planar objects

As already explained, our visual servoing scheme does not necessitate any a priori information about the 3D model of the considered object. In man-made environment, it is very common to find planar or nearly planar surfaces. It is thus important that the algorithm estimating the camera displacement provides accurate results when the considered object is planar, even if it corresponds to a degenerate case of the epipolar geometry.

We thus consider here objects composed of 16 coplanar points randomly chosen in a square of 30×30 cm^2. In Figure 6 are given the mean of the error, its standard deviation and the maximal error computed over 40000 samples varying randomly the camera displacement and the structure of the points in the square. More precisely, 40 planar objects and 100 camera displacements have been considered, and for each of these configurations, 10 experiments adding random image noise have been realized. As for the camera orientation, it varies randomly from a nominal position in front of the plane with a maximal displacement of $\pm 60°$. The translation of the camera is chosen such that the points remains in the camera field of view. The initial distance from the plane is 50 cm.

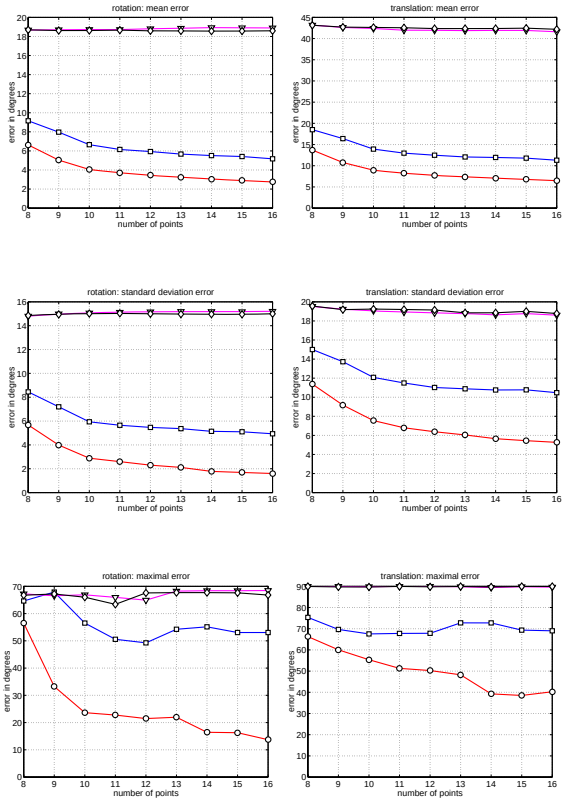


Fig. 6. Planar object: rotation and translation error versus number of points.

As expected, considering a planar object is unfavourable for the algorithms EL and NL based on the fundamental matrix estimation. Important mean errors (18° and 40° for the rotational and translational errors) are obtained using these algorithms whatever the number of points. Results using our HL algorithm are satisfactory (the mean

error is 6° for the rotation and 15° for the translation) since the most accurate and stable. Finally, the VP method gives intermediary results since, even if the displacement is computed from the homography, the homography is estimated jointly with the epipole, which introduces important perturbations.

5.2. Accuracy at the final position

We now consider the case of a small camera displacement. In visual servoing, since this displacement is a priori unknown, it may thus be small, even for the initial camera position. This is typically the case for robot stabilization and target tracking tasks. Furthermore, whatever the initial camera position, it is obvious that, at convergence of the visual servoing scheme, the displacement has to be as small as possible. To preserve the stability of the control law, it is thus extremely important that the algorithm used to estimate the camera displacement provides an accurate and stable result in the case where $\mathbf{R} = \mathbf{I}$ and $\mathbf{t} = \mathbf{0}$, even if the epipolar geometry is degenerate (since the epipole is undefined in the image).

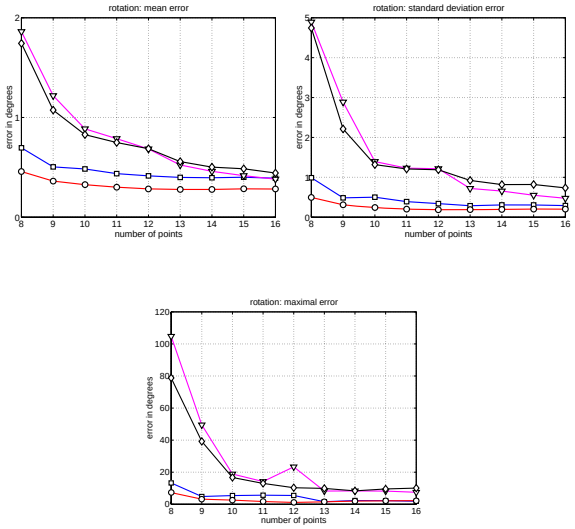


Fig. 7. Rotation error versus number of points when the camera is at its final position.

