Feature Depth Observation for Image-based Visual Servoing: Theory and Experiments

Alessandro De Luca, Giuseppe Oriolo and Paolo Robuffo Giordano

The International Journal of Robotics Research 2008; 27; 1093
DOI: 10.1177/0278364908096706

Alessandro De Luca
Giuseppe Oriolo
Paolo Robuffo Giordano

Dipartimento di Informatica e Sistemistica,
Università di Roma "La Sapienza",
Via Ariosto 25, 00185 Roma, Italy
{deluca, oriolo, robuffo}@dis.uniroma1.it

Feature Depth Observation for Image-based Visual Servoing: Theory and Experiments

Abstract

In the classical image-based visual servoing framework, error signals are directly computed from image feature parameters, allowing, in principle, control schemes to be obtained that need neither a complete three-dimensional (3D) model of the scene nor a perfect camera calibration. However, when the computation of control signals involves the interaction matrix, the current value of some 3D parameters is required for each considered feature, and typically a rough approximation of this value is used. With reference to the case of a point feature, for which the relevant 3D parameter is the depth Z, we propose a visual servoing approach where Z is observed and made available for servoing. This is achieved by interpreting depth as an unmeasurable state with known dynamics, and by building a non-linear observer that asymptotically recovers the actual value of Z for the selected feature. A byproduct of our analysis is the rigorous characterization of camera motions that actually allow such observation. Moreover, in the case of a partially uncalibrated camera, it is possible to exploit complementary camera motions in order to preliminarily estimate the focal length without knowing Z. Simulations and experimental results are presented for a mobile robot with an on-board camera in order to illustrate the benefits of integrating the depth observation within classical visual servoing schemes.

KEY WORDS—Image-Based Visual Servoing, Nonlinear Observers, Depth Observation, Focal Length Observation, Mobile Robots

1. Introduction

The introduction of visual information into the control loop of robot systems has increased the flexibility and the accuracy of the tasks commonly performed by these systems (Espiau et al. 1992; Hutchinson et al. 1996), by providing higher position accuracy, robustness to sensor noise and calibration uncertainties, and reactiveness to environmental changes. This is especially true for the class of mobile robots, where the elaboration of visual cues is often crucial for self-localization and navigation. Another interesting use of visual feedback is the powerful approach known as *visual servoing* where the robotic task is directly specified in terms of some image features extracted from a target object. These features are then used to control the robot/camera motion through the scene. Two main approaches have been proposed in recent years to deal with this kind of task, namely *position-based visual servoing* (PBVS) and *image-based visual servoing* (IBVS) schemes (Hutchinson et al. 1996; Chaumette and Hutchinson 2006a,b).

In PBVS control, the features extracted from the images are used to estimate the relative three-dimensional (3D) pose between the camera and the target (Wilson et al. 1996; Taylor et al. 2000). This error signal can be used by a control law to move the camera/robot system towards its desired pose. Usually, PBVS methods need an *a priori* 3D model of the target in order to reconstruct the relative pose with respect to the camera. In addition, uncertainty in the camera calibration parameters will lead to errors in the 3D reconstruction and, thus, to inaccurate task execution. Finally, there is no direct control on the motion of the features on the image plane, so that an object of interest (for example, the target itself) may leave the field of view during motion, causing the failure of the servoing task.

On the other hand, IBVS methods compute the error signal directly in terms of the features extracted from the image, whose motion on the image plane is related to the linear/angular velocity of the camera through an interaction ma-

The International Journal of Robotics Research
Vol. 27, No. 10, October 2008, pp. 1093–1116
DOI: 10.1177/0278364908096706

Figures 2–6, 8–18 appear in color online: http://ijr.sagepub.com

trix. As a consequence, there is no need for a complete 3D model of the target, and convergence is generally robust with respect to disturbances and uncertainties in the camera/robot model (Espiau 1993). Moreover, direct control of the feature motion on the image plane allows strategies to be devised that aim at always keeping the target in the field of view of the camera (Corke and Hutchinson 2001).

There are, however, some drawbacks to be considered when dealing with IBVS methods. First, the interaction matrix (which depends on the current value of the features and on other 3D information, such as depth) can become singular during the servoing, or false equilibria may be reached due to the presence of unrealizable feature motions (Chaumette 1998). In addition, considering, for example, point features, the actual value of the depth for each point is usually unknown, and some estimate must be used (for example, the constant value at the desired position). Thus, the convergence of the scheme can be guaranteed only locally (Malis and Rives 2003). Local convergence may also result from a rough approximation of the camera intrinsic parameters. If the camera is not accurately calibrated, or if its intrinsic parameters are changing over time, it is still possible to plan a path for the features in a suitable *invariant space* (Malis 2004), but any global convergence property for the servoing scheme is lost.

To alleviate some of the above problems, a number of *hybrid* schemes have been proposed recently (Deguchi 1998; Malis et al. 1999; Morel et al. 2000; Malis and Chaumette 2002). In these methods, 3D information (usually obtained from epipolar geometry considerations) is used to control a subset of the camera configuration vector, while the remaining degrees of freedom (DOFs) are regulated through an IBVS scheme. As a consequence, the robustness domain (that is, the amount of calibration/approximation errors that can be tolerated by the system) is enlarged, and analytical conditions that ensure global stability can also be obtained. Adaptive IBVS schemes have been devised by Conticelli et al. (1999) for a camera mounted on a non-holonomic mobile robot via on-line observation of a constant unknown parameter (the height of the object points), and by Conticelli and Allotta (2001) where the authors treat depth as a time-varying uncertain parameter, proving uniform boundedness of the whole feature/depth error, but not convergence to the desired features. Finally, a recent result proposes a locally stable IBVS control law that does not need any prior information about the 3D structure of the target, but is directly computed in terms of the homography matrix between the current and the desired image (Benhimane and Malis 2006).

In this paper, we address the IBVS stability and convergence issues due to imprecise knowledge of the depths Z of point features, by designing a non-linear observer that recovers the current value of Z during robot/camera motion, and by using it within the visual servoing feedback loop. In fact, the observation framework provides techniques to estimate unmeasurable time-varying states of known dynamical systems. The long-term objective of this approach is to obtain an IBVS scheme independent of any prior 3D knowledge, such as depth or additional structure relative to the target at the robot final pose. This would be relevant, for instance, in the case of navigation/exploration tasks for mobile robots. While exploring the environment, a robot could store several images of interesting locations without the need of extracting 3D information, a procedure which can in some cases be difficult or even impossible. When asked to reach again the stored locations, the robot would have the possibility to recover the missing information on-line thanks to the proposed observer, thus greatly enhancing its capability to fulfill the visual task.

The observation of the depth is based on the idea that, since the motion of a point feature on the image plane depends upon the current value of its depth, it is possible to estimate this value by comparing the measured feature motion on the image plane with that predicted by using the current estimate of Z, under the assumption of a perfect knowledge of the camera 3D motion and of its intrinsic parameters. This is one instance of the more general paradigm of *motion and structure reconstruction*, whose purpose is to design identification schemes to estimate both the camera motion and the structure (that is, the 3D geometry) of the scene. Our work assumes a known relative motion between the camera and the target, which can be achieved, for example, if the feature is fixed in the world and the camera is mounted on the end-effector of a robot manipulator. Furthermore, we show that the hypothesis of a perfect calibrated camera can be partially relaxed. As a matter of fact, the same structure of the observer can be extended to also cover the case of identification of the focal length. This can be performed preliminarily (and once for all), without requiring the knowledge of Z.

In recent years, several works have addressed structure identification with known motion. Chaumette et al. (1996) propose a general methodology to recover the 3D information of several geometric primitives (points, lines, cylinders, spheres, etc.) by measuring the current values of the features, of the image motion (the feature time derivatives) and of the camera velocity twist. However, owing to the presence of noise and discrete sampling, the extraction of the image motion is not trivial, and some constraints on the allowed camera motions must be considered. Matthies et al. (1989) derived and compared two algorithms based on a Kalman filter, with the first algorithm estimating a continuous depth map of the scene and the second extracting the depth of a discrete set of features. Both methods need the computation of the current image motion, and impose several constraints on the camera motion in order to simplify the problem. In particular, the second method assumes a camera which translates orthogonally to the optical axis (without rotations), so that the depth of the features is kept constant and the problem is simplified considerably. A similar technique is found in the work of Smith and Papanikolopoulos (1994), where, again, only lateral camera motions are allowed. With respect to these works, our contribution is that

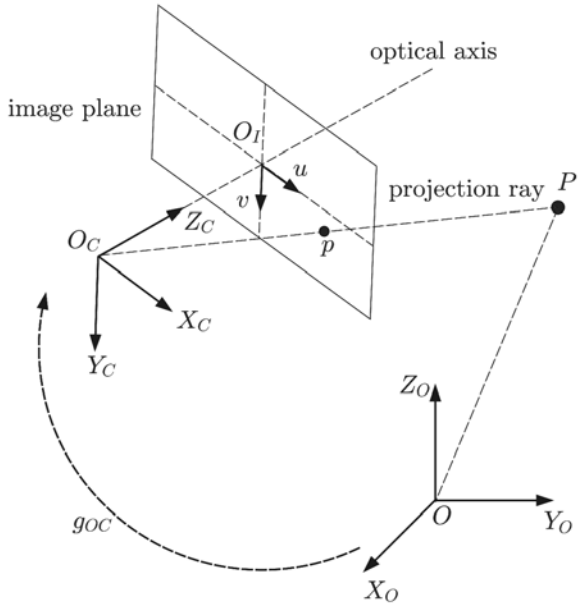


Fig. 1. World and camera frame definitions.

we solve the problem of depth reconstruction for static point features without any special constraints on the camera motion, and without the need for image motion estimation; the only information used is the current value of the features measured on the image plane.

The paper is organized as follows. In Section 2 we recall the basic kinematic and perspective relationships of the camera/target system. In Section 3 we design a non-linear observer to estimate the unknown value of Z, while in Section 4 the extension to cover the unknown focal length case is considered. The performance of the proposed observers is evaluated in Section 5 by means of simulations. Section 6 discusses the integration of the proposed observer in a visual servoing loop. Finally, Section 7 presents a collection of experiments for a mobile robot equipped with an on-board camera, showing the performance of the integrated observer-based IBVS scheme.

2. Perspective Camera Model

In this section we give an overview of the basic kinematic and perspective relationships modeling a camera which moves through the scene. Most of the concepts can be found, for example, in the work of Murray et al. (1994) and Ma et al. (2004).

2.1. Kinematic Modeling

With reference to Figure 1, consider an inertial world reference frame $\mathcal{F}_O : \{O;\ X_O,\ Y_O,\ Z_O\}$ and a pin-hole camera associated with the moving frame $\mathcal{F}_C : \{O_C;\ X_C,\ Y_C,\ Z_C\}$, with Z_C coincident with the camera optical axis. The image plane, perpendicular to the optical axis, lies at a distance λ (the focal length) from O_C, and is endowed with a 2D reference frame $\mathcal{F}_I : \{O_I;\ u,\ v\}$ with axes parallel to X_C and Y_C, respectively. The 3D pose of $\mathcal{F}_C$ with respect to $\mathcal{F}_O$ is an element g_{OC} of the group $SE(3)$ with homogeneous matrix representation

$$\bar{g}_{OC} = \begin{bmatrix} R_{OC} & {}^{O}T_{OC} \\ 0 & 1 \end{bmatrix} \in \mathbb{R}^{4\times 4},$$

where $R_{OC} \in SO(3)$ is the rotation matrix of $\mathcal{F}_C$ with respect to $\mathcal{F}_O$, and ${}^{O}T_{OC} \in \mathbb{R}^3$ is the vector from O to O_C expressed in $\mathcal{F}_O$. Therefore, the coordinates of a point $P \in \mathbb{R}^3$ in $\mathcal{F}_C$ are related to $\mathcal{F}_O$ by the formula

$${}^{O}\bar{P} = \bar{g}_{OC}\,{}^{C}\bar{P}, \tag{1}$$

with $\bar{P} = [X\ Y\ Z\ 1]^{\mathrm{T}} \in \mathbb{R}^4$ being the homogeneous representation of P. The inverse element of g_{OC}, denoted as $g_{OC}^{-1} = g_{CO}$, takes the matrix form

$$\bar{g}_{CO} = \begin{bmatrix} R_{OC}^{T} & -R_{OC}^{T}\,{}^{O}T_{OC} \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} R_{CO} & -R_{CO}\,{}^{O}T_{OC} \\ 0 & 1 \end{bmatrix}.$$

The velocity of point P in $\mathcal{F}_C$ can be found by inverting and differentiating (1) with respect to time. Indeed, the time derivative of ${}^{C}\bar{P} = \bar{g}_{CO}\,{}^{O}\bar{P}$ is

$$\begin{aligned} {}^{C}\dot{\bar{P}} &= \dot{\bar{g}}_{CO}\,{}^{O}\bar{P} + \bar{g}_{CO}\,{}^{O}\dot{\bar{P}} = \dot{\bar{g}}_{CO}\bar{g}_{OC}\,{}^{C}\bar{P} + \bar{g}_{CO}\,{}^{O}\dot{\bar{P}} \\ &= {}^{C}\widehat{V}_{CO}\,{}^{C}\bar{P} + \bar{g}_{CO}\,{}^{O}\dot{\bar{T}}_{P}. \end{aligned} \tag{2}$$

In this equation, ${}^{O}\dot{\bar{T}}_{P} \in \mathbb{R}^4$ is the homogeneous representation[1] of the absolute velocity of point P in $\mathcal{F}_O$, and ${}^{C}\widehat{V}_{CO} = \dot{\bar{g}}_{CO}\bar{g}_{OC}$, called *twist*, is an element of the Lie algebra $se(3)$ of the matrix group $SE(3)$, representing the linear/angular velocity of $\mathcal{F}_O$ with respect to $\mathcal{F}_C$ expressed in $\mathcal{F}_C$. The homogeneous matrix representation of ${}^{C}\widehat{V}_{CO}$ is

$$\begin{aligned} {}^{C}\widehat{V}_{CO} &= \begin{bmatrix} \dot{R}_{CO}R_{OC} & -R_{CO}\,{}^{O}\dot{T}_{OC} \\ 0 & 0 \end{bmatrix} \\ &= \begin{bmatrix} {}^{C}\widehat{\omega}_{CO} & -{}^{C}\dot{T}_{OC} \\ 0 & 0 \end{bmatrix}, \end{aligned}$$

with ${}^{C}\widehat{\omega}_{CO} \in so(3)$ the skew-symmetric matrix associated with the angular velocity ${}^{C}\omega_{CO} \in \mathbb{R}^3$. By expanding (2) we finally obtain

1. We recall that the homogeneous representation of a vector $v \in \mathbb{R}^3$ is $\bar{v} = [v_x\ v_y\ v_z\ 0]^{\mathrm{T}}$.

$$
{}^{C}\dot{\bar{P}} = \begin{bmatrix} R_{CO}\,{}^{O}\dot{T}_P - {}^{C}\dot{T}_{OC} + {}^{C}\widehat{\omega}_{CO}\,{}^{C}P \\ 0 \end{bmatrix}
= \begin{bmatrix} {}^{C}\dot{T}_P - {}^{C}\dot{T}_{OC} - {}^{C}\widehat{\omega}_{OC}\,{}^{C}P \\ 0 \end{bmatrix}. \tag{3}
$$

To simplify the notation, the dependency on $\mathcal{F}_C$ is dropped in the following, since we always refer to quantities expressed in the camera frame, unless otherwise stated. Furthermore, by letting ${}^{C}\dot{T}_P = v_P$, ${}^{C}\dot{T}_{OC} = v_C$ and ${}^{C}\omega_{OC} = \omega_C$, we can rearrange (3) in a more convenient matrix form

$$
\begin{bmatrix} \dot{X} \\ \dot{Y} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} -1 & 0 & 0 & 0 & -Z & Y \\ 0 & -1 & 0 & Z & 0 & -X \\ 0 & 0 & -1 & -Y & X & 0 \end{bmatrix} \times \begin{bmatrix} v_C - v_P \\ \omega_C \end{bmatrix} \tag{4}
$$

which will be useful in the next developments. Note that v_C and ω_C (the camera linear and angular velocity) can be seen as the control inputs of the system, while v_P (the world velocity of point P) represents an exogenous quantity. Throughout the paper, we always assume that $v_P = 0$, that is, that point P is fixed in the scene.

