

# Direct visual tracking control of remote cellular robots

Ricardo Carelli<sup>a,\*</sup>, José Santos-Victor<sup>b</sup>, Flavio Roberti<sup>a</sup>, Santiago Tosetti<sup>a</sup>

<sup>a</sup> Instituto de Automática, Facultad de Ingeniería, Universidad Nacional de San Juan, Av. San Martín Oeste 1109, 5400 San Juan, Argentina

<sup>b</sup> Instituto Superior Técnico, Instituto de Sistemas e Robótica, Av. Rovisco Pais, 1049-001 Lisboa, Portugal

Received 13 December 2004; received in revised form 10 May 2006; accepted 12 May 2006

Available online 13 July 2006

## Abstract

This paper presents the design of a stable non-linear control system for the remote visual tracking of cellular robots. The robots are controlled through visual feedback based on the processing of the image captured by a fixed video camera observing the workspace. The control algorithm is based only on measurements on the image plane of the visual camera – direct visual control – thus avoiding the problems related to camera calibration. In addition, the camera plane may have any (unknown) orientation with respect to the robot workspace. The controller uses an on-line estimation of the image Jacobians. Considering the Jacobians' estimation errors, the control system is capable of tracking a reference point moving on the image plane – defining the reference trajectory – with an ultimately bounded error. An obstacle avoidance strategy is also developed in the same context, based on the visual impedance concept. Experimental results show the performance of the overall control system.

© 2006 Elsevier B.V. All rights reserved.

**Keywords:** Cellular robots; Visual tracking; Mobile robots; Lyapunov stability

## 1. Introduction

The use of robots for remote observation and task performance in a workspace can be of interest for many applications, e.g. surveillance, inspection, mine localization, space exploration, underwater tasks, etc. [10,11]. In this paper, we address the problem of using visual feedback for controlling simple and inexpensive cellular robots that can operate individually or cooperatively in a large remote area [14, 10]. Likewise in [10] it is assumed that a fixed camera observes the workspace, allowing a remote user to monitor the scene, to specify new tasks for the robot and to observe its operation. It is also assumed that the image plane may have a general (unknown) orientation with respect to the workspace. Each robot has a very simple design and it communicates via radio with the central processing unit including all the costly system components, such as the camera, frame grabbers and computing processor. In order to operate the system, the user specifies on the image the trajectory to be followed by each

robot, with its time evolution. The control of the robots is based on the processing of the images captured by the video camera. The general setup of the control system is illustrated in Fig. 1.

The main references for this work are the following ones. In [10], a linear controller to drive the robot to a final position has been designed and tested on an experimental cellular robot. In [6], an adaptive tracking controller for a mobile robot using a non-calibrated camera system is developed. The controller generates torque signals for the actuators and it requires, in addition to the estimation of kinematic variables in the image space, to know the actual robot orientation and angular velocities from the robot encoders. Besides, the approach relies on using a fixed camera whose image plane must be parallel to the working space, thus largely simplifying the viewing geometry. Also, in [8], to control a planar manipulator, the image plane is assumed to be parallel to the robot's workspace, and stability is proved with respect to the camera orientation about the optical axis. Instead, in the present approach, the camera may be installed in a general (unknown) position and orientation, thus offering a much greater flexibility. Indeed, if a camera is deployed at a remote site, it is difficult (if not impossible) to ensure that the image plane be parallel to the ground surface. Even if some nominal configuration could be

\* Corresponding author. Fax: +54 264 4213672.

E-mail addresses: [rcarelli@inaut.unsj.edu.ar](mailto:rcarelli@inaut.unsj.edu.ar) (R. Carelli), [jasv@isr.ist.utl.pt](mailto:jasv@isr.ist.utl.pt) (J. Santos-Victor), [froberti@inaut.unsj.edu.ar](mailto:froberti@inaut.unsj.edu.ar) (F. Roberti), [stosetti@inaut.unsj.edu.ar](mailto:stosetti@inaut.unsj.edu.ar) (S. Tosetti).