For the simulation, we set $\mathbf{R} = \mathbf{I}$, $\mathbf{t} = \mathbf{0}$ and use 100 objects composed of 16 points randomly chosen in a cube of $30 \times 30 \times 30$ cm^3. The results obtained for 10000 tests (100 tests with different noise for each object) are shown in Figure 7. As expected, the HL algorithm produces more accurate results than the VP algorithm, since the homography is not estimated jointly to the epipole. As expected also, the EL and NL algorithms are less accurate than the HL and VP algorithms, especially when the number of considered points is small. These results confirm that, in the singular cases, the use of an homography matrix is preferable to obtain the motion parameters.

5.3. Accuracy with a rotating camera

In this simulation, we consider a stationary camera that performs a pure rotation of 10° around a random axis (10000 tests corresponding to 20 objects and 50 different axes of rotation have been done). As can be seen in Figure 8, we obtain very similar results to the previous simulation and the HL algorithm produces again the bests results in this degenerate case of the epipolar geometry.

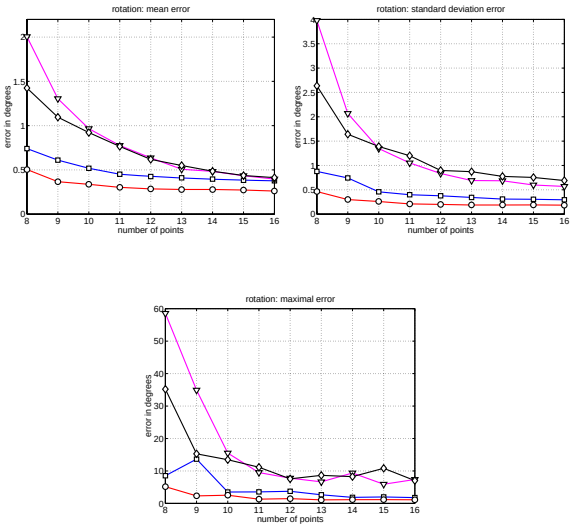


Fig. 8. Pure rotation of the camera: error versus number of points

5.4. Accuracy with random camera displacement

Figure 9 shows the results obtained with random generic displacements (once again, 10000 samples have been done to deal with 20 objects and 50 displacements). In that case, the NL algorithm produces, as expected, the best results, but we can note that those obtained using the HL algo-

rithm are satisfactory in regard to those obtained using the EL method (since they are very close). Finally, the VP algorithm gives the worst results, since the joint estimation of the epipole and of the homography matrix induces perturbations on the camera motion estimation.

We can remark that, for all methods, the errors are most important in this experiment than in the previous ones. This is due to the fact that the random camera displacement may imply that the object is very small in the image, which of course induces less accurate results.

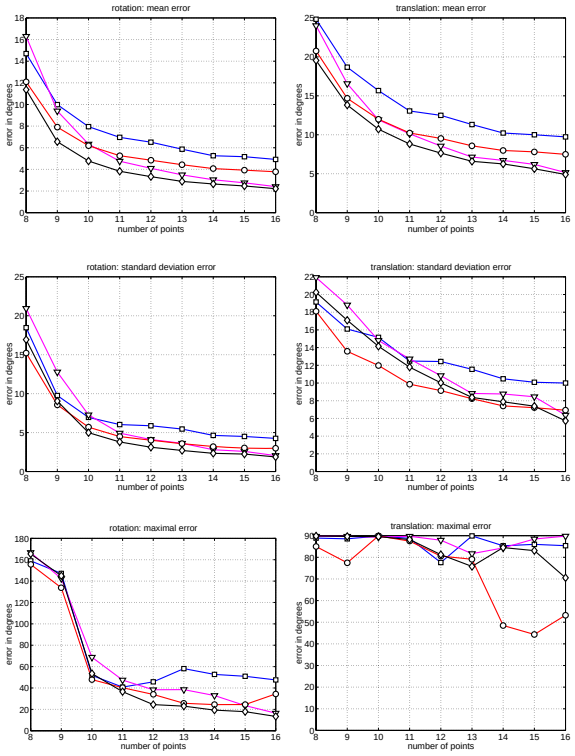


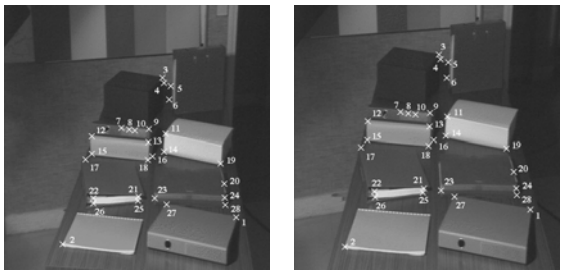
Fig. 9. Generic displacement: results vs number of points

To conclude, we recall that the choice of the three reference points is important in our HL method (as already explained, they are selected to maximize the area of the corresponding triangles in both images). The quality of the results when these points are matched with large imprecision can be very bad. However, in all the presented simulation results, the variance of the noise on all points was of 1 pixel, which means that the algorithm is accurate even in presence of noisy images. Finally, as it will be explained below, dealing with outliers (mismatched points) was not in the scope of this paper.