2.2. Image and Camera Model

In the computer vision literature, an *image feature* is a real-valued quantity associated to a geometric primitive (for example, the coordinates of a point, the area of an ellipse, the angular coefficient of a line, etc.) in the image plane. Given a vector of features $f = [f_1 \dots f_k]^{\mathrm{T}} \in \mathbb{R}^k$, the velocity twist $(v_C,\ \omega_C)$ of the camera is mapped to $\dot{f}$ by a $k \times 6$ matrix $J_f(f,\ \chi)$ called the *interaction matrix*

$$
\dot{f} = J_f(f, \chi) \begin{bmatrix} v_C \\ \omega_C \end{bmatrix}, \tag{5}
$$

where χ is a vector representing 3D information associated to f. It is possible to determine the interaction matrix for many features of interest, see Espiau et al. (1992) for the case of points, lines, planes, circles, etc., and and Chaumette (2004) and Tahri and Chaumette (2005) for the set of image moments. In the case of a 3D point feature P with homogeneous coordinates $\bar{P} = [X\ Y\ Z\ 1]^{\mathrm{T}}$, its projection on the image plane is a 2D point feature p with homogeneous normalized coordinates $\bar{p} = [\bar{p}_u\ \bar{p}_v\ 1]^{\mathrm{T}} = [X/Z\ Y/Z\ 1]^{\mathrm{T}}$. The corresponding coordinates in pixels are denoted $\widetilde{p} = [\widetilde{p}_u\ \widetilde{p}_v\ 1]^{\mathrm{T}} = A\bar{p}$, where A is a non-singular matrix containing the camera intrinsic parameters:

$$
A = \begin{bmatrix} \widetilde{\lambda} k_u & -\widetilde{\lambda} k_u / \tan\delta & u_0 \\ 0 & \widetilde{\lambda} k_v / \sin\delta & v_0 \\ 0 & 0 & 1 \end{bmatrix}. \tag{6}
$$

Here, $[u_0\ v_0]^{\mathrm{T}}$ are the coordinates of the principal point (in pixels), $\widetilde{\lambda}$ is the focal length (in meters), k_u and k_v are the magnifications in the u and v directions (in pixels per meter), and δ is the angle between these axes. Assuming, as is usually done, that $\delta = \pi/2$ and $k_u = k_v$, we can rewrite (6) as

$$
A = \begin{bmatrix} \widetilde{\lambda} k_u & 0 & u_0 \\ 0 & \widetilde{\lambda} k_u & v_0 \\ 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \lambda & 0 & u_0 \\ 0 & \lambda & v_0 \\ 0 & 0 & 1 \end{bmatrix} \tag{7}
$$

where $\lambda = \widetilde{\lambda} k_u$ is the focal length in pixels. If matrix A is known, that is, the camera is calibrated, it is always possible to transform the measured point $\widetilde{p}$ back to $\bar{p} = A^{-1}\widetilde{p}$, and thus derive the expression of the interaction matrix in this completely "normalized" space (see Hutchinson et al. (1996) for an explicit derivation). However, to obtain a relationship which can be used also in the case of a partially uncalibrated camera, we only assume that the values of $(u_0,\ v_0)$ are known. Hence, by defining $f_u = \widetilde{p}_u - u_0$ and $f_v = \widetilde{p}_v - v_0$, that is, centering the pixel coordinates of $\widetilde{p}$ with respect to the camera principal point, we use the partially normalized relationship

$$
\begin{bmatrix} \dot{f}_u \\ \dot{f}_v \end{bmatrix} = \begin{bmatrix} -\frac{\lambda}{Z} & 0 & \frac{f_u}{Z} & \frac{f_u f_v}{\lambda} & -\left(\lambda + \frac{f_u^2}{\lambda}\right) & f_v \\ 0 & -\frac{\lambda}{Z} & \frac{f_v}{Z} & \lambda + \frac{f_v^2}{\lambda} & -\frac{f_u f_v}{\lambda} & -f_u \end{bmatrix} \times \begin{bmatrix} v_C \\ \omega_C \end{bmatrix} = J_f(f_u,\ f_v,\ Z) \begin{bmatrix} v_C \\ \omega_C \end{bmatrix}. \tag{8}
$$

The 2×6 matrix $J_f(f_u,\ f_v,\ Z)$ is referred to as the *interaction matrix* of a point feature.

Note that, in the case of point features, the 3D information represented by χ simply reduces to the point depth Z. Moreover, since only the first three columns of J_f are affected by the value of Z, a pure camera rotation does not bring any useful information for depth observation: a camera translation must be necessarily present. This intuitive conclusion, already established by Inaba et al. (2000) in the context of the observability of dynamical systems with perspective outputs, are reobtained in Section 3 as a byproduct of the *persistency of excitation* condition.

3. Observation of a Feature Depth

The purpose of this section is to design a non-linear observer which estimates the value of Z during the motion of the camera. We first assume a calibrated camera, that is, that the value of λ is known. The resulting observer was preliminarily presented by De Luca et al. (2007b).

It is convenient to rewrite (4) and (8) in a more familiar form. Let $x = [f_u \ f_v \ Z]^{\mathrm{T}} \in \mathbb{R}^3$ be the state vector and $u = [v_C^T \ \omega_C^T]^{\mathrm{T}} \in \mathbb{R}^6$ be the input vector. Hence, using (8) and the last row of (4), the state dynamics are expressed by the driftless system

$$\begin{aligned} \dot{x} &= \begin{bmatrix} -\frac{\lambda}{x_3} & 0 & \frac{x_1}{x_3} & \frac{x_1 x_2}{\lambda} & -\left(\lambda + \frac{x_1^2}{\lambda}\right) & x_2 \\ 0 & -\frac{\lambda}{x_3} & \frac{x_2}{x_3} & \lambda + \frac{x_2^2}{\lambda} & -\frac{x_1 x_2}{\lambda} & -x_1 \\ 0 & 0 & -1 & -\frac{x_2 x_3}{\lambda} & \frac{x_1 x_3}{\lambda} & 0 \end{bmatrix} u \\ y &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \end{aligned} \tag{9}$$

where the output vector $y \in \mathbb{R}^2$ represents the measurable variables, that is, the coordinates of the point p on the image plane. Consider the change of coordinates (see also Conticelli and Allotta (2001))

$$\widetilde{x} = \begin{bmatrix} x_1 \\ x_2 \\ \dfrac{1}{x_3} \end{bmatrix},$$

which is globally defined since $x_3(t) > \widetilde{\lambda} > 0$ (that is, the point P is supposed to lie always in front of the image plane, otherwise the camera sensor, and hence the visual servoing, would fail). In the new coordinates, system (9) becomes

$$\begin{aligned} \dot{\widetilde{x}} &= \begin{bmatrix} -\lambda\widetilde{x}_3 & 0 & \widetilde{x}_1\widetilde{x}_3 & \frac{\widetilde{x}_1\widetilde{x}_2}{\lambda} & -\left(\lambda + \frac{\widetilde{x}_1^2}{\lambda}\right) & \widetilde{x}_2 \\ 0 & -\lambda\widetilde{x}_3 & \widetilde{x}_2\widetilde{x}_3 & \lambda + \frac{\widetilde{x}_2^2}{\lambda} & -\frac{\widetilde{x}_1\widetilde{x}_2}{\lambda} & -\widetilde{x}_1 \\ 0 & 0 & \widetilde{x}_3^2 & \frac{\widetilde{x}_2\widetilde{x}_3}{\lambda} & -\frac{\widetilde{x}_1\widetilde{x}_3}{\lambda} & 0 \end{bmatrix} u \\ y &= \begin{bmatrix} \widetilde{x}_1 \\ \widetilde{x}_2 \end{bmatrix}. \end{aligned} \tag{10}$$

Since (10) is driftless, its time-invariant linear approximation at any point is unobservable and admits no standard linear observer. As a consequence, a more general class of observation schemes must be considered in order to solve the estimation problem.

Let $\widehat{x} \in \mathbb{R}^3$ be the estimate of the (partially) unknown state $\widetilde{x}$. We seek an update law in the form

$$\dot{\widehat{x}} = \alpha(\widehat{x}, y)u + \beta(\widehat{x}, y, u) \tag{11}$$

which guarantees $\lim_{t\to\infty} \|\widetilde{x}(t) - \widehat{x}(t)\| = 0$, $\forall\, \widehat{x}(t_0)$. Drawing a parallel with standard observers in the linear domain, the term $\alpha(\widehat{x}, y)u$ plays the role of a copy of the original system (10), while $\beta(\widehat{x}, y, u)$ provides the feedback action needed to recover the observed states. The design of the functions $\alpha(\widehat{x}, y)$ and $\beta(\widehat{x}, y, u)$ will be based on the following result, known as the *persistency of excitation lemma*, a proof of which was given by Marino and Tomei (1995).

Lemma 1. Consider the linear time-varying system

$$\begin{cases} \dot{\xi} &= H\xi + \Omega^T(t)z, \quad \xi \in \mathbb{R}^n \\ \dot{z} &= -\Lambda\,\Omega(t)P\xi, \quad z \in \mathbb{R}^p \end{cases} \tag{12}$$

where H is an $n \times n$ Hurwitz matrix, P is an $n \times n$ symmetric positive-definite matrix such that $H^T P + PH = -Q$, with Q symmetric positive definite, and Λ is a $p \times p$ symmetric positive-definite matrix. If $\|\Omega(t)\|$, $\|\dot{\Omega}(t)\|$ are uniformly bounded and the *persistency of excitation* condition is satisfied, that is, there exist two positive real numbers T and γ such that

$$\int_t^{t+T} \Omega(\tau)\Omega^T(\tau)\,\mathrm{d}\tau \geq \gamma I > 0, \quad \forall\, t \geq t_0, \tag{13}$$

then $(\xi, z) = 0$ is a globally exponentially stable equilibrium point.

We now perform some manipulation in order to be able to apply Lemma 1 to our case. Let $e = \widetilde{x} - \widehat{x}$ be the error vector, and note that the subvector $[e_1 \ e_2]^{\mathrm{T}}$ is directly accessible for measurements. Thus, if we define the observer as in (11) with

$$\begin{aligned} \alpha(\widehat{x}, y) &= \begin{bmatrix} -\lambda\widehat{x}_3 & 0 & y_1\widehat{x}_3 & \frac{y_1 y_2}{\lambda} & -\left(\lambda + \frac{y_1^2}{\lambda}\right) & y_2 \\ 0 & -\lambda\widehat{x}_3 & y_2\widehat{x}_3 & \lambda + \frac{y_2^2}{\lambda} & -\frac{y_1 y_2}{\lambda} & -y_1 \\ 0 & 0 & \widehat{x}_3^2 & \frac{y_2\widehat{x}_3}{\lambda} & -\frac{y_1\widehat{x}_3}{\lambda} & 0 \end{bmatrix} \\ \beta(\widehat{x}, y, u) &= \begin{bmatrix} k_1 e_1 \\ k_2 e_2 \\ \left\{ \begin{array}{l} k_3((-\lambda u_1 + y_1 u_3)e_1 \\ \quad +(-\lambda u_2 + y_2 u_3)e_2) \end{array} \right\} \end{bmatrix} \end{aligned} \tag{14}$$

with $k_1, k_2, k_3 > 0$, we obtain the error dynamics

$$\begin{aligned}
\dot{e}_1 &= -k_1 e_1 + (-\lambda u_1 + y_1 u_3) e_3 \\
\dot{e}_2 &= -k_2 e_2 + (-\lambda u_2 + y_2 u_3) e_3 \\
\dot{e}_3 &= -k_3 \begin{bmatrix} -\lambda u_1 + y_1 u_3 & -\lambda u_2 + y_2 u_3 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} \\
&+ (\tilde{x}_3^2 - \hat{x}_3^2) u_3 + \left(\frac{y_2 u_4 - y_1 u_5}{\lambda} \right) e_3. \qquad (15)
\end{aligned}$$

By setting

$$\begin{aligned}
\xi &= [e_1 \; e_2]^{\mathrm{T}} \\
z &= e_3 \\
H &= \begin{bmatrix} -k_1 & 0 \\ 0 & -k_2 \end{bmatrix} \\
\Omega(t) &= [-\lambda u_1 + y_1 u_3 \quad -\lambda u_2 + y_2 u_3] \\
\Lambda &= k_3 \\
P &= I, \qquad (16)
\end{aligned}$$

system (15) is very close to the formulation in (12), the only difference being the last two terms in the e_3 dynamics.

It is worth noting that, when

$$u_3(t) \equiv u_4(t) \equiv u_5(t) \equiv 0, \qquad (17)$$

the two formulations match exactly and the global exponential stability of e is guaranteed, as long as the conditions of Lemma 1 are met. While we thoroughly discuss such conditions in the forthcoming analysis, we would like to emphasize that (17) corresponds to a camera motion which keeps the depth Z constant. As explained in Section 1, in this case the problem is considerably simplified and can be attacked with various techniques. The purpose of our analysis is to show that (15) can converge also when (17) does not hold.

Proposition 1. Assume that $\Omega(t)$ as given by (16) satisfies the persistency of excitation (13). Then, by using the observer (11)–(14), the origin of the error system (15) can be made globally exponentially stable.

Proof. Rewrite (15) as $\dot{e} = A(t)e + g(e, t)$ with

$$A(t) = \begin{bmatrix} -k_1 & 0 & \Omega_1(t) \\ 0 & -k_2 & \Omega_2(t) \\ -k_3\Omega_1(t) & -k_3\Omega_2(t) & 0 \end{bmatrix}$$

$$g(e, t) = \begin{bmatrix} 0 \\ 0 \\ \left(2\tilde{x}_3 u_3 + \frac{y_2 u_4 - y_1 u_5}{\lambda}\right) e_3 - u_3 e_3^2 \end{bmatrix} \qquad (18)$$

We can consider $g(e, t)$ as a perturbation on the nominal system $\dot{e} = A(t)e$ which, if (13) holds, is guaranteed by Lemma 1 to be globally exponentially stable. In fact, the uniform boundedness of $\|\Omega(t)\|$ and $\|\dot{\Omega}(t)\|$ required in Lemma 1 is ensured since the image plane has a finite size, and the camera linear velocity and acceleration are obviously limited.