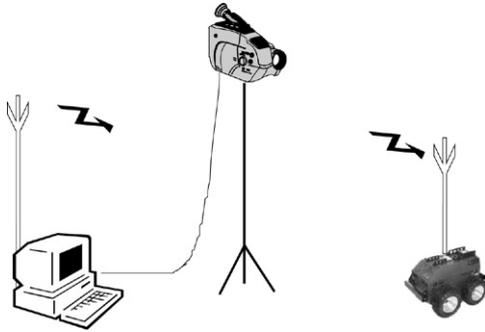


Fig. 1. Overall system setup.

established, it may always change over time, since the camera is subject to disturbances.

In summary, the contribution of the present paper is the design of a stable tracking control system based only on the visual feedback from a fixed camera, without requiring any knowledge about camera configuration. The controller uses the image Jacobians for computing the control signals. The Jacobians are estimated online and the associated estimation errors are taken into consideration, when analysing the stability proof.

The paper is organized as follows. Section 2 presents the system modelling including the image formation model. Section 3 describes the control objective and the controller design, which includes the stability proof. In Section 4 some experiments are discussed to show the practical feasibility and the performance of the proposed tracking control system. Finally, Section 5 draws some conclusions and establishes future directions of work.

## 2. System modelling

### 2.1. Robot kinematics

In this work a unicycle-like robot is considered with a differential architecture having two independent motors to drive the left and right wheels. The kinematics equations, which relate the linear and angular velocities of the robot to the angular velocity of each wheel can be expressed as:

$$\begin{bmatrix} v_w \\ \omega_w \end{bmatrix} = \begin{bmatrix} R/2 & R/2 \\ -R/D & R/D \end{bmatrix} \begin{bmatrix} \omega_L \\ \omega_R \end{bmatrix}$$

where  $\omega_R, \omega_L$  represent the right and left wheel angular velocities;  $v_w, \omega_w$  denote the linear and angular velocities of a robot coordinate frame attached to the central point of the robot. Parameter  $R$  stands for the wheel radius, and the distance between wheels is  $2D$ , as illustrated in Fig. 2.

The robot can be considered as a point  $c$  moving on the plane of an absolute coordinate system  $(x, y)$  as shown in Fig. 3. The usual set of kinematics equations, which describe the vehicle's Cartesian position  $(x, y)$  and heading  $\varphi$  are [1],

$$\begin{aligned} \dot{x}_w &= v_w \cos \varphi_w \\ \dot{y}_w &= v_w \sin \varphi_w \\ \dot{\varphi}_w &= \omega_w. \end{aligned} \quad (2.1)$$

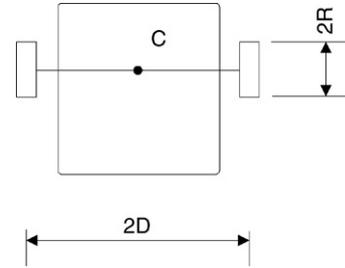


Fig. 2. Unicycle robot.

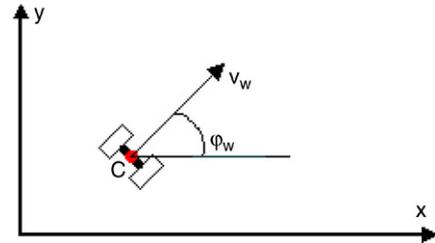


Fig. 3. Robot and reference system.

### 2.2. Robot dynamics

The linear and rotational dynamics of the robot are approximated by the following linear differential equations,

$$\begin{aligned} v_w(t) + T_v \dot{v}_w(t) &= K_v u_+(t) \\ \omega_w(t) + T_\omega \dot{\omega}_w(t) &= K_\omega u_-(t) \end{aligned} \quad (2.2)$$

$$u_+ = u_R + u_L; \quad u_- = u_R - u_L$$

where  $T_v, T_\omega$  are the linear and rotational time constants;  $K_v, K_\omega$  are the corresponding model gains;  $u_R, u_L$  are the input voltages applied to the right and left motors respectively, and  $u_+, u_-$  are the common and differential voltages.