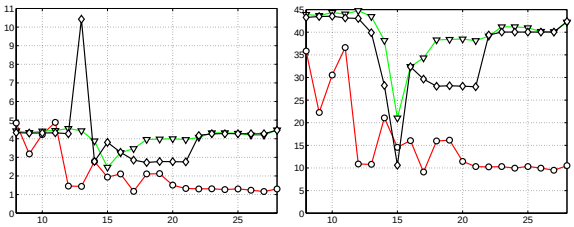
6. Experimental results

6.1. Camera displacement estimation using a real scene

We now consider a real scene 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 $\mathbf{r} = \begin{bmatrix} 2.1 & -3.1 & -0.7 \end{bmatrix}^T$ dg. The points (matched using the Image Matching software[1], developed by Zhang) in both images[2] were numbered from 1 to 28 (see Figure 10a and Figure 10b).



(a) first image (b) second image



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

Fig. 10. Real scene: results versus number of points

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 10c and Figure 10d respectively. On the whole, the NL algorithm gives better results than the EL algorithm (surprisingly except for 13 points). According to

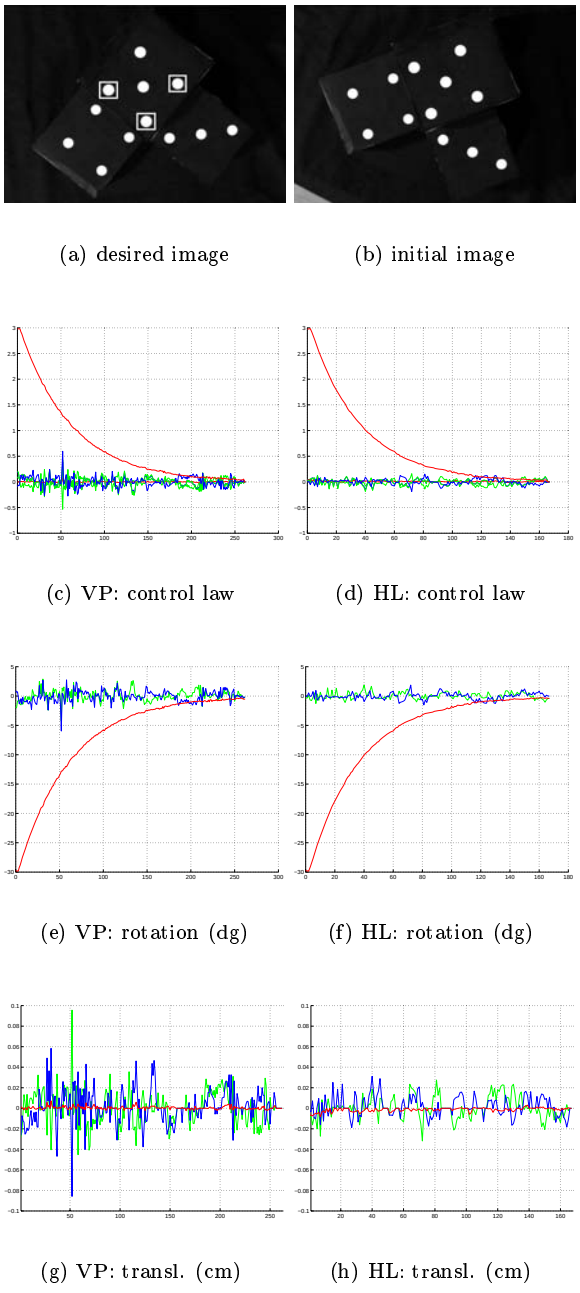
the simulation results, the HL algorithm produces more accurate results than the EL algorithm. Finally, it is quite surprising that the HL method gives more accurate results than the NL method. This is due to the fact that the camera displacement is not important in regard of the dimension of the scene, which means that the considered example is not far from a degenerate case of the epipolar geometry.

6.2. 2 1/2 D Visual servoing results

The HL method has been integrated in the visual servoing scheme described in Section 2 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, we have used the pixel and focal lengths given by the constructor in order to compute the image coordinates u and v from their measured values (in pixels) in the image. The center of the image has been used for the principal point. The object was a black board with twelve white marks on three parallel planes (see Figure 11). 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, such as the ones considered in the experiments, point matching between initial and reference images is a difficult computer vision problem. This problem has not been considered here because of the simplicity of the considered object. Furthermore, this matching has to be done only once, just before the beginning of the visual servoing, where real time issue is not needed. Finally, in the robotics applications we are working on, this matching process can be solved thanks to the help of a human operator.