Converse Lyapunov theorems (see, for example, Khalil (1996, Theorem 3.12)) can be used to claim that the nominal system admits a Lyapunov function $V(e, t)$ such that

$$\begin{aligned}
c_1\|e\|^2 &\leq V \leq c_2\|e\|^2 \\
\dot{V}(e, t) &= \frac{\partial V}{\partial t} + \frac{\partial V}{\partial e} A(t) e \leq -c_3\|e\|^2 \\
\left\| \frac{\partial V}{\partial e} \right\| &\leq c_4\|e\|,
\end{aligned}$$

with $c_1 \ldots c_4$ positive constants. At this point, we exploit the fact that $g(e, t)$ is a vanishing perturbation, that is, $g(0, t) = 0, \; \forall\, t$. In particular, let $S_c = \{e \mid V(e, t) \leq c\}$ be any level set of function V. Since V is radially unbounded, S_c is a compact set. Owing to the boundedness of $u(t)$, $g(e, t)$ is (locally) Lipschitz and there exists a positive constant M such that $\|g(e, t)\| \leq M\|e\|$ in S_c. Using V as a Lyapunov candidate for the perturbed system, we obtain

$$\dot{V}(e, t) \leq -c_3\|e\|^2 + \left\| \frac{\partial V}{\partial e} \right\| \|g(e, t)\| \leq -c_3\|e\|^2 + c_4 M\|e\|^2.$$

In view of the structure of $g(e, t)$ in (18), if $|u_3(t)|$, $|u_4(t)|$ and $|u_5(t)|$ are sufficiently small, one has $M < c_3/c_4$ so that $\dot{V}$ is negative definite on S_c. Since S_c is an arbitrary compact set, the origin of system (15) is globally exponentially stable. ■

This result shows that to guarantee global exponential convergence it is sufficient that u_3, u_4 and u_5 are small enough. This can be taken into account in a visual servoing controller based on the proposed observation scheme by suitably scaling down these velocity inputs, when needed. On the other hand, if u_3, u_4 and u_5 are exogenous signals, one can still guarantee that $M < c_3/c_4$ (and, thus, the validity of the proof) if the observer (in particular, $\hat{x}_3$; see (18)) is initialized sufficiently close to the true value to be estimated. This means that only local asymptotic stability is obtained for arbitrary inputs u_3, u_4 and u_5.

Proposition 1 demonstrates the possibility to recover the depth by exploiting the known camera motion and the measured position of the point feature on the image plane. To this end, the persistency of excitation condition (13) must hold, that

is, there must not exist a $\bar{t}$ such that $\forall\, t > \bar{t}$, $\|\Omega(t)\| \equiv 0$. By direct inspection of the expression of $\Omega(t)$, we can conclude that the persistency of excitation condition for depth observation is violated if and only if

1. $\exists\, \bar{t} \mid \forall\, t > \bar{t}$: $u_1(t) \equiv 0$, $u_2(t) \equiv 0$, $u_3(t) \equiv 0$, that is, no translations are involved in the camera motion;
2. $\exists\, \bar{t} \mid \forall\, t > \bar{t}$: $\lambda u_1 = y_1 u_3$, $\lambda u_2 = y_2 u_3$, which is equivalent to
$$\frac{u_1}{u_3} = \frac{X}{Z}, \quad \frac{u_2}{u_3} = \frac{Y}{Z},$$
that is, the camera is translating along the projection ray of the selected point p.

It is interesting to note that such a persistency of excitation condition, essential for the observation convergence, is basically due to the scale ambiguity present in every perspective system. Indeed, it is well known (see, for example, Ma et al. (2004)) that within a perspective system it is impossible to distinguish between an object and the same object twice as large, twice as far away and moving twice as fast. The condition of non-zero (and known) camera translational velocity introduces a scale information which is essential to disambiguate among all of the equivalent states and to successfully recover the actual feature depth. Note that, in this case, such a property also implies the *necessity* of Lemma 1 requirements (which in general are only sufficient), that is, the depth can be recovered *if and only if* (13) holds.

4. Observation of the Focal Length

As discussed in the previous section, pure camera rotations do not allow depth observation because in this case the feature motion does not depend on Z. However, this motion depends on λ: therefore, it is possible to exploit the same observer design approach to estimate the constant value of λ without being affected by uncertainties on Z.

Assume a pure rotational motion of the camera. With reference to (8), let $x = [f_u \;\; f_v \;\; \lambda \;\; 1/\lambda]^{\mathrm{T}} \in \mathbb{R}^4$ be the state vector and $u = \omega_C \in \mathbb{R}^3$ be the input vector. The dynamic equations for this case are then expressed by the driftless system

$$\begin{aligned} \dot{x} &= \begin{bmatrix} x_1 x_2 x_4 & -(x_3 + x_1^2 x_4) & x_2 \\ x_3 + x_2^2 x_4 & -x_1 x_2 x_4 & -x_1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} u \\ y &= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \end{aligned} \tag{19}$$

The introduction of the component $x_4 = 1/\lambda$, which at first glance may seem unnecessary, is aimed at obtaining a linear dependence on the unmeasurable states in (19).

Proceeding as in the previous section, we define $\hat{x} \in \mathbb{R}^4$ as the current estimate of x, and seek an update law in the form (11) so as to guarantee $\lim_{t\to\infty} \|x(t) - \hat{x}(t)\| = 0, \forall\, \hat{x}(t_0)$. By defining $e = x - \hat{x}$ as the observation error vector, and

$$\begin{aligned} \alpha(\hat{x}, y) &= \begin{bmatrix} y_1 y_2 \hat{x}_4 & -(\hat{x}_3 + y_1^2 \hat{x}_4) & y_2 \\ \hat{x}_3 + y_2^2 \hat{x}_4 & -y_1 y_2 \hat{x}_4 & -y_1 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \\ \beta(\hat{x}, y, u) &= \begin{bmatrix} k_1 e_1 \\ k_2 e_2 \\ k_3(-u_5 e_1 + u_4 e_2) \\ \left\{ \begin{array}{l} k_4((y_1 y_2 u_4 - y_1^2 u_5) e_1 \\ +(y_2^2 u_4 - y_1 y_2 u_5) e_2), \end{array} \right\} \end{bmatrix} \end{aligned} \tag{20}$$

with $k_1, k_2, k_3, k_4 > 0$, we obtain the error dynamics

$$\begin{aligned} \begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \end{bmatrix} &= \begin{bmatrix} -k_1 & 0 \\ 0 & -k_2 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} \\ &\quad + \begin{bmatrix} -u_5 & y_1 y_2 u_4 - y_1^2 u_5 \\ u_4 & y_2^2 u_4 - y_1 y_2 u_5 \end{bmatrix} \begin{bmatrix} e_3 \\ e_4 \end{bmatrix} \\ \begin{bmatrix} \dot{e}_3 \\ \dot{e}_4 \end{bmatrix} &= -\begin{bmatrix} k_3 & 0 \\ 0 & k_4 \end{bmatrix} \\ &\quad \times \begin{bmatrix} -u_5 & u_4 \\ y_1 y_2 u_4 - y_1^2 u_5 & y_2^2 u_4 - y_1 y_2 u_5 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \end{bmatrix}. \end{aligned} \tag{21}$$

Therefore, if we let

$$\begin{aligned} \xi &= [e_1 \; e_2]^{\mathrm{T}} \\ z &= [e_3 \; e_4]^{\mathrm{T}} \\ H &= \begin{bmatrix} -k_1 & 0 \\ 0 & -k_2 \end{bmatrix} \end{aligned}$$

$$\Omega(t) = \begin{bmatrix} -u_5 & u_4 \\ y_1 y_2 u_4 - y_1^2 u_5 & y_2^2 u_4 - y_1 y_2 u_5 \end{bmatrix}$$

$$\Lambda = \begin{bmatrix} k_3 & 0 \\ 0 & k_4 \end{bmatrix}$$

$$P = I, \tag{22}$$

we obtain a formulation that exactly matches (12). Hence, global convergence to the origin of (21) is guaranteed under the same assumptions of Proposition 1.

In this case, however, the physical meaning of the persistency of excitation (13) is not as evident as in the previous section. Indeed, since $\Omega(t)$ in (22) is a square matrix rather than a vector as in (16), condition (13) is violated if and only if there exists a $\bar{t}$ such that

$$\det \Omega(t) = y_1 y_2 u_5^2 - y_2^2 u_4 u_5 - y_1 y_2 u_4^2 + y_1^2 u_4 u_5 \equiv 0, \quad \forall\, t > \bar{t}. \tag{23}$$

To gain insight into (23), rewrite it in terms of 3D coordinates of point P, that is,

$$\frac{f^2}{Z^2}(Y u_4 - X u_5)(X u_4 + Y u_5) = \frac{f^2}{Z^2}\,\eta_1\,\eta_2 \equiv 0, \quad \forall\, t > \bar{t}. \tag{24}$$

Recalling that $P = [X\ Y\ Z]^{\mathrm{T}}$ and $\omega_C = [u_4\ u_5\ u_6]^{\mathrm{T}}$, it is easy to see that η_1 is the third component of vector $\omega_C \times P$, while η_2 is the third component of vector $\omega_C \times P_{\pi/2}$, where $P_{\pi/2} = [-Y\ X\ Z]^{\mathrm{T}}$ is point P rotated $\pi/2$ radians about the camera Z_C axis. Therefore, the persistency of excitation condition for focal length observation is met if and only if:

1. a camera rotation about its X_C and/or Y_C axes is present, that is, $u_4 \neq 0$ or $u_5 \neq 0$ (note that a rotation about the optical axis Z_C is useless for the observation);
2. the camera rotation ω_C induces on points P and $P_{\pi/2}$ a non-zero linear velocity along the Z_C axis, that is, these two points must either become closer to or further from the image plane as a consequence of the camera rotation.

Obviously, there exists a number of more sophisticated (and usually off-line) tools for solving the camera calibration problem, see, for example, Sepp et al. (2005), Strobl and Paredes (2005) and Bouguet (2007). A benefit of our observation scheme for focal length is that it can be used, in a sequential fashion, together with the depth observer discussed in the previous section. Indeed, thanks to the complementarity of the camera motions useful for depth and for focal length observation, one can preliminarily estimate the focal length without knowing Z and then solve the depth observation problem.

5. Simulation Results with the Observers

In the following, we present some simulation results to illustrate the properties of the observation techniques developed so far. The simulations are implemented in MATLAB (Mathworks 2007) and Webots[2] (Cyberbotics 2007) environments, under the assumption that the camera is carried by the end-effector of a robot system which can provide the chosen linear/angular motion.

5.1. Depth Observation

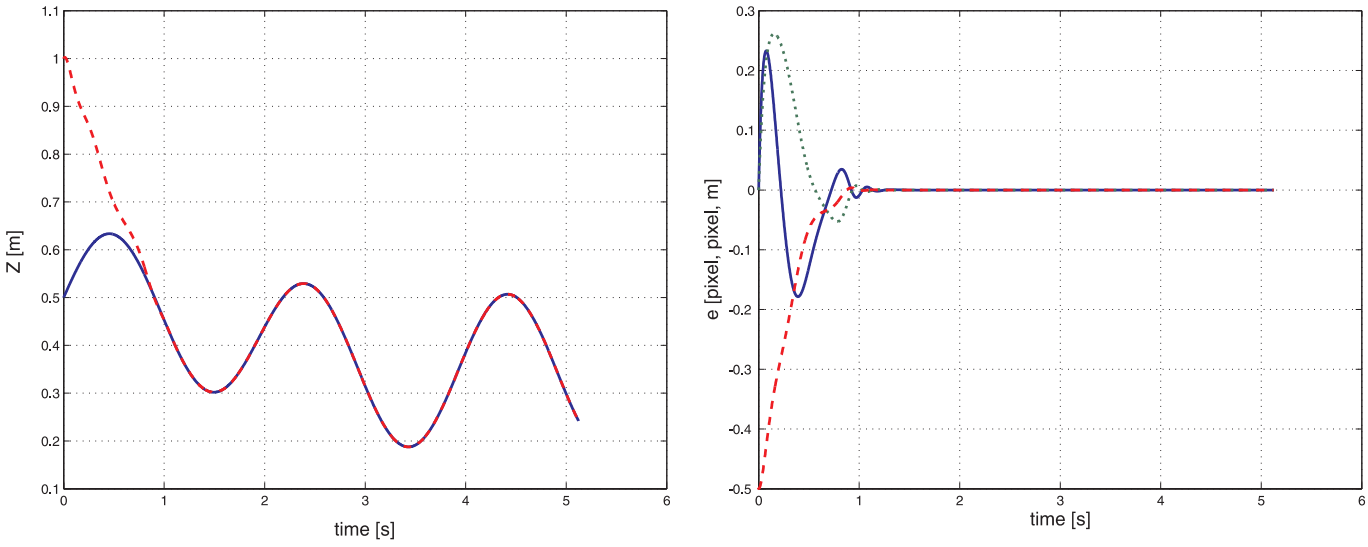
In order to show the performance of the observer (11)–(14) derived in Section 3, we analyze the case of a camera translating and a rotating about the X and Z axes, a case that, for example, could not be addressed with the methods of Matthies et al. (1989) or Smith and Papanikolopoulos (1994) (see Section 1). The simulation data are:

$$\begin{aligned} \widetilde{x}(t_0) &= [10 \quad -10 \quad 2]^{\mathrm{T}} \\ \widehat{x}(t_0) &= [10 \quad -10 \quad 1]^{\mathrm{T}} \\ u_1(t) &= 0.1 \cos 2\pi t \\ u_3(t) &= 0.5 \cos \pi t \\ u_4(t) &= 0.6 \cos \pi/2\, t \\ u_6(t) &= 1 \\ k_1 &= 20 \\ k_2 &= 20 \\ k_3 &= 0.5 \\ \lambda &= 128 \end{aligned}$$

Figure 2 depicts the behavior of $e(t)$ during the simulation and shows how the estimate of Z approaches the true value. Note that the first two components of the observation error are initially zero because the feature position is measured. Practically zero error is reached after 1 [s] of motion.

As an additional case study, we implemented the proposed algorithm in the Webots environment by considering a camera with $\lambda = 128$ pixels mounted on the end-effector of a mobile manipulator made of a unicycle-like platform carrying a 3R spatial arm (see Figure 3). The idea was to test the performance of the proposed observer against the noise automatically introduced by the Webots engine. This noise is added on the image perceived by the camera and thus directly reflects on the feature extraction process. The objective is to estimate the depth

2. Webots is a commercial robot simulation software developed by Cyberbotics Ltd. Its simulation engine automatically takes into account the presence of image noise as well as motion disturbances.



(a) (b)

Fig. 2. Depth observer (MATLAB simulation): (a) true (solid blue line) and estimated (dashed red line) Z; (b) behavior of e_1 (solid blue line), e_2 (dotted green line) and e_3 (dashed red line) versus time.

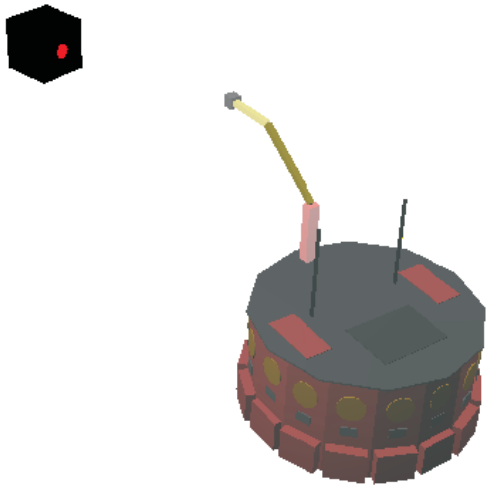


Fig. 3. Webots environment with a mobile manipulator carrying a camera mounted on the end-effector.

of the target point (the red dot on the cube in Figure 3), while the first and second link of the manipulator move according to the velocity profiles:

$$\dot{q}_1 = 0.2 \sin 0.4\pi t$$

$$\dot{q}_2 = 0.1 \sin 0.8\pi t.$$

The initial value of the estimated depth is set at $1/\hat{x}_3(t_0) = 0.05$ and the gains were chosen as $k_1 = k_2 = 10$ and $k_3 = 0.8$. Despite the noise, the observer is able to estimate the actual value of the depth Z accurately, as shown in Figure 4. A video clip of this simulation can be found at http://www.dis.uniroma1.it/∼labrob/research/depth_est.html.