### 2.3. Image formation model

A fixed vision camera monitors the robot evolution on the workplane, and the controller uses image measurements to drive the vehicle. This strategy is called direct vision control, because it does not involve coordinate transformations between the image- and workplanes. This strategy has the advantage of avoiding the problems associated with camera calibration and the calculation of coordinate transformations, because the control errors are directly computed in terms of image coordinates. Since the controller design is based on image measurements, the system's dynamics will be expressed in terms of the image plane's coordinates.

Points in the workspace and their image projections are related by a planar-projective transformation (homography) between these two planes [2]. Similarly, the geometry of image formation introduces a (non-linear) mapping,  $H_\varphi$ , between angles  $\varphi_w$ , measured on the workplane, and the corresponding angle  $\varphi$  projected on the image plane [10]:

$$\varphi = H_\varphi(\varphi_w). \quad (2.3)$$

Following the same rationale, the camera also introduces a mapping  $H_s$ , between a distance,  $s_w$  on the workplane and the corresponding distance,  $s$ , measured on the image plane:

$$s = H_s(s_w). \quad (2.4)$$

In general, both  $H_\varphi$  and  $H_s$  depend on the camera position and orientation (in the same way as homographies do) relative to the working plane, and on the camera's intrinsic parameters. Under normal viewing conditions, these functions are smooth differentiable maps, whereas singularities arise for extreme configurations, like when the image plane is perpendicular to the workplane. By taking time derivatives of (2.3) and (2.4), it yields:

$$\begin{aligned} \dot{\varphi} &= H'_\varphi(\varphi_w)\dot{\varphi}_w = J_\omega(\varphi_w)\dot{\varphi}_w \\ \dot{s} &= H'_s(s_w)\dot{s}_w = J_v(s_w)\dot{s}_w \end{aligned} \quad (2.5)$$

where  $J_v$ ,  $J_\omega$  are referred to as the image Jacobians, and will be used to obtain the overall system dynamics in terms of the image plane coordinates.

When using a calibrated camera, the image coordinates can always be transformed back into the workspace coordinate system. However, this transformation is not feasible in a non-calibrated setting. The objective of this work is the design of a stable control system, in spite of the existence of this unknown, non-linear mapping. To this aim, some properties of the viewing geometry will be used to provide the necessary technical constraints for the control system design, as explained in the following paragraphs. Without loss of generality, it is assumed that the camera is always positioned above the workplane, thus excluding the singular configuration of having the image plane perpendicular to the workplane. Under these normal viewing conditions the following properties hold:

- (a) By defining coherent coordinates both on the image plane and on the workplane, positive increments on angles (or distances) on the working space always correspond to positive increments of angles (and distances) on the image plane [10]. That is, both Jacobian functions  $J_v$ ,  $J_\omega$  are positive.
- (b) As a second observation, it is noted that the Jacobian functions that relate velocities on the image space to velocities on the working space are bounded,

$$\begin{aligned} J_\omega^m &< J_\omega(\varphi_w) < J_\omega^M \\ J_v^m &< J_v(s_w) < J_v^M \end{aligned}$$

and, due to the dynamics of the robot, are such that they change slowly.

#### 2.4. Dynamics on the image plane

The image formation model has been defined by the mappings  $H_\varphi$  and  $H_s$ , and the Jacobian functions  $J_v$ ,  $J_\omega$ . From (2.5), the following expressions can be obtained:

$$\begin{aligned} v &= J_v v_w \\ \omega &= J_\omega \omega_w. \end{aligned} \quad (2.6)$$

These functions allow expressing the dynamic model of (2.2) in terms of image measurements as,

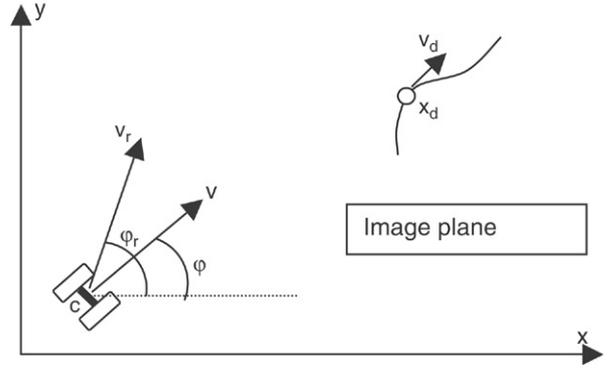


Fig. 4. Robot reference velocity and reference point on the image plane.