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 when the epipole is undefined, which unfortunately occurs when the camera reaches its desired position. For this reason, only the VP and HL methods were tested. Finally, 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 11a where a square has been superimposed around each reference point).



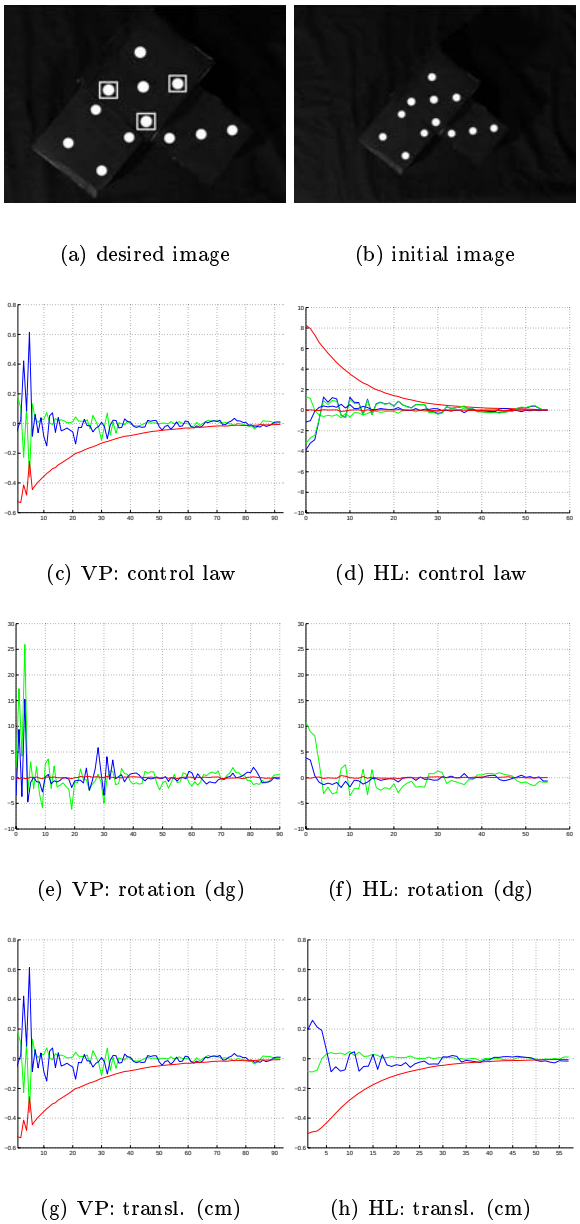
(a) desired image (b) initial image

(c) VP: control law (d) HL: control law

(e) VP: rotation (dg) (f) HL: rotation (dg)

(g) VP: transl. (cm) (h) HL: transl. (cm)

Fig. 11. Rotational camera displacement: results versus iteration number



(a) desired image (b) initial image

(c) VP: control law (d) HL: control law

(e) VP: rotation (dg) (f) HL: rotation (dg)

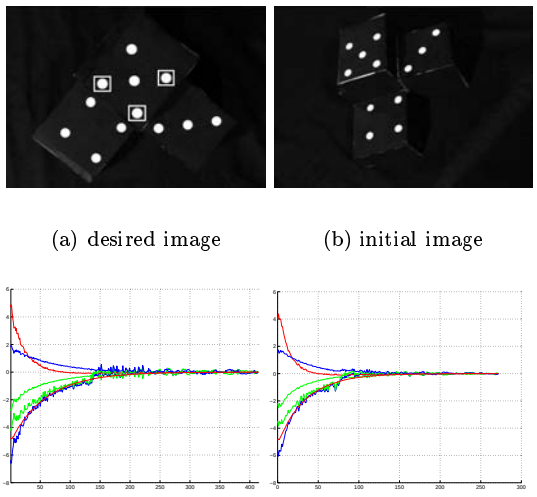
(g) VP: transl. (cm) (h) HL: transl. (cm)

Fig. 12. Translational camera displacement results versus iteration number

6.2.1. Pure Rotation. The results of the 2 1/2 D visual servoing, obtained when the camera displacement were a pure rotation of -30 dg around the z axis, are given in Figure 11. The HL and VP algorithms produce good results even if the epipole is undefined all along the experiment. However, it can be observed that the rotation (Figure 11f) and the scaled translation (Figure 11h) estimated using the HL algorithm are less noisy than the ones estimated using the VP algorithm (see Figure 11e and Figure 11g). This implies a more stable control law (see Figure 11c and Figure 11d), and demonstrates the interest of our method with respect to classical ones.