5.2. Focal Length Observation

We consider now one example of the focal length observation process. Following the structure of the previous section, we show a MATLAB simulation which demonstrates the good convergence properties of (11)–(20). The simulation data are

$$x(t_0) = [10 \quad -10 \quad 128 \quad 1/128]^{\mathrm{T}}$$

$$\hat{x}(t_0) = [10 \quad -10 \quad 0 \quad 0]^{\mathrm{T}}$$

$$u_4(t) = 0.5 \cos \pi/2t$$

$$u_5(t) = 0.3 \sin \pi t$$

$$k_1 = 50$$

$$k_2 = 50$$

$$k_3 = 5000$$

$$k_4 = 1$$

In Figure 5(a)–(c), the evolution of the error vector $e(t)$ is reported, from which we can check that convergence is practically reached after 4 s of motion. It is worth noting that the behavior of $e_3(t)$ appears a bit erratic, alternating between

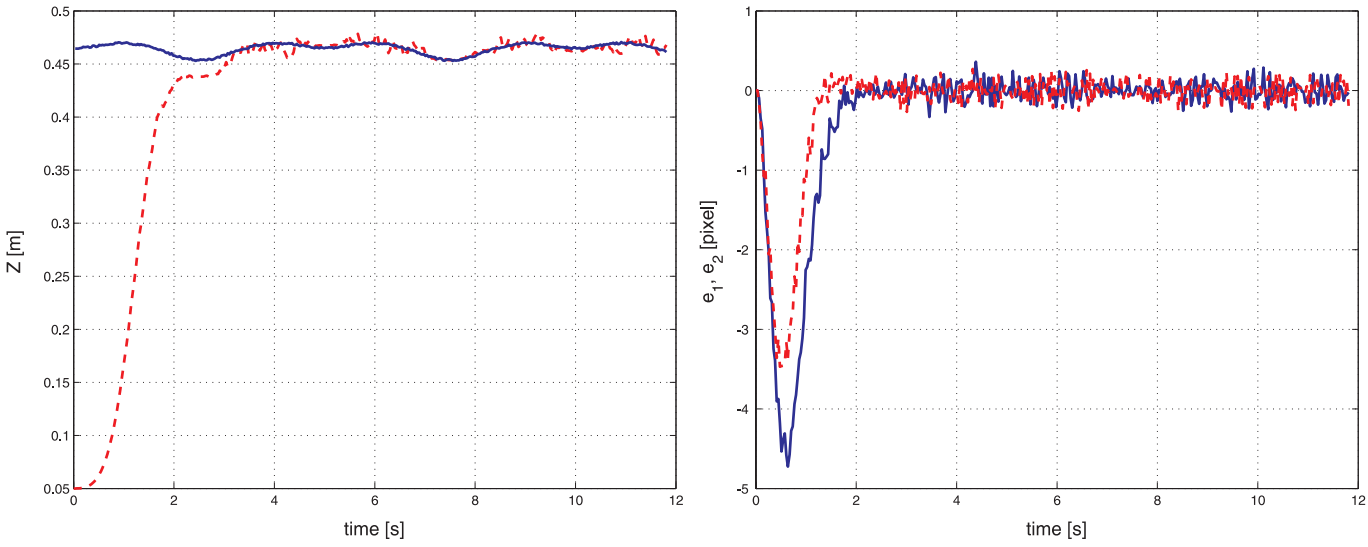


Fig. 4. Depth observer (Webots simulation): (a) true (solid blue line) and estimated (dashed red line) Z versus time; (b) behavior of e_1 (solid blue line) and e_2 (dashed red line) versus time.

fast decreases (around $t = 2$ s) and nearly flat transients (at $t = 1$ or 3 s). This is in close agreement with the persistency of excitation condition. Indeed, whenever matrix $\Omega(t)$ is far from singularity, the observation process can converge quickly, while when $\Omega(t)$ is close to ill-conditioning, convergence nearly stops. By defining $\sigma(A)$ as the smallest singular value of a matrix A, we report in Figure 5(d) the evolution of $\eta_1(t)$, $\eta_2(t)$ and $20 \cdot \sigma(\Omega(t))$ versus time (the factor 20 is introduced to obtain a comparable scale between $\sigma(\Omega(t))$ and $\eta_1(t)$, $\eta_2(t)$). As expected, the slow convergence phases correspond to a small value of $\sigma(\Omega(t))$ which goes to zero when either $\eta_1(t) \to 0$ or $\eta_2(t) \to 0$.

6. Plugging the Observer into the IBVS Loop

In these last few sections, the integration of depth/focal length observers and IBVS schemes is analyzed by means of several experiments run on a real robot carrying a camera. The aim is twofold: to effectively show how depth observation and IBVS can be integrated, and to point out the benefits of such integration in terms of stability and overall performance.

In order to better illustrate our approach, let us consider first the case of a generic robot committed with an IBVS task. Without loss of generality, let $q \in \mathbb{R}^n$ be the robot configuration vector, $u = \dot{q}$ the available inputs, and $f \in \mathbb{R}^s$ a vector of features describing the chosen visual task. The differential mapping between the commands u and $\dot{f}$ can be easily obtained by dividing the problem into two steps, and deriving the *image Jacobian*, at a given configuration, as the product of two matrices

$$\dot{f} = J_{\text{image}}(f, \chi, q)u = J_f(f, \chi)J_c(q)u, \qquad (25)$$

where:

- the $6 \times n$ matrix J_c gives the linear and angular velocity pair (v_C, ω_C) of the camera mounted on the end-effector (expressed in the camera frame), in response to the robot commands u;
- the $s \times 6$ matrix J_f is given by (5). In particular, if f is defined as the coordinates of k point features $[f_{u1} f_{v1} \ldots f_{uk} f_{vk}] \in \mathbb{R}^{2k}$, $s = 2k$, J_f is the stack of k point feature interaction matrices J_{f_i} as in (8), each one accounting for one point feature

$$\dot{f} = J_f(f, Z) \begin{bmatrix} v_C \\ \omega_C \end{bmatrix} = \begin{bmatrix} J_{f_1}(f_1, Z_1) \\ \vdots \\ J_{f_k}(f_k, Z_k) \end{bmatrix} \begin{bmatrix} v_C \\ \omega_C \end{bmatrix},$$

with $Z = [Z_1 \ldots Z_k]^{\mathrm{T}}$ the vector of depths associated with the k feature points.

The most simple and widely used IBVS control strategy is based on the inversion of (25) in terms of a desired feature velocity $\dot{f}$, that is,

$$u = H_{\text{image}}\dot{f}, \qquad (26)$$

where H_{image} is any generalized inverse of J_{image}. A common choice is $H_{\text{image}} = J^{\dagger}_{\text{image}}$, that is, the unique pseudo-inverse of J_{image} (Maciejewski and Klein 1989), and

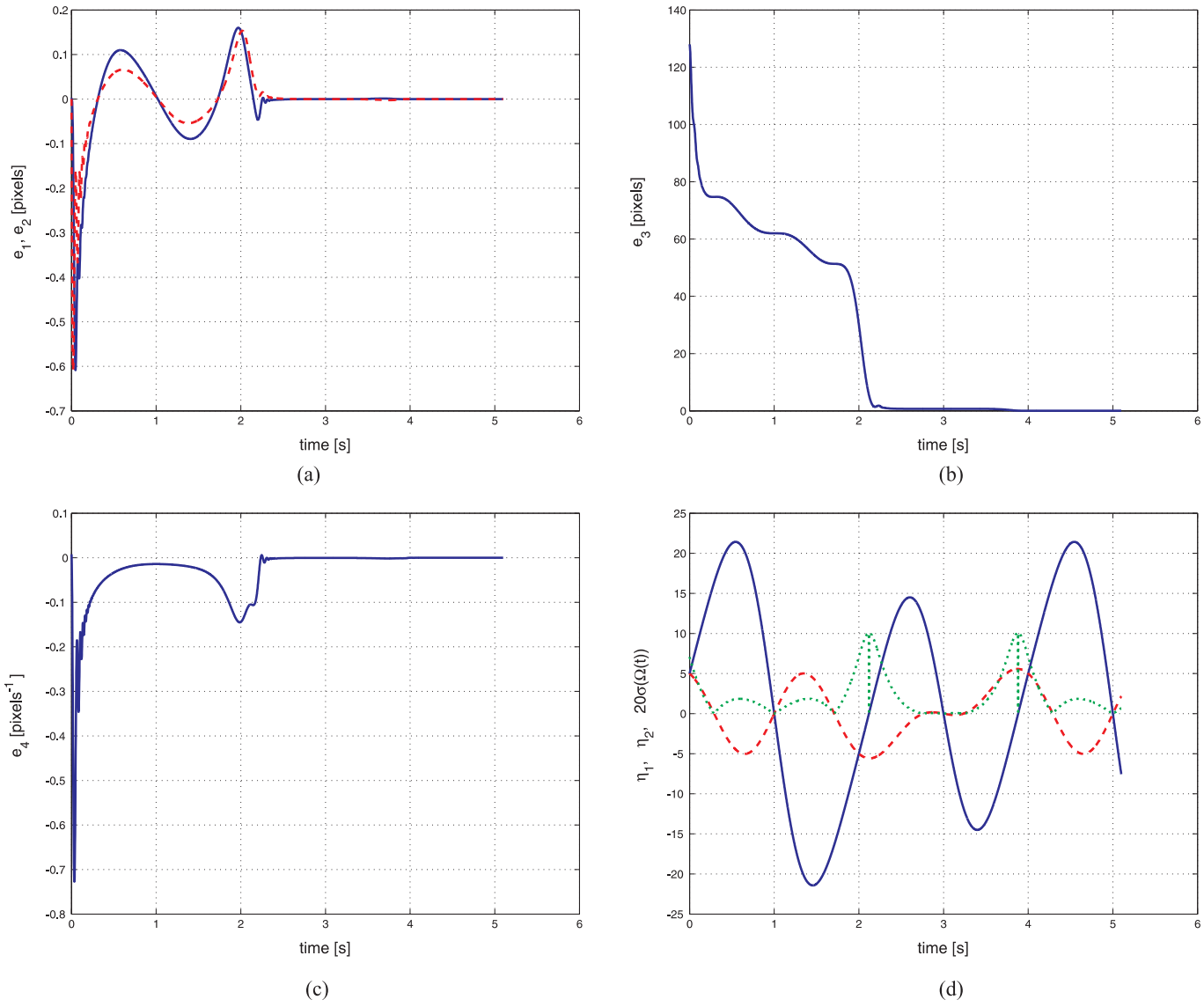


Fig. 5. Focal length observer (MATLAB simulation): (a) evolution of e_1 (solid blue line) and e_2 (dashed red line) versus time; (b) evolution of $e_3 = \lambda - \widehat{\lambda}$ versus time; (c) evolution of $e_4 = 1/\lambda - 1/\widehat{\lambda}$ versus time; (d) evolution of η_1 (solid blue line), η_2 (dashed red line), and $20 \cdot \sigma(\Omega(t))$ (dotted green line) versus time.

$$\dot{f} = K(f_d - f), \quad K > 0, \tag{27}$$

where f_d represents the constant desired value to be reached by task f. Feedback (26)–(27) guarantees a linear decoupled exponentially stable closed-loop behavior for the task error $e = f_d - f$, that is,

$$\dot{e} = -Ke. \tag{28}$$

Clearly, (26)–(27) represent just an ideal case, since in practice one does not have full knowledge of the matrix J_{image}, and some model/approximation/estimation $\widehat{J}_{\text{image}}$ must be used instead as discussed in the introduction. This yields the closed-loop behavior

$$\dot{e} = -J_{\text{image}}\widehat{J}^{\dagger}_{\text{image}}Ke \tag{29}$$

which does not possess the nice properties of (28) and can possibly become unstable if a very rough approximation is made. Uncertainties on $J_{\text{image}} = J_f J_c$ can have different sources, for example:

- poor knowledge of camera hand–eye calibration, that is, the fixed pose of the camera with respect to its mounting point on the robot body; this uncertainty obviously only affects matrix J_c;
- poor knowledge of matrix A, that is, the camera intrinsic parameters; this uncertainty only affects matrix J_f;

- poor knowledge of the 3D information present in vector χ (see (5)).

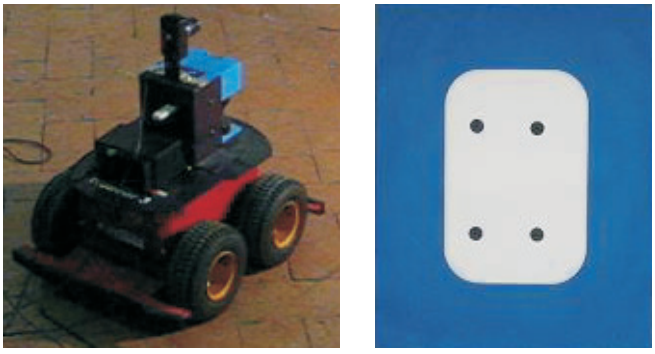
While typically both the camera hand–eye and intrinsic parameters are unknown but constant, and therefore they can be accurately estimated off-line once and for all (Strobl and Hirzinger 2006), vector $\chi(t)$ is made of unmeasurable time-varying quantities which need to be approximated in real implementations. As discussed in the introduction, a common choice is to perform the servoing by replacing $\chi(t)$ with the constant value χ_d relative to the final robot pose. Indeed, χ_d can be provided during the learning stage when the desired image is stored, or estimated off-line by means of any (partial) pose estimation algorithm. However, such a solution does not represent a pure 2D visual servoing scheme, since χ_d must still be measured preliminarily and independently from the servoing task. Our proposal is to replace the current $\chi(t)$ with its estimate $\widehat{\chi}(t)$ obtained from the observer (11)–(14). This can be considered as a concrete step towards a pure 2D visual servoing scheme, since the on-line observation of $\chi(t)$ is based only on image measurements (the positions of the point features) and the known camera motion. Furthermore, the use of $\widehat{\chi}(t)$ instead of χ_d has at least two general advantages:

1. it allows a visual task to be executed successfully without any preliminary 3D information, not even the value of χ_d;

2. it enlarges the stability domain and the transient performance of (29), in comparison with a scheme that uses a constant χ_d, providing a better approximation of $\widehat{J}_{\text{image}}$ thanks to the observer convergence during robot motion.

The first feature becomes particularly relevant in the case of navigation/exploration tasks for mobile robots equipped with cameras, whenever a set of locations is specified only in terms of images acquired during motion without the possibility of storing at the same time the corresponding depth information.

Note that, while in the linear domain the *separation principle* (Friedland 1986) would guarantee global stability of the combined servoing/observer system, when considering nonlinear systems this property is lost in general and convergence can be proved only locally, for example, if the observer initial conditions are close enough to the true state values. Obtaining an analytical characterization of the actual stability region with respect to initial task/observer errors/states is a difficult problem due to the high non-linearities present in the system dynamics and is currently the object of ongoing research. Promisingly, the experiments reported in the following sections have shown a good tolerance of the combined servoing/observer system with respect to observer initial errors, camera noise, and calibration uncertainties.



(a) (b)

Fig. 6. (a) Robot and (b) target object.

7. An Experimental Case Study

The experiments considered in this section have been conducted on a unicycle-like robot equipped with a fixed camera mounted on its top (Figure 6(a)). This design can be seen as a particular case of the more general class of standard manipulator arms mounted on a non-holonomic mobile platform, thus realizing a non-holonomic mobile manipulator (NMM). Indeed, such a class of robots merges the intrinsic dexterity of the arm with the enhanced workspace capabilities of the mobile base, and it is therefore a suitable choice for environment exploration/interaction tasks. As for the target to be tracked, we chose a vertical planar object with four black dots placed at the vertexes of a rectangular shape, see Figure 6(b).