$$J_v^{-1}v + T_v J_v^{-1}\dot{v} + T_v \frac{d(J_v^{-1})}{dt}v = K_v u_+ \quad (2.7)$$

$$J_\omega^{-1}\omega + T_\omega J_\omega^{-1}\dot{\omega} + T_\omega \frac{d(J_\omega^{-1})}{dt}\omega = K_\omega u_-.$$

Considering Property (b), that regards slow variations of the Jacobians, the dynamics of (2.7) can be approximated by,

$$\begin{aligned} J_v^{-1}v + T_v J_v^{-1}\dot{v} &= K_v u_+ \\ J_\omega^{-1}\omega + T_\omega J_\omega^{-1}\dot{\omega} &= K_\omega u_- \end{aligned} \quad (2.8)$$

The dynamic model of the system, expressed in image coordinates, will be used in the next section to design the tracking controller.

### 3. Control system design

#### 3.1. Tracking control algorithms

This section presents the design of a tracking controller exclusively based on the visual feedback from a fixed camera, without requiring any knowledge about camera configuration. The commands, the measurements and the control objective are specified directly on the image obtained by the camera, i.e. direct visual control. Given a reference point moving on the image plane and, assuming that the attitude and velocity of the robot can be measured on the image plane, the control objective is to provide commands such that the robot follows the reference point on the image plane.

In a first step, a reference velocity vector for the robot is specified as a time function, calculated as follows:

$$\begin{aligned} \dot{\mathbf{x}}_{cr} &\triangleq \dot{\mathbf{x}}_d + \mathbf{K}(\tilde{\mathbf{x}})\tilde{\mathbf{x}} \\ \tilde{\mathbf{x}} &= \mathbf{x}_d - \mathbf{x}_c. \end{aligned} \quad (3.1)$$

In (3.1),  $\dot{\mathbf{x}}_{cr}$  is the reference velocity ( $\dot{\mathbf{x}}_{cr} = v_r \angle \varphi_r$ );  $\mathbf{x}_d$ ,  $\dot{\mathbf{x}}_d$  is the position and velocity of the reference point to be followed;  $\mathbf{x}_c$  is the actual position of the robot (Fig. 4). The matrix function  $\mathbf{K}(\tilde{\mathbf{x}})$  is introduced to avoid the saturation of the reference velocity, and it is selected so as to make that  $\tilde{\mathbf{x}}^T \mathbf{K}(\tilde{\mathbf{x}})$  be globally positive definite. For example, it can be selected as

$$\mathbf{K}(\tilde{\mathbf{x}}) = \text{diag} \left\{ \frac{k}{a + |\tilde{x}_i|} \right\}. \quad (3.2)$$

From (3.1), if  $\dot{\mathbf{x}}_c \equiv \dot{\mathbf{x}}_{cr}$ , then  $\dot{\tilde{\mathbf{x}}} + \mathbf{K}(\tilde{\mathbf{x}})\tilde{\mathbf{x}} = \mathbf{0}$ . By introducing the following Lyapunov candidate function and its time derivative,

$$V = \frac{1}{2} \tilde{\mathbf{x}}^T \tilde{\mathbf{x}}$$

$$\dot{V} = \tilde{\mathbf{x}}^T (-\mathbf{K}(\tilde{\mathbf{x}})\tilde{\mathbf{x}}) = -\tilde{\mathbf{x}}^T \mathbf{K}(\tilde{\mathbf{x}})\tilde{\mathbf{x}} < 0$$

it is clear that  $\tilde{\mathbf{x}}(t) \rightarrow 0$  asymptotically. Of course, this condition is verified for the ideal case that the robot follows exactly the reference velocity. This will not be so for a real controller, which will eventually reach the reference velocity asymptotically. The convergence of the control error to zero under this real condition will be analyzed at the end of this section.

The control commands of common and differential voltages to the robot actuators will be calculated to ensure the robot reaches the velocity reference asymptotically. These control laws are of the inverse dynamics type [4], based on the dynamics of the unicycle-type robot expressed in image coordinates (Eq. (2.8)). They rely on the online estimation of