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 ($\mathbf{e} = \mathbf{p}_1$). The obtained results are displayed on Figure 12. As can be seen on the plots, from iteration 0 to 5, the VP algorithm is very unstable since it is near its singularity, while the HL algorithm is always more accurate and stable. Once again, we can note that the estimation of the parameters involved in our control scheme of course reflects on the computed control law, which is thus more stable and satisfactory using the HL method.

6.2.3. Generic camera displacement. In this last experiment (see Figure 13), a generic camera displacement is performed: $\mathbf{t} = \begin{bmatrix} -1.3 & 55.2 & 4.1 \end{bmatrix}^T$ cm and $\mathbf{r} = \begin{bmatrix} 36.2 & -17.2 & 48.4 \end{bmatrix}^T$ dg. Once again and according to the simulation results, the HL algorithm produces more stable results than the VP algorithm (see the output control law in Figure 13d and in Figure 13c respectively).

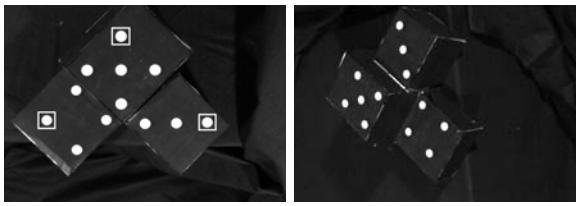


(a) desired image (b) initial image

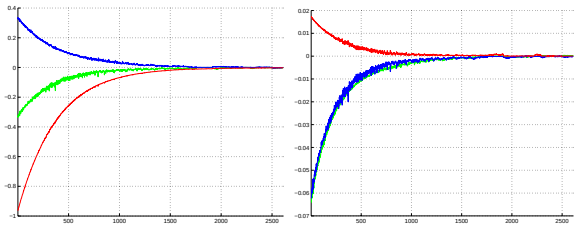
(c) VP: control law (d) HL: control law

Fig. 13. Generic camera displacement results versus iteration number

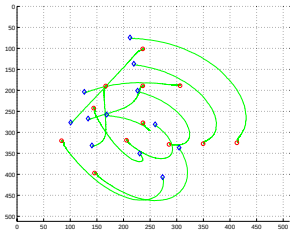
From the initial to final camera poses, the estimated rotational displacement using the HL algorithm is $\overline{\mathbf{r}} = \left[\begin{array}{ccc} 34.8 & -14.9 & 48.3 \end{array}\right]^T$ dg. Similarly, the estimated direction of translation is $\overline{\mathbf{t}}/\|\overline{\mathbf{t}}\| = \left[\begin{array}{ccc} -0.04 & 0.99 & 0.04 \end{array}\right]^T$(while the real direction of translation was $\mathbf{t}/\|\mathbf{t}\| = \left[\begin{array}{ccc} -0.02 & 0.99 & 0.07 \end{array}\right]^T$). The algorithm is thus accurate (maximal rotational error is around 2°, as well as the angle error on the direction of translation) despite the coarse calibration which has been used.



(a) desired image (b) initial image



(c) velocity of rotation (dg/s) (d) velocity of translation (cm/s)

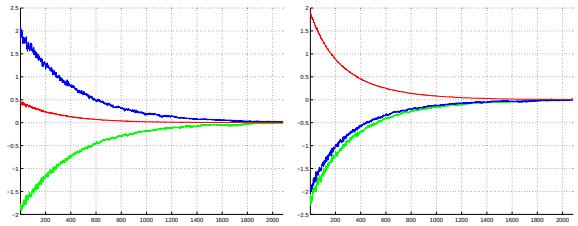


(e) image trajectories

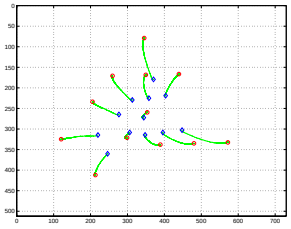
Fig. 14. Another experiment with large displacement

We finally present on Figure 14 the results obtained using the HL algorithm when the three reference points are taken spread in the image (see Figure 14a). The images corresponding to the desired and initial camera position are given in Figure 14a and 14b respectively. The points trajectory in the image recorded during the experiment are plotted on Figure 14e. We can note that all points remain in the camera field of view (which is not the case using classical position-based and image-based approaches [25]). Furthermore, the trajectory of the point selected as input of the control scheme is easily identified since it looks like a straight line in the image. Our scheme is thus particularly robust with respect to modelling errors since it is not disturbed by the use of a coarse camera calibration and a coarse approximation of Z^* (in the experiment, Z^* has been set to 50 cm while its real value is equal to 60 cm). Finally, we can note on Figure 14c and 14d the improvement on the stability of the control law brought by an adequate choice of the 3 reference points used to define the virtual plane π.