In order to obtain the analytic expression of matrix J_c, needed in (25) for the computation of J_{image}, we follow the general methodology developed by De Luca et al. (2006); De Luca et al. (2007a) for kinematic modeling and control of NMMs (paying special attention to IBVS tasks). Let $q = [x\ y\ \theta]^{\mathrm{T}} \in \mathbb{R}^3$ be the platform configuration vector, $u = [v_p\ \omega_p]^{\mathrm{T}} \in \mathbb{R}^2$ be the linear and angular platform velocities, vector $r = [r_x\ r_y\ r_z]^{\mathrm{T}}$ be the relative displacement between the unicycle reference point and O_C (the camera optical center), and ϕ be the angle between camera Z_C axis and platform main axis (Z_C lies on the horizontal plane, see Figure 7). With this notation, we can obtain the following 6×2 matrix J_c

$$J_c = \begin{bmatrix} \sin\phi & -r_x\cos\phi - r_y\sin\phi \\ 0 & 0 \\ \cos\phi & r_x\sin\phi - r_y\cos\phi \\ 0 & 0 \\ 0 & -1 \\ 0 & 0 \end{bmatrix}$$

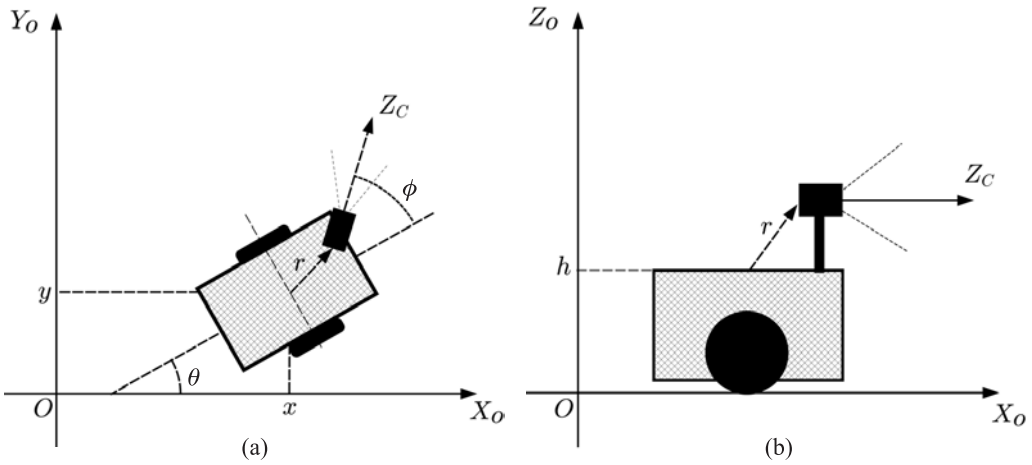


Fig. 7. Definition of frames, variables and constants: (a) top view; (b) side view.

mapping the robot inputs $[v_p\ \omega_p]^{\mathrm{T}}$ to the camera linear/angular velocity vector $[v_C\ \omega_C]^{\mathrm{T}}$ expressed in the camera frame. The geometric data of the robot are

$$r = [0.07\ 0.02\ 0.13]^{\mathrm{T}}\ [\mathrm{m}]$$

$$\phi = 0\ [\mathrm{rad}].$$

Note that J_c is completely independent of the platform configuration coordinates $(x,\ y,\ \theta)$. As explained by De Luca et al. (2007a), this property always holds when dealing with visual tasks for NMMs. In particular, it can be proven that J_c is a function of the manipulator arm variables only (not present in this case), thus making the computation of J_{image} independent of the mobile base absolute localization.

The following sections are arranged as follows. Section 7.1 presents experimental results on focal length observation, where the value of λ is estimated independently of the knowledge of Z. Then, Sections 7.2 and 7.3 illustrate the integration of depth observation in IBVS schemes based on point features and image moments, respectively[3].

7.1. Experiments on Focal Length Observation

As discussed in Section 4, it is possible to exploit the persistency of excitation lemma to observe the value of the focal length λ independently of Z. The purpose of this section is to provide an experimental validation of both the theoretical and simulation results presented in Sections 4 and 5.2.

With the robot chosen for our experiments, it is easy to check that a pure angular motion cannot be imposed on the camera. Indeed, for any angular velocity command ω_p of the platform, a camera linear velocity will always result due to the r_x and r_y components of the camera offset vector r. However, the camera is not very far from the platform center, that is, the values of r_x and r_y are small, and, as a consequence, the undesired camera linear velocity v_C can be considered negligible with respect to the imposed camera angular velocity ω_C. Therefore, we tested the algorithm by considering v_C as an additional external disturbance in addition to noise and modeling uncertainties. The feature point tracked during the experiment was the top-left black dot of the planar target in Figure 6(b).

The robot was commanded with the desired velocity profiles $v_d = 0$ [m/s], $\omega_d = 6\cos\pi/4t$ [degrees/s]. Figure 8(a) shows the commanded ω_d and actual ω_p velocity of the platform during a typical experiment. Note that ω_p was obtained by numerical differentiation of the wheels encoder readings. From the time behavior of $(v_C,\ \omega_C)$ reported in Figure 8(b), it can be readily seen that the main camera angular motion (ω_{Cy}, represented by the solid blue line) dominates all components of the undesired v_C.

The parameters of the focal length observation algorithm were set to

$$\hat{x}(t_0) = [17.01\ 32.99\ 0\ 0]^{\mathrm{T}}$$

$$k_1 = 30$$

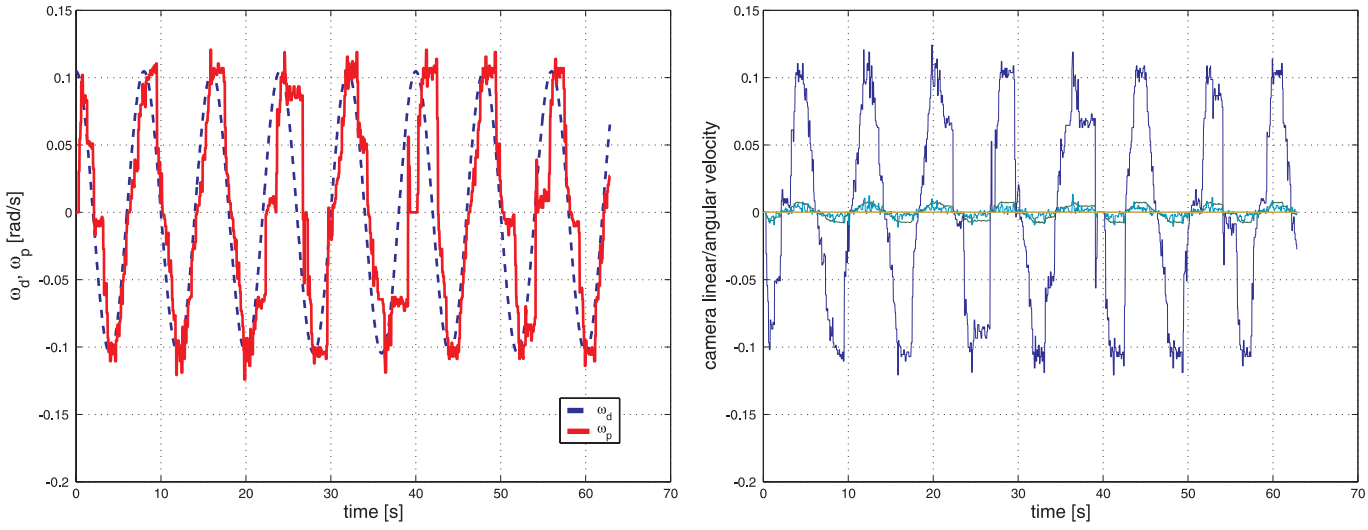
$$k_2 = 30$$

$$k_3 = 1500$$

$$k_4 = 0.01.$$

In order to have a reference value for comparing our observation results, we performed a standard off-line camera calibration by using the *MATLAB calibration toolbox* (Bouguet 2007). The focal length value obtained at this preliminary stage

3. The outcome of one of these experiments is also shown in Extension 1.



(a) (b)

Fig. 8. (a) Robot and (b) camera velocities during the focal length observation experiment: (a) ω_d (dashed blue line) and ω_p (solid red line) versus time; (b) ω_{Cy} (solid blue line) and the other components of the camera linear/angular velocity versus time.

was $\lambda_1 = 1096$ pixels. Figure 9 shows the result of the observation process. The estimated focal length $\widehat{x}_3(t)$ (solid blue line) reaches a steady state after about $t = 30$ s of motion (represented by a vertical line), despite noise and other disturbance sources. The average value of $\widehat{x}_3(t)$ after $t = 30$ s is $\lambda_2 = 1092.32$ pixels, thus very close to the value computed with the off-line calibration technique. In Figure 9, λ_1 and λ_2 are represented by the (almost coincident) horizontal green and red solid lines, respectively.

7.2. Experiments with Point Features

In this first set of four experiments, we tested the feedback control law (26)–(27), with $\chi(t)$ replaced by $\widehat{\chi}(t)$, for the case of regulation of 4 point features, see Figure 10(b)–(d). The same servoing task (same initial and final robot poses) was run with different initial values for the observer (11)–(14), with the aim of assessing the robustness and reliability of the integrated approach. In the following, we use superscript $j = 1 \dots g$ ($g = 4$) to denote the jth experiment, and subscript $i = 1 \dots k$ ($k = 4$) to denote the ith point feature.

For this visual task we have $f = [f_{u_1}\ f_{v_1} \dots f_{u_k}\ f_{v_k}]^{\mathrm{T}} \in \mathbb{R}^s$, $s = 2k = 8$, matrix $J_f \in \mathbb{R}^{2k \times 6}$ is made of the stack of k point feature Jacobian J_{f_i} as in (8), and $\chi = [Z_1 \dots Z_k]^{\mathrm{T}} \in \mathbb{R}^k$.

At the initial robot pose we have $Z_i(t_0) \simeq 3.9$ [m] for each point feature i, while at the desired final pose it is $Z_{d_i} \simeq 0.98$ m and $f_d = [46\ 379\ 143\ 374\ 139\ 201\ 38\ 203]^{\mathrm{T}}$. Note that, in this case,

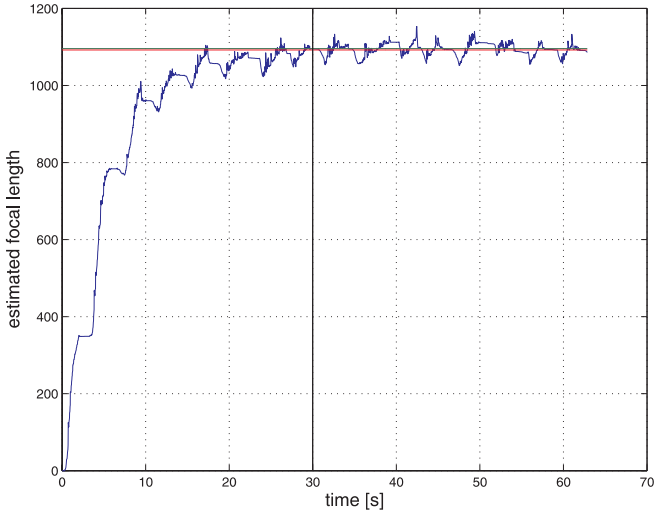


Fig. 9. Evolution of $\widehat{x}_3(t) = \widehat{\lambda}(t)$ (solid blue line) versus time. The estimate of λ reaches a steady-state after about 30 s of motion (vertical solid line).

$$\widehat{\chi}(t) = \left[\widehat{Z}_1 \dots \widehat{Z}_k\right]^{\mathrm{T}} = \left[1/\widehat{x}_{3_1} \dots 1/\widehat{x}_{3_k}\right]^{\mathrm{T}},$$

that is, it can be directly computed in terms of the observer output. In all four experiments, the first two observer states were always initialized as $\widehat{x}^j_{1_i}(t_0) = f^j_{u_i}(t_0)$ and $\widehat{x}^j_{2_i}(t_0) = f^j_{v_i}(t_0)$, $j = 1, \dots, 4$, that is, matching the measured initial feature positions. On the other hand, we considered four differ-

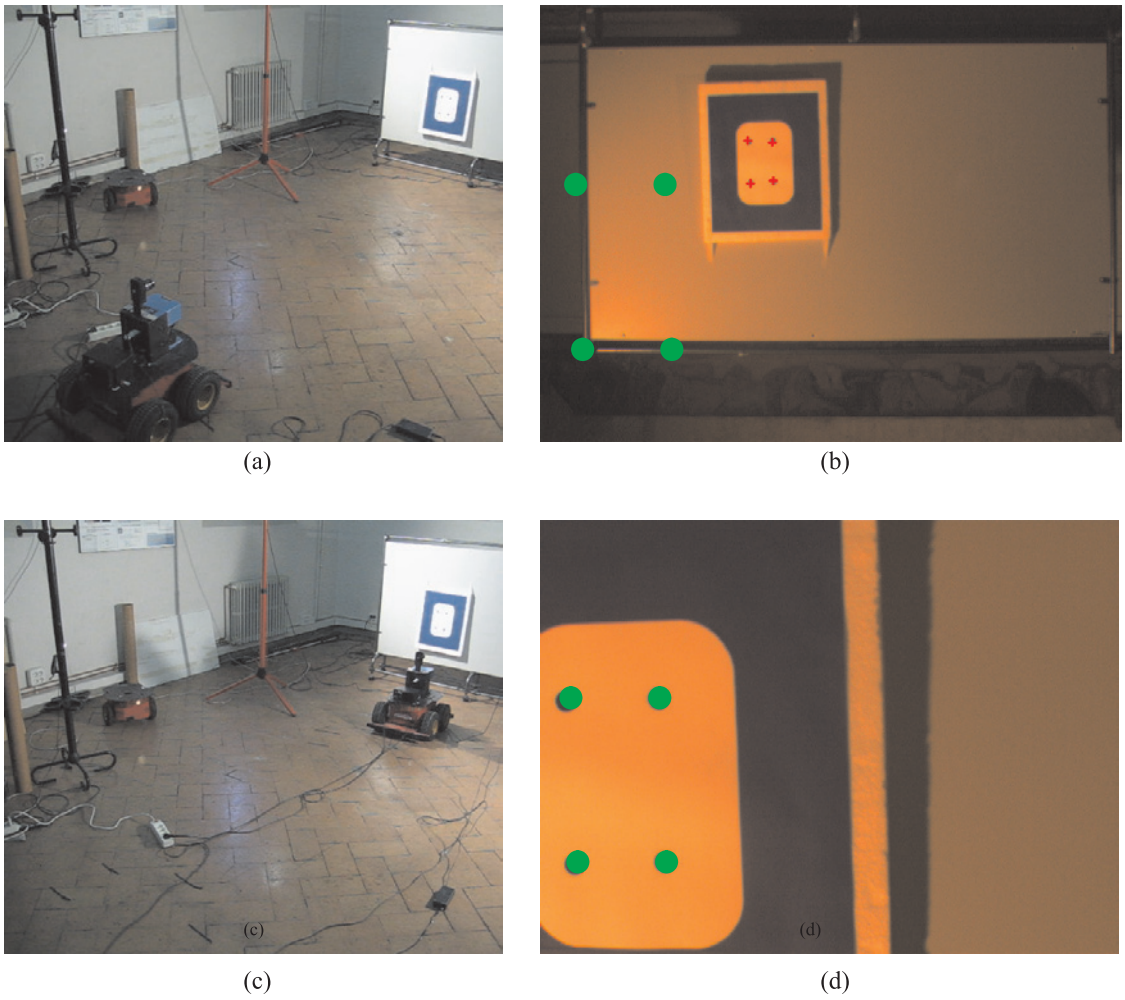


Fig. 10. Observer-based IBVS experiments: external and camera views. The green dots are the desired positions of the four point features: (a) initial pose (external view); (b) initial pose (camera view); (c) final pose (external view); (d) final pose (camera view).

ent values for the initial depth guesses, one for each experiment:

$$\begin{cases} \widehat{Z}_i^1(t_0) = 1/\widehat{x}_{3_i}^1(t_0) &= 1.6 \text{ [m]} \\ \widehat{Z}_i^2(t_0) = 1/\widehat{x}_{3_i}^2(t_0) &= 3 \text{ [m]} \\ \widehat{Z}_i^3(t_0) = 1/\widehat{x}_{3_i}^3(t_0) &= 5 \text{ [m]} \\ \widehat{Z}_i^4(t_0) = 1/\widehat{x}_{3_i}^4(t_0) &= 8 \text{ [m]}. \end{cases} \tag{30}$$

The feedback/observer gains were chosen as $K = 0.5I$, $k_1 = k_2 = 48$ and $k_3 = 44{,}000$.