(a) desired image (b) initial image



(c) rotational velocity (dg/s) (d) translational velocity (cm/s)



(e) image trajectories

Fig. 15. Results obtained using a planar object

Numerous other experiments are detailed in [24, 25]. We refer an interested reader to these references where it is shown that the convergence domain of the 2 1/2 D visual servoing is larger than for the classical position-based and image-based schemes. Important camera and hand-eye calibration errors are also considered.

6.2.4. Experiment on a planar object. We now present the results obtained using a planar object (see Figure 15 where the 12 points now lies on a plane). We recall that our method, as the others, is theoretically unefficient to deal with this case where the epipolar geometry is degenerate. However, as already explained, as soon as noise exists in the image measurements, our method is able to provide satisfactory results. This is demonstrated on Figure 15c and 15d where the components of the computed control law are depicted. We can note that, even if the level of noise is very low (approximatively 0.1 pixels with so simple images), the estimation of the parameters involved in our control scheme is as stable as for a non planar object, since it is difficult to find any difference in the level of noise of the control law between this experiment and the previous one.

7. Conclusion

The visual servoing scheme presented in this paper has many advantages over the standard methods. The most important one is that our scheme does not need any 3D model of the observed object. 2 1/2 visual servoing presents also very interesting decoupling and stability properties, and it is particularly robust with respect to modelling errors. The control scheme is designed from an Euclidean reconstruction which can be obtained either from the essential matrix or from an homography matrix. However, we have shown and confirmed by simulation and experimental results that recovering the camera displacement from the homography matrix gives more stable results when the camera comes near its desired position. Future work will be devoted to the application of 2 1/2 D visual servoing on real images, where image processing and features matching have to be considered carefully.

Acknowledgements

This work was supported by INRIA and 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 this paper.

Appendix

The j-row of the measurement matrix $\mathbf{C}_{\bar{h}}$ (see (26)) can be written in function of the image points coordinates as follows:

$$
\begin{aligned}
c_1 &= w_i w_j v_k u_k^* (u_j^* v_i^* - u_i^* v_j^*) + \\
&\quad w_i w_k v_j u_j^* (u_i^* v_k^* - u_k^* v_i^*) + \\
&\quad w_j w_k v_i u_i^* (u_k^* v_j^* - u_j^* v_k^*) \\
c_2 &= w_i w_j u_k v_k^* (u_i^* v_j^* - u_j^* v_i^*) + \\
&\quad w_i w_k u_j v_j^* (u_k^* v_i^* - u_i^* v_k^*) + \\
&\quad w_j w_k u_i v_i^* (u_j^* v_k^* - u_k^* v_j^*) \\
c_3 &= v_i v_k w_j u_j^* (u_i^* w_k^* - u_k^* w_i^*) + \\
&\quad v_i v_j w_k u_k^* (u_j^* w_i^* - u_i^* w_j^*) + \\
&\quad v_j v_k w_i u_i^* (u_k^* w_j^* - u_j^* w_k^*) \\
c_4 &= u_i u_k w_j v_j^* (v_i^* w_k^* - v_k^* w_i^*) + \\
&\quad u_i u_j w_k v_k^* (v_j^* w_i^* - v_i^* w_j^*) + \\
&\quad u_j u_k w_i v_i^* (v_k^* w_j^* - v_j^* w_k^*) \\
c_5 &= v_j v_k u_i w_i^* (u_j^* w_k^* - u_k^* w_j^*) + \\
&\quad v_i v_k u_j w_j^* (u_k^* w_i^* - u_i^* w_k^*) + \\
&\quad v_i v_j u_k w_k^* (u_i^* w_j^* - u_j^* w_i^*) \\
c_6 &= u_j u_k v_i w_i^* (v_j^* w_k^* - v_k^* w_j^*) + \\
&\quad u_i u_k v_j w_j^* (v_k^* w_i^* - v_i^* w_k^*) + \\
&\quad u_i u_j v_k w_k^* (v_i^* w_j^* - v_j^* w_i^*) \\
c_7 &= u_i v_k w_j (u_k^* v_j^* w_i^* - u_j^* v_i^* w_k^*) + \\
&\quad u_k v_i w_j (u_j^* v_k^* w_i^* - u_i^* v_j^* w_k^*) + \\
&\quad u_i v_j w_k (u_k^* v_i^* w_j^* - u_j^* v_k^* w_i^*) + \\
&\quad u_j v_i w_k (u_i^* v_k^* w_j^* - u_k^* v_j^* w_i^*) + \\
&\quad u_k v_j w_i (u_j^* v_i^* w_k^* - u_i^* v_k^* w_j^*) + \\
&\quad u_j v_k w_i (u_i^* v_j^* w_k^* - u_k^* v_i^* w_j^*)
\end{aligned}
$$