Figure 11 shows the motion of the point features on the image plane in each of the four experiments. The visual task is successfully completed in all cases, that is, the feature points reach their desired positions without crossing the image plane boundaries (the black thick box in the plots). This is a direct consequence of the convergence properties of the observer, which tolerates wrong initial guesses for the depths and estimates their true time-varying values $Z_i(t)$ during robot motion.

We remark that, as already mentioned in the introduction, any visual servoing law in the form (26) is affected by the presence of local minima corresponding to unrealizable motions on the image plane (Chaumette 1998). When the error vector Ke belongs to the null space of H_{image}, the servoing law does not produce any motion command even if there is still a position error for the features. This is a well-known drawback of this class of visual servoing schemes, which becomes particularly relevant when considering more than three feature points with a camera with six DOFs. Indeed, the accurate knowledge of $\chi(t)$ obtained with our observer will provide no substan-

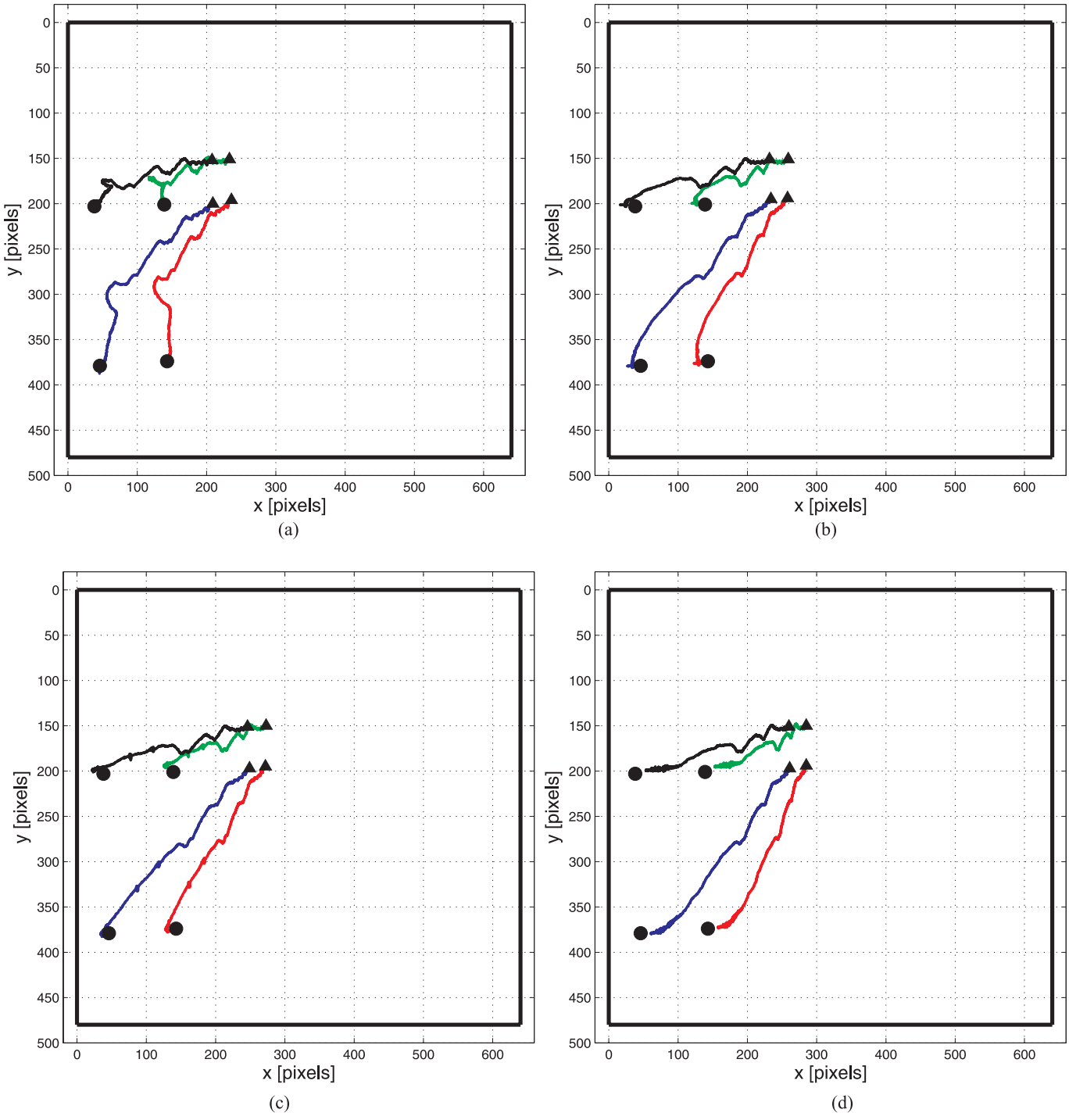


Fig. 11. Feature trajectories during the four servoing experiments: (a) $\widehat{Z}_i^1(t_0) = 1.6$ m; (b) $\widehat{Z}_i^2(t_0) = 3$ m; (c) $\widehat{Z}_i^3(t_0) = 5$ m; (d) $\widehat{Z}_i^4(t_0) = 8$ m. The black thick box represents the boundary of the image plane.

tial advantage in this context. In the performed experiments, however, we did not run into this situation and the considered servoing tasks could always be fulfilled.

Convergence to the true depth values can also be seen in Figure 12, which shows the time behavior of the estimated depths for each point feature i and each experiment j. De-

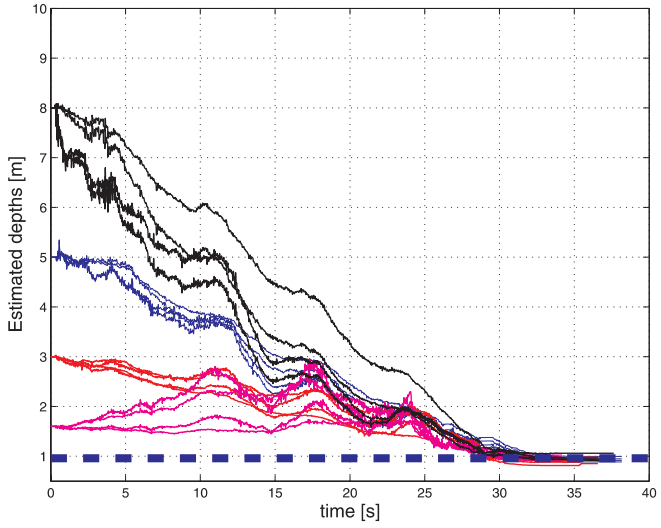


Fig. 12. Evolution of the estimated depths of the four point features during the four servoing experiments. The blue horizontal line represents the (common) depth $Z_{d_i} = 0.96$ m at the desired pose.

spite the different initial values used in the experiments, at the end of the motion every $\widehat{Z}_i^j(t)$ approaches the final value $Z_{d_i} \simeq 0.96$ m, that is, the true depth at the desired pose (represented in the plot by a blue dashed horizontal line).

It is also worth noting that the initial and final robot poses are such that, in each experiment, the linear motion of the camera occurs mainly along the Z axis, implying a continuously time-varying behavior for $Z_i(t)$. This can be verified in Figure 13, where the camera velocities v_C and ω_C of the fourth experiment are shown. The noisiness of v_C and ω_C is due to the numerical differentiation step needed to obtain the actual robot velocities from discrete sampling of wheel encoders. These results confirm the claims made in Sections 3 and 5.1 that the observer (11)–(14) is able to cope with large variations of the feature depth, showing in addition also good robustness against noise.

Finally, Figure 14 shows the feature motion on the image plane when considering a constant depth during the servoing, thus discarding the observer and relying just on prior knowledge about the scene. For comparison, the same previous four experiments were run with $\widehat{Z}_i^j(t) \equiv \widehat{Z}_i^j(t_0)$, that is, assuming that the initial depth guesses in (30) represent our best knowledge of the 3D scene. All of the experiments failed because the servoing scheme was not able to complete the task while keeping the feature points inside the boundaries of the image plane. This can be attributed to a too rough approximation used in $\widehat{J}_{\text{image}}$ due to the wrong depth information.

7.3. Experiments with Image Moments

As an additional case study, we tested the integrated approach by considering a more sophisticated feature set, that is, image moments instead of individual feature point coordinates. A thorough analysis and evaluation of the use of moments within the IBVS framework can be found in the work of Chaumette (2004) and Tahri and Chaumette (2005). For the reader's convenience, in the following we briefly summarize the main results.

Given a planar object with 3D plane equation $\alpha_1 X + \alpha_2 Y + \alpha_3 Z + \alpha_4 = 0$, in all non-degenerate cases[4] it is possible to express the inverse of the depth Z of each point on the plane in terms of its image coordinates

$$\frac{1}{Z} = A f_u + B f_v + C, \tag{31}$$

where $A = -\alpha_1/\alpha_4$, $B = -\alpha_2/\alpha_4$, $C = -\alpha_3/\alpha_4$ and $(f_u,\ f_v)$ represents the projection of the 3D point $[X\ Y\ Z]^{\mathrm{T}}$ (see Section 2.2). Note that parameters $(A,\ B,\ C)$ can be interpreted as the plane unit normal vector scaled by the plane distance from the camera frame.

Assume that a collection of w separate points are extracted and tracked from the target planar object. Analogously to the continuous case, it is possible to define the $(i,\ j)$th discrete moment m_{ij} as

$$m_{ij} = \sum_{l=1}^{w} f_{u_l}^i f_{v_l}^j,$$

and the $(i,\ j)$th discrete centered moment μ_{ij} as

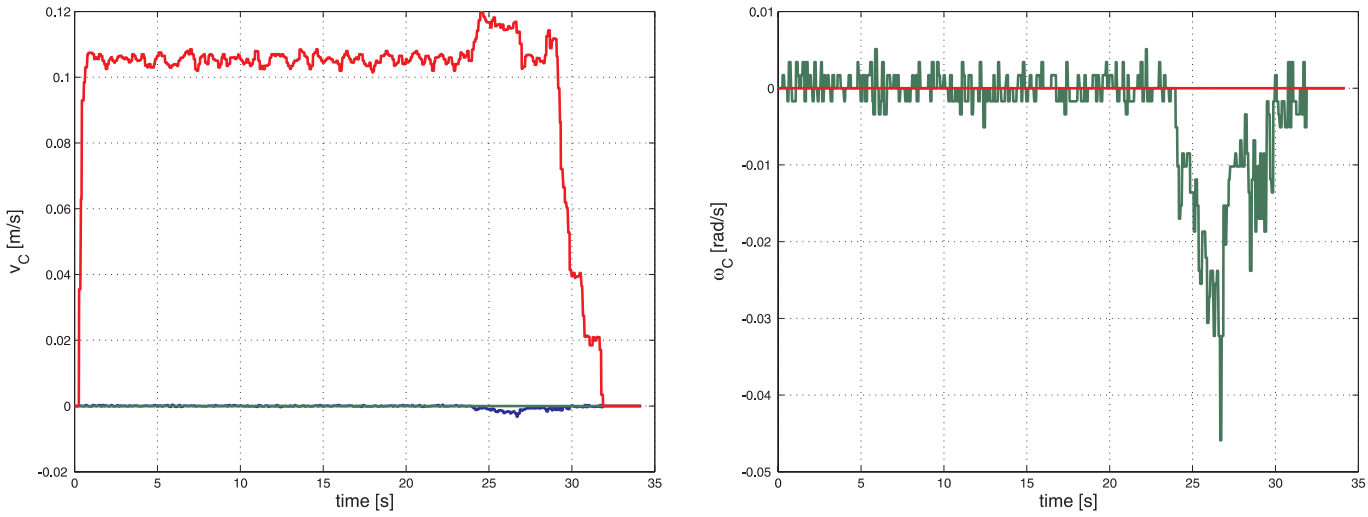
$$\mu_{ij} = \sum_{l=1}^{w} (f_{u_l} - x_g)^i (f_{v_l} - y_g)^j,$$

where $x_g = m_{10}/w$ and $y_g = m_{01}/w$ are the barycenter coordinates. Following the derivation of Tahri and Chaumette (2005), an analytical expression for the interaction matrices associated to m_{ij} and μ_{ij} can be obtained in terms of image moments and plane parameters $(A,\ B,\ C)$. In terms of our notation, given a generic feature vector $f \in \mathbb{R}^k$ made of k image moments, either m_{ij} or μ_{ij}, the interaction matrix will have the expression

$$\dot{f} = J_f(m_{kl},\ \mu_{kl},\ \chi)u,$$

where m_{kl} and μ_{kl} stand for generic $(k,\ l)$th moments of order up to $i + j + 1$ and $\chi = [A,\ B,\ C]^{\mathrm{T}}$. Note that m_{kl} and μ_{kl} can be directly measured on the image plane, while 3D information in χ reduces always to three parameters independently of the number of points considered. In addition to depth, some additional scene structure information should be known, that is, the current orientation of the plane in the camera frame.

4. A degenerate case occurs whenever the camera optical center belongs to the target plane.



(a) (b)

Fig. 13. (a) Linear (v_C) and (b) angular (ω_C) velocity of the camera during the fourth experiment. The dominant components are v_{C_z} and ω_{C_y}, respectively.

In our experiments, we considered a set of discrete moments defined by the four points present in the target object of Figure 6(b). In order to control the robot motion, we chose a feature set made of three moments, namely

$$f = \left[x_g \; y_g \; 1/\sqrt{a}\right]^{\mathrm{T}}, \tag{32}$$

where $a = \mu_{20} + \mu_{02}$ is a measure of the area enclosed by the selected points. Similarly to the case of point features, an approximation of $\chi(t)$ is required for the actual computation of the interaction matrix associated to the chosen image moments. As before, a standard choice is to set $\chi(t) \equiv \chi_d$. However, one can again exploit observer (11)–(14) to obtain an *indirect* estimate $\widehat{\chi}(t)$ during the servoing task.