Let us now suppose that two points are related by a collineation (which is the case for a planar

object or when the camera displacement is a pure rotation). In that case, we have:

$$\begin{bmatrix} u \\ v \\ w \end{bmatrix} = \begin{bmatrix} \overline{g}_u & 0 & 0 \\ 0 & \overline{g}_v & 0 \\ 0 & 0 & \overline{g}_w \end{bmatrix} \begin{bmatrix} u^* \\ v^* \\ w^* \end{bmatrix} \qquad (37)$$

We thus have $u^* = u/\overline{g}_u$, $v^* = v/\overline{g}_v$, $w^* = w/\overline{g}_w$, from which we can deduce:.

$$\begin{aligned}
c_1 &= (w_i w_j v_k u_k (u_j v_i - u_i v_j) + \\
&\quad w_i w_k v_j u_j (u_i v_k - u_k v_i) + \\
&\quad w_j w_k v_i u_i (u_k v_j - u_j v_k))/(\overline{g}_u^2 \overline{g}_v) \\
c_2 &= (w_i w_j u_k v_k (u_i v_j - u_j v_i) + \\
&\quad w_i w_k u_j v_j (u_k v_i - u_i v_k) + \\
&\quad w_j w_k u_i v_i (u_j v_k - u_k v_j))/(\overline{g}_v^2 \overline{g}_u) \\
c_3 &= (v_i v_k w_j u_j (u_i w_k - u_k w_i) + \\
&\quad v_i v_j w_k u_k (u_j w_i - u_i w_j) + \\
&\quad v_j v_k w_i u_i (u_k w_j - u_j w_k))/(\overline{g}_u^2 \overline{g}_w) \\
c_4 &= (u_i u_k w_j v_j (v_i w_k - v_k w_i) + \\
&\quad u_i u_j w_k v_k (v_j w_i - v_i w_j) + \\
&\quad u_j u_k w_i v_i (v_k w_j - v_j w_k))/(\overline{g}_v^2 \overline{g}_w) \\
c_5 &= (v_j v_k u_i w_i (u_j w_k - u_k w_j) + \\
&\quad v_i v_k u_j w_j (u_k w_i - u_i w_k) + \\
&\quad v_i v_j u_k w_k (u_i w_j - u_j w_i))/(\overline{g}_w^2 \overline{g}_u) \\
c_6 &= (u_j u_k v_i w_i (v_j w_k - v_k w_j) + \\
&\quad u_i u_k v_j w_j (v_k w_i - v_i w_k) + \\
&\quad u_i u_j v_k w_k (v_i w_j - v_j w_i))/(\overline{g}_w^2 \overline{g}_v) \\
c_7 &= (u_i v_k w_j (u_k v_j w_i - u_j v_i w_k) + \\
&\quad u_k v_i w_j (u_j v_k w_i - u_i v_j w_k) + \\
&\quad u_i v_j w_k (u_k v_i w_j - u_j v_k w_i) + \\
&\quad u_j v_i w_k (u_i v_k w_j - u_k v_j w_i) + \\
&\quad u_k v_j w_i (u_j v_i w_k - u_i v_k w_j) + \\
&\quad u_j v_k w_i (u_i v_j w_k - u_k v_i w_j))/(\overline{g}_u \overline{g}_v \overline{g}_w)
\end{aligned}$$

Posing $c'_1 = c_1(\overline{g}_u^2 \overline{g}_v)$, $c'_2 = c_2(\overline{g}_v^2 \overline{g}_v)$, $c'_3 = c_3(\overline{g}_u^2 \overline{g}_w)$, $c'_4 = c_4(\overline{g}_v^2 \overline{g}_w)$, $c'_5 = c_5(\overline{g}_w^2 \overline{g}_u)$, $c'_6 = c_6(\overline{g}_w^2 \overline{g}_v)$ and $c'_7 = c_7(\overline{g}_u \overline{g}_v \overline{g}_w)$, and expanding the equations, we obtain after some tedious computations:

$$\begin{aligned}
&c'_1 = \alpha\ , \quad c'_2 = -\alpha\ , \quad c'_3 = -\alpha\ , \\
&c'_4 = \alpha\ , \quad c'_5 = \alpha\ , \quad\ \ c'_6 = -\alpha\ , \\
&c'_7 = 0
\end{aligned}$$

where:

$$\begin{aligned}
\alpha &= u_j u_k v_i v_k w_i w_j - u_i u_k v_j v_k w_i w_j + \\
&\quad u_i u_j v_j v_k w_i w_k - u_j u_k v_i v_j w_i w_k + \\
&\quad u_i u_k v_i v_j w_j w_k - u_i u_j v_i v_k w_j w_k
\end{aligned}$$

We can note that $\alpha \neq 0$, except when the three points involved in (26) are collinear.