Rearranging (31) and considering the k selected points, we obtain the linear system

$$\begin{bmatrix} f_{u_1} & f_{v_1} & 1 \\ & \vdots & \\ f_{u_k} & f_{v_k} & 1 \end{bmatrix} \begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} \frac{1}{Z_1} \\ \vdots \\ \frac{1}{Z_k} \end{bmatrix}$$

which can be easily solved as

$$\begin{bmatrix} A \\ B \\ C \end{bmatrix} = \begin{bmatrix} f_{u_1} & f_{v_1} & 1 \\ & \vdots & \\ f_{u_k} & f_{v_k} & 1 \end{bmatrix}^{\dagger} \begin{bmatrix} \frac{1}{Z_1} \\ \vdots \\ \frac{1}{Z_k} \end{bmatrix}$$

$$= \Lambda^{\dagger} \begin{bmatrix} \frac{1}{Z_1} \\ \vdots \\ \frac{1}{Z_k} \end{bmatrix}, \tag{33}$$

that is, by computing the "best" plane passing through the given points. Since matrix Λ is made of quantities directly measured on the image plane, an estimate of (A, B, C) can be obtained by replacing the true depths Z_i in (33) with the estimated $\widehat{Z}_i$ obtained from the observer. Therefore, in the following we use

$$\widehat{\chi}(t) = \begin{bmatrix} \widehat{A} \\ \widehat{B} \\ \widehat{C} \end{bmatrix} = \Lambda^{\dagger} \begin{bmatrix} \frac{1}{\widehat{Z}_1} \\ \vdots \\ \frac{1}{\widehat{Z}_k} \end{bmatrix}. \tag{34}$$

Following the same testing structure of the previous section, we again applied the feedback control law (26)–(27) based on $\widehat{\chi}(t)$ in $g = 5$ different experiments, keeping approximately the same initial and final robot poses as before and varying the initial observer state. Figure 15 shows the initial and final robot poses from external and camera views. Note that the central dot in Figure 15(b) and Figure 15(d) does not represent a physical point but the computed barycenter of the four point features. At the initial robot pose we have $Z_i(t_0) \simeq 4.1$ m

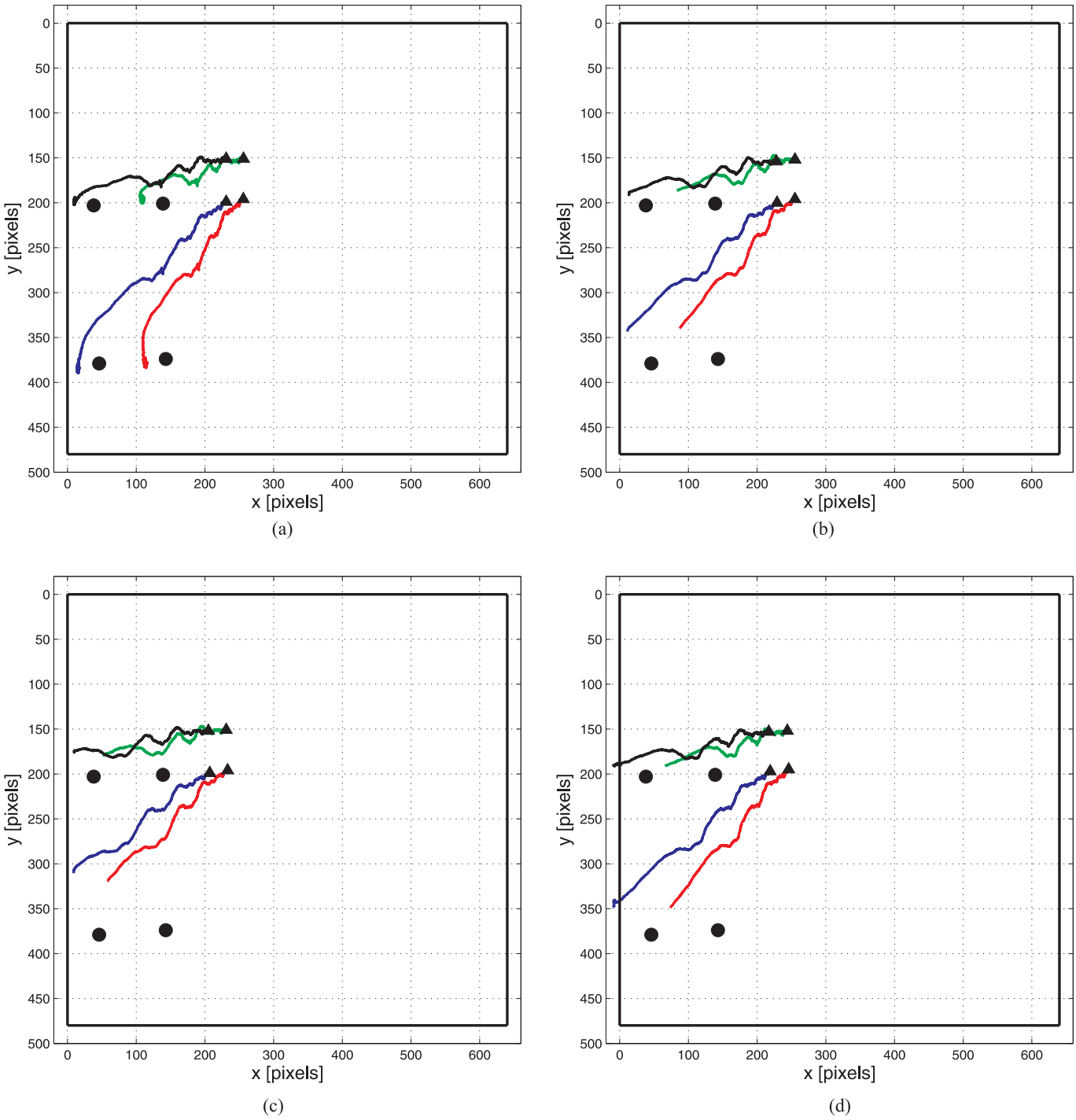


Fig. 14. Feature trajectories during the four servoing experiments *without* using the observer: (a) $\widehat{Z}_i^1(t) \equiv 1.6$ m; (b) $\widehat{Z}_i^2(t) \equiv 3$ m; (c) $\widehat{Z}_i^3(t) \equiv 5$ m; (d) $\widehat{Z}_i^4(t) \equiv 8$ m. The black thick box represents the boundary of the image plane.

for each point feature i, while at the desired final pose it is $Z_{d_i} \simeq 0.9$ m, $x_{g_d} = 422$ pixels, $y_{g_d} = 283.2$ pixels and $a_d = 18{,}571$ pixels^2.

In all experiments, the first two observer states were again initialized with the current feature position measured on the image plane. For the last observer state we chose

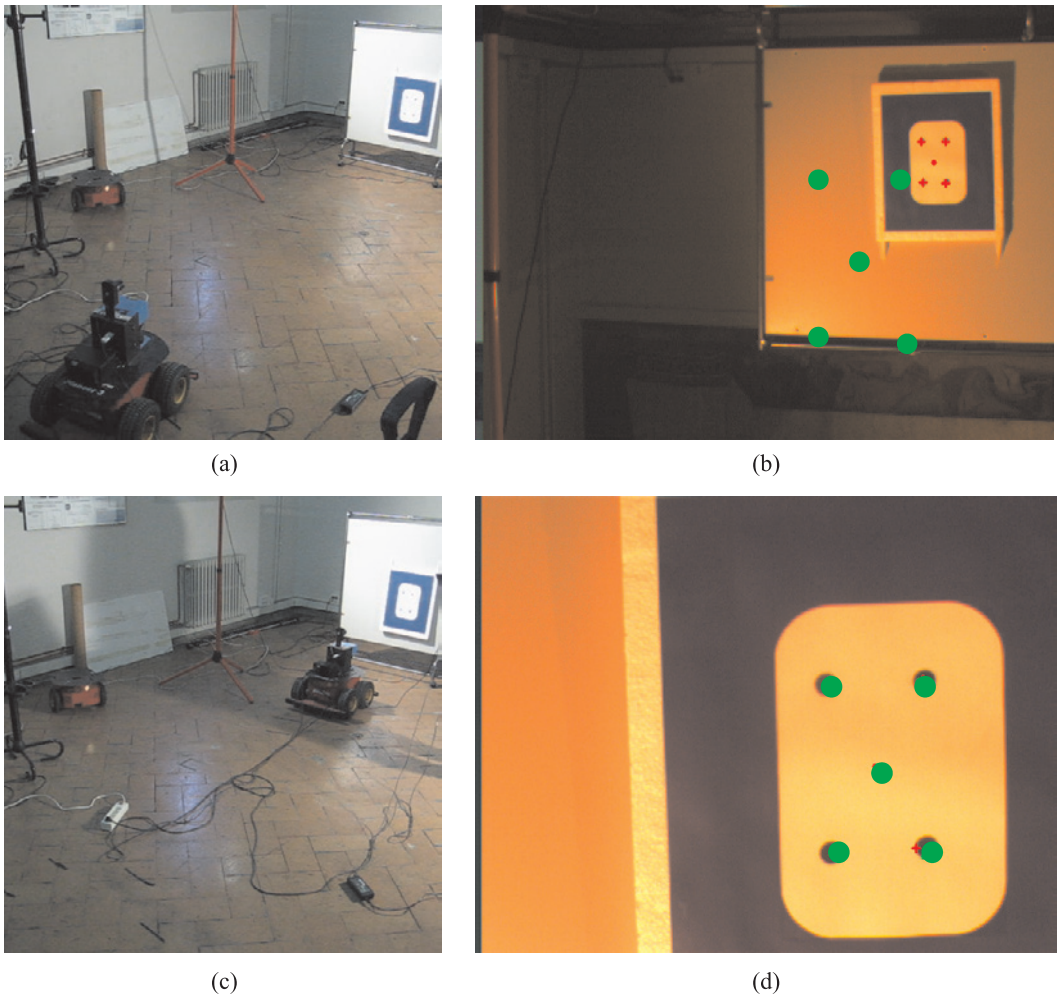


Fig. 15. Observer-based IBVS experiments: (a) initial pose (external view); (b) initial pose (camera view); (c) final pose (external view); (d) final pose (camera view). The green dots are the desired positions of the 4 point features and of their barycenter $(x_g,\ y_g)$.

$$\begin{cases} \widehat{Z}_i^1(t_0) = 1/x_{3_i}^1(t_0) &= 0.9\ [\text{m}] \\ \widehat{Z}_i^2(t_0) = 1/x_{3_i}^2(t_0) &= 3\ [\text{m}] \\ \widehat{Z}_i^3(t_0) = 1/x_{3_i}^3(t_0) &= 5\ [\text{m}] \\ \widehat{Z}_i^4(t_0) = 1/x_{3_i}^4(t_0) &= 8\ [\text{m}] \\ \widehat{Z}_i^5(t_0) = 1/x_{3_i}^5(t_0) &= 16\ [\text{m}]. \end{cases}$$

The gains in the feedback law and the observer were set as in the previous section.

Figure 16(a)–(e) shows the feature trajectories during the five experiments. White triangles/circles represent the initial/final positions of the $k = 4$ feature points, while a filled triangle/circle is used to denote the initial/final position of the barycenter $(x_g,\ y_g)$, that is, the feature over which we have "direct" control (see (32)). For the first experiment, Figure 16(f) also shows the behavior of $e_a(t) = 1/\sqrt{a_d} - 1/\sqrt{a(t)}$, that is, the task error relative to the third feature chosen for the servoing.

The integrated observer/controller approach is again able to fulfill the visual task with all of the different initial depth guesses and despite the indirect estimation of the plane parameters $(A,\ B,\ C)$. Convergence to the true depth values can be checked in Figure 17, where the blue dashed horizontal line represents the common depth value of the four feature points at the final robot pose.

As a final test, the same experiments were run using a constant value for $\widehat{\chi}(t)$, as computed from (34) by setting $\widehat{Z}_i^j(t) \equiv \widehat{Z}_i^j(t_0)$. The obtained results are shown in Figure 18. It is interesting to note that in the first two cases (Figure 18(a) and (b)) the servoing is still completed despite the approxima-

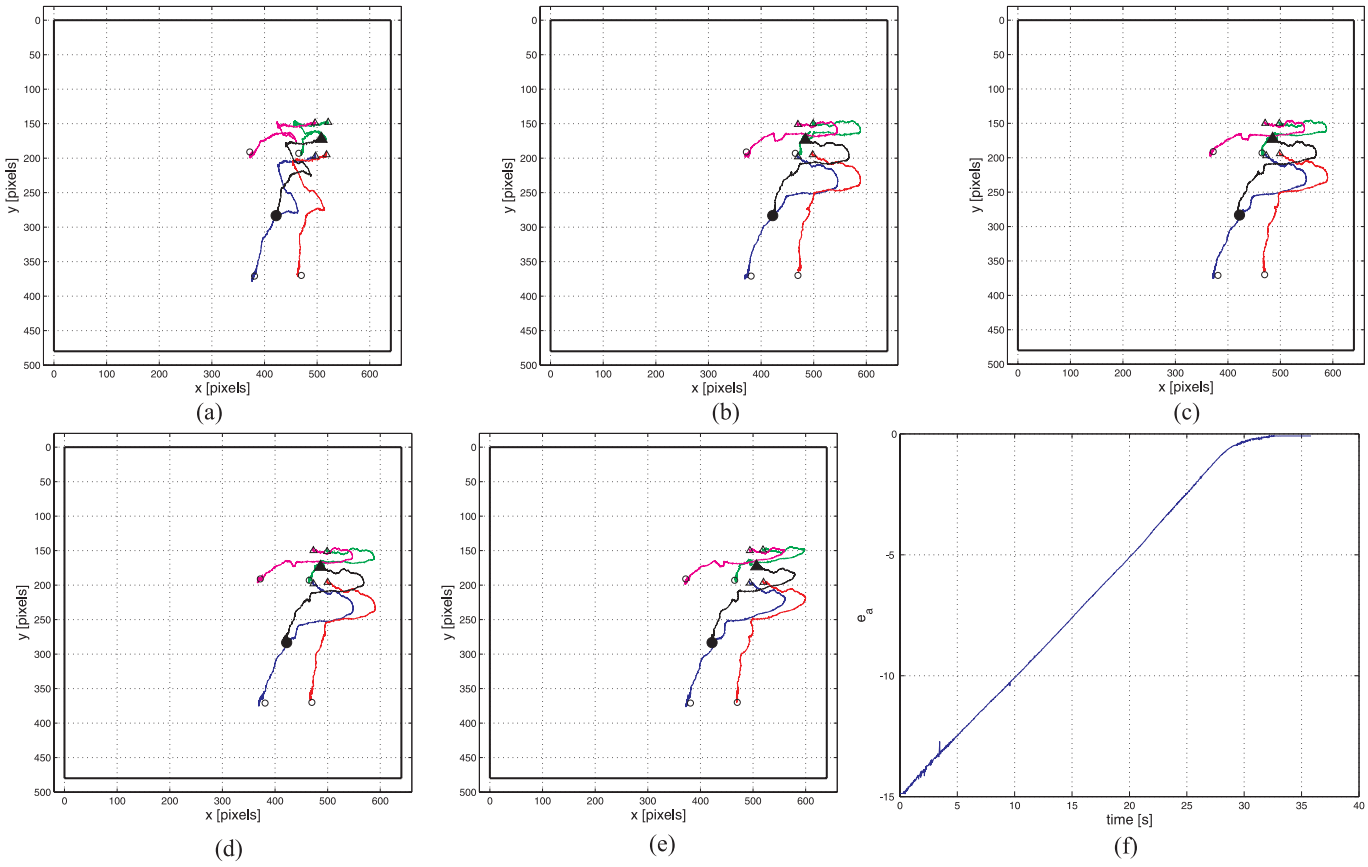


Fig. 16. Feature trajectories during the five servoing experiments: (a) $\widehat{Z}_i^1(t_0) = 0.9$ m; (b) $\widehat{Z}_i^2(t_0) = 3$ m; (c) $\widehat{Z}_i^3(t_0) = 5$ m; (d) $\widehat{Z}_i^4(t_0) = 8$ m; (e) $\widehat{Z}_i^5(t_0) = 16$ m. The black thick box represents the boundary of the image plane. (f) The time behavior of $e_a(t) = 1/\sqrt{a_d} - 1/\sqrt{a(t)}$ in the first experiment.