Notes

1. available on http://www.inria.fr/robotvis/personnel/zzhang/zzhang-eng.html
2. provided by the Inria Syntim project (http://www-syntim.inria.fr/syntim/analyse/paires-eng.html)

References

1. R. Basri, E. Rivlin, and I. Shimshoni. Visual homing: Surfing on the epipoles. *IEEE Int. Conf. on Computer Vision*, pp. 863–869, Bombay, India, January 1998.
2. B. Boufama and R. Mohr. Epipole and fundamental matrix estimation using the virtual parallax property. *IEEE Int. Conf. on Computer Vision*, pp. 1030–1036, Cambridge, USA, 1995.
3. F. Chaumette. Potential problems of stability and convergence in image-based and position-based visual servoing. D. Kriegman, G. Hager, and A. Morse, editors, *The confluence of vision and control*, Vol. 237 of *LNCIS Series*, pp. 66–78. Springer Verlag, 1998.
4. 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.
5. A. Criminisi, I. Reid, and A. Zisserman. Duality, rigidity and planar parallax. *European Conf. on Computer Vision*, Vol. 2, pp. 846–861, Freiburg, Germany, June 1998.
6. D. Dementhon and L. S. Davis. Model-based object pose in 25 lines of code. *Int. Journal of Computer Vision*, 15(1/2):123–141, June 1995.
7. R. Deriche, Z. Zhang, Q.-T. Luong, and O. Faugeras. Robust recovery of the epipolar geometry for an uncalibrated stereo rig. *European Conf. on Computer Vision*, Stockholm, Sweden, 1994.
8. B. Espiau. Effect of camera calibration errors on visual servoing in robotics. *3rd Int. Symp. on Experimental Robotics*, Kyoto, Japan, October 1993.
9. 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.
10. O. Faugeras. *Three-dimensionnal computer vision: a geometric viewpoint*. MIT Press, Cambridge, Massachusetts, 1993.
11. O. Faugeras and F. Lustman. Motion and structure from motion in a piecewise planar environment. *Int. Journal of Pattern Recognition and Artificial Intelligence*, 2(3):485–508, 1988.

12. R. I. Hartley. Estimation of relative camera positions for uncalibrated cameras. *European Conf. on Computer Vision*, pp. 579–587, Santa Margherita Ligure, Italy, May 1992.
13. R. I. Hartley. In defense of the eight-point algorithm. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(6):580–593, June 1997.
14. K. Hashimoto. *Visual Servoing: Real Time Control of Robot manipulators based on visual sensory feedback*, Vol. 7 of *World Scientific Series in Robotics and Automated Systems*. World Scientific Press, Singapore, 1993.
15. K. Hosoda and M. Asada. Versatile visual servoing without knowledge of true jacobian. *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, Munchen, Germany, September 1994.
16. 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.
17. 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.
18. M. Irani, P. Anadan, and D. Weinshall. From reference frames to reference planes: multi-view parallax geometry and applications. *European Conf. on Computer Vision*, Vol. 2, pp. 829–845, Freiburg, Germany, June 1998.
19. M. Jgersand, O. Fuentes and R. Nelson. Experimental evaluation of uncalibrated visual servoing for precision manipulation. *IEEE Int. Conf. on Robotics and Automation*, Vol. 3, pp. 2874-2880, Albuquerque, New Mexico, April 1997.
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. *1st Conf. on Artificial Intelligence Applications*, pp. 395–397, Denver, 1984.
23. Q.-T. Luong and O. Faugeras. The fundamental matrix: Theory, algorithms, and stability analysis. *Int. Journal of Computer Vision*, 17(1):43–75, January 1996.
24. E. Malis. *Contributions à la modélisation et à la commande en asservissement visuel*. PhD thesis, Université de Rennes I, IRISA, November 1998.
25. E. Malis, F. Chaumette, and S. Boudet. 2 1/2 D Visual Servoing *IEEE Trans. on Robotics and Automation*, 15(2):234-246, April 1999.
26. C. Samson, M. Le Borgne, and B. Espiau. *Robot Control: the Task Function Approach*, Vol. 22 of *Oxford Engineering Science Series*. Clarendon Press, Oxford, England, 1991.
27. P. Torr, A. W. Fitzgibbon, and A. Zisserman. Maintaining multiple motion model hypotheses over many views to recover matching and structure. *IEEE Int. Conf. on Computer Vision*, pp. 485–491, Bombay, India, January 1998.
28. R. Y. Tsai and T. S. Huang. Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 6(1):13–27, January 1984.
29. 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.
30. 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.
31. Z. Zhang and A. R. Hanson. Scaled euclidean 3D reconstruction based on externally uncalibrated cameras. *IEEE Symp. on Computer Vision*, Coral Gables, Florida, 1995.
32. Z. Zhang. Determining the Epipolar Geometry and its Uncertainty - A Review. *Int. Journal of Computer Vision*, 27(2):161-195, March 1998.