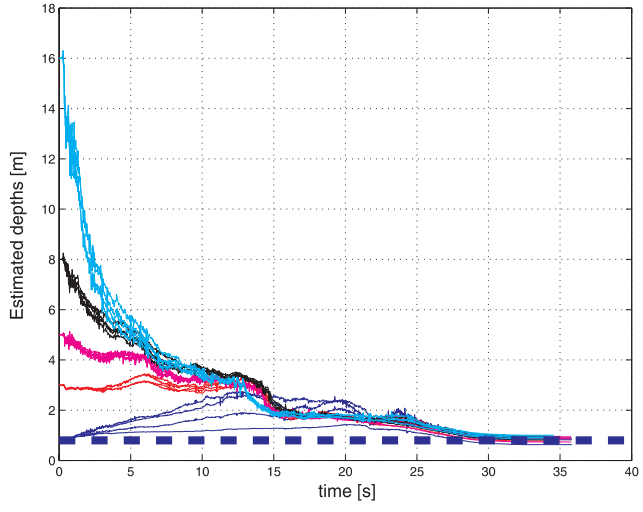


Fig. 17. Evolution of the estimation of the four point features' depths during the five servoing experiments. The blue horizontal line represents the (common) depth of the four point features at the desired pose.

tion adopted in the feedback law, while in the other cases the features exit the image plane boundaries during robot motion. Convergence in case (a), with constant depths $\widehat{Z}_i^1(t) \equiv Z_{d_i}$ set equal to their values at to the final robot pose, is not surprising. As mentioned in the introduction, many works (for example, those of Chaumette (2004) and Tahri and Chaumette (2005)) have shown that this choice allows for the fulfillment of the servoing task when the relative camera/target pose is inside the region of stability of the control scheme (as it was in this case). On the other hand, convergence in case (b) demonstrates some tolerance of the feedback (26)–(27) based on image moments to approximations in the depth/plane structure (in this case $\widehat{Z}_i^2(t) \equiv 3$ m are not so close to their true final depth values). In comparison, similar (and even milder) approximations used with point features in the previous section led to the failure of the servoing task (see Figure 14(a) and (b)). Nonetheless, in the remaining cases reported in Figure 18(c)–(f) convergence was not obtained due to the even larger depth approximations, in contrast to what happened in the corresponding experiments run by integrating the depth observer in the loop (Figure 16(c)–(f)).

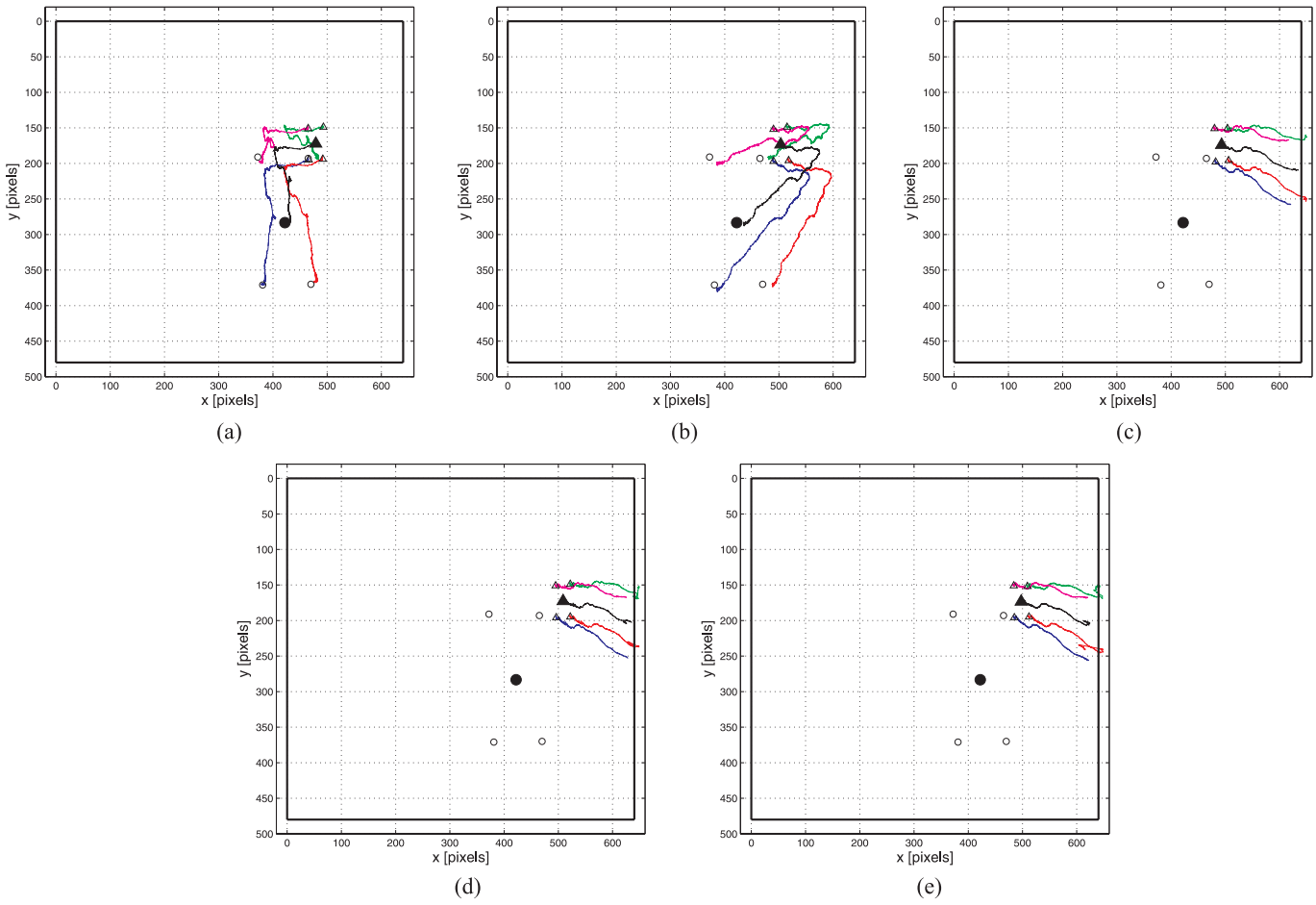


Fig. 18. Feature trajectories during the five servoing experiments *without* using the observer: (a) $\widehat{Z}_i^1(t) \equiv 0.9$ m; (b) $\widehat{Z}_i^2(t) \equiv 3$ m; (c) $\widehat{Z}_i^3(t) \equiv 5$ m; (d) $\widehat{Z}_i^4(t) \equiv 8$ m; (e) $\widehat{Z}_i^5(t) \equiv 16$ m. The black thick box represents the boundary of the image plane.

8. Conclusions and Future Work

In this work an integrated approach consisting of standard IBVS schemes and on-line observation of 3D structure was proposed. In particular, by exploiting the persistency of excitation lemma, we derived a non-linear observer able to recover the depth of stationary feature points during the robot/camera motion without approximation on the camera/point dynamics, special requirements on the allowed camera motion, or the need to estimate the feature image motion. An extension of this observer was tailored to the case of camera focal length identification. Simulation and experimental results allowed the assessment of good performance and reliability of the proposed approach. The integrated observer/controller scheme proved its effectiveness in dealing with standard IBVS tasks without any preliminary 3D knowledge of the scene, such as depth or other structure relative to the final pose.

Future developments will address a rigorous characterization of the stability of the closed-loop dynamics with the observer-based IBVS scheme, as well as the extension of the proposed observation scheme to generic planar shapes. Indeed, in this work we considered only the observation of depth for isolated feature points tracked during the camera motion. Whenever the extraction of point features is not easy or possible, as in the case of planar dense objects (for example, circles or ellipses), more general features such as region-based image moments can be exploited for observation and servoing. Recently, we proposed Robuffo Giordano et al. 2008 a theoretical framework able to deal with this kind of situations. We are currently seeking an experimental validation of the integrated observer/controller, similarly to what has been presented in this paper.

Concerning the proposed observation schemes, we are also investigating the possibility of recovering depth and focal length simultaneously, so as to obtain a common formulation for the two observation tasks discussed here separately. Finally, we would like to remove the assumption of a known camera motion which was used in this work. This could be

achieved, for instance, by first estimating the relative camera/target motion as done by Soatto et al. (1996), and then feeding this information to our observer algorithm for scene structure identification.

Acknowledgment

We wish to thank Professor Prattichizzo and his team from the University of Siena for their kind support which made it possible for us to run the experiments.

Appendix: Index to Multimedia Extensions

The multimedia extension page is found at http://www.ijrr.org

Table of Multimedia Extensions

Extension	Type	Description
1	Video	The video shows the external and camera views of the experiment described in Section 7.3, Figure 16(d). The depth observer is initialized with a value of 8 m, while the robot starts at about 4 m from the target.

References

Benhimane, S. and Malis, E. (2006). Homography-based 2D visual servoing. *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, pp. 2397–2402.

Bouguet, J.-Y. (2007). Camera Calibration Toolbox for MATLAB. http://www.vision.caltech.edu/bouguetj/calib_doc.

Chaumette, F. (1998). Potential problems of stability and convergence in image-based and position-based visual servoing. *The Confluence of Vision and Control*, Kriegman, D., Hager, G. and Morse, A. (eds), *Lecture Notes in Control and Information Science*, Vol. 237. Berlin, Springer, pp. 66–78.

Chaumette, F. (2004). Image moments: A general and useful set of features for visual servoing. *IEEE Transactions on Robotics and Automation*, **20**(4): 713–723.

Chaumette, F., Boukir, S., Bouthemy, P. and Juvin, D. (1996). Structure from controlled motion. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **18**(5): 492–504.

Chaumette, F. and Hutchinson, S. (2006a). Visual servo control. I. Basic approaches. *IEEE Robotics and Automation Magazine*, **13**(4): 82–90.

Chaumette, F. and Hutchinson, S. (2006b). Visual servo control. II. Advanced approaches. *IEEE Robotics and Automation Magazine*, **14**(1): 109–118.

Conticelli, F. and Allotta, B. (2001). Nonlinear controllability and stability analysis of adaptive image-based systems. *IEEE Transactions on Robotics and Automation*, **17**(2): 208–214.

Conticelli, F., Allotta, B. and Khosla, P. K. (1999). Image-based visual servoing of nonholonomic mobile robots. *Proceedings of the 38th IEEE Conference on Decision and Control*, pp. 3496–3501.

Corke, P. I. and Hutchinson, S. A. (2001). A new partitioned approach to image-based visual servo control. *IEEE Transactions on Robotics and Automation*, **17**(4): 507–515.

Cyberbotics (2007). Webots. http://www.cyberbotics.com.

De Luca, A., Oriolo, G. and Robuffo Giordano, P. (2007a). Image-based visual servoing schemes for nonholonomic mobile manipulators. *Robotica*, **25**(2): 131–145.

De Luca, A., Oriolo, G. and Robuffo Giordano, P. (2006). Kinematic modeling and redundancy resolution for nonholonomic mobile robots. *Proceedings of the 2006 IEEE International Conference on Robotics and Automation*, pp. 1867–1873.

De Luca, A., Oriolo, G., and Robuffo Giordano, P. (2007b). On-line estimation of feature depth for image-based visual servoing schemes. *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*, pp. 2823–2828.

Deguchi, K. (1998). Optimal motion control for image-based visual servoing by decoupling translation and rotation. *Proceedings of the 1998 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 705–711.

Espiau, B. (1993). Effect of camera calibration errors on visual servoing in robotics. *Proceedings of the 3rd International Symposium on Experimental Robotics*, pp. 182–192.

Espiau, B., Chaumette, F. and Rives, P. (1992). A new approach to visual servoing in robotics. *IEEE Transactions on Robotics and Automation*, **8**(3): 313–326.

Friedland, B. (1986). *Control System Design: An Introduction to State-Space Methods*. New York, McGraw-Hill.

Hutchinson, S., Hager, G. D. and Corke, P. I. (1996). A tutorial on visual servo control. *IEEE Transactions on Robotics and Automation*, **12**(5): 651–670.

Inaba, H., Yoshida, A., Abdursul, R. and Ghosh, B. K. (2000). Observability of perspective dynamical systems. *Proceedings of the 39th IEEE Conference on Decision and Control*, pp. 5157–5162.

Khalil, H. K. (1996). *Nonlinear Systems* (2nd edn). Englewood Cliffs, NJ, Prentice-Hall.

Ma, Y., Soatto, S., Košecká, J. and Sastry, S. S. (2004). *An Invitation to 3-D Vision*. New York, Springer.

Maciejewski, A. A. and Klein, C. A. (1989). The singular value decomposition: computation and applications to robotics. *International Journal of Robotics Research*, **8**(6): 63–79.

Malis, E. (2004). Visual servoing invariant to changes in camera-intrinsic parameters. *IEEE Transactions on Robotics and Automation*, **20**(1): 72–81.

Malis, E. and Chaumette, F. (2002). Theoretical improvements in the stability analysis of a new class of model-free visual servoing methods. *IEEE Transactions on Robotics and Automation*, **18**(2): 176–186.

Malis, E., Chaumette, F. and Boudet, S. (1999). 2-1/2-D visual servoing. *IEEE Transactions on Robotics and Automation*, **15**(2): 238–250.

Malis, E. and Rives, P. (2003). Robustness of image-based visual servoing with respect to depth distribution errors. *Proceedings of the 2003 IEEE International Conference on Robotics and Automation*, pp. 1056–1061.

Marino, R. and Tomei, P. (1995). *Nonlinear Control Design: Geometric, Adaptive and Robust*. London, Prentice-Hall.

Mathworks (2007). MATLAB. http://www.mathworks.com/products/matlab/.

Matthies, L., Szelinski, R. and Kanade, T. (1989). Kalman filter-based algorithms for estimating depth from image sequences. *International Journal of Computer Vision*, **3**: 209–236.

Morel, G., Liebezeit, T., Szewczyk, J., Boudet, S. and Pot, J. (2000). Explicit incorporation of 2D constraints in vision based control of robot manipulators. *Experimental Robotics VI*, Corke, P. and Trevelyan, J. (eds). Berlin, Springer.

Murray, R. M., Li, Z. and Sastry, S. S. (1994). *A Mathematical Introduction to Robotic Manipulation*. Boca Raton, FL, CRC Press.

Robuffo Giordano, P., De Luca, A. and Oriolo, G. (2008). 3D structure identification from image moments. *Proceedings of the 2008 IEEE International Conference on Robotics and Automation*, pp. 93–100.

Sepp, W., Fuchs, S. and Arbter, K. (2005). DLR Calibration Detection Toolbox (CalDe 0.99.0). http://www.dlr.de/rm/callab.

Smith, C. E. and Papanikolopoulos, N. P. (1994). Computation of shape through controlled active exploration. *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pp. 2516–2521.

Soatto, S., Frezza, R. and Perona, P. (1996). Motion estimation via dynamic vision. *IEEE Transactions on Automatic Control*, **41**(3): 393–413.

Strobl, K. H. and Hirzinger, G. (2006). Optimal hand-eye calibration. *Proceedings of the IEEE/RSJ International Conference on Robots and Intelligent Systems*, pp. 4647–4653.

Strobl, K. H. and Paredes, C. (2005). DLR Calibration Laboratory (CalLab 0.99.5). http://www.dlr.de/rm/callab.

Tahri, O. and Chaumette, F. (2005). Point-based and region-based image moments for visual servoing of planar objects. *IEEE Transactions on Robotics*, **21**(6): 1116–1127.

Taylor, C., Ostrowski, J. and Jung, S. (2000). Robust vision-based pose control. *Proceedings of the 2000 IEEE International Conference on Robotics and Automation*, pp. 2734–2740.

Wilson, W. J., Hulls, C. C. W. and Bell, G. S. (1996). Relative end-effector control using cartesian position based visual servoing. *IEEE Transactions on Robotics and Automation*, **12**(5): 684–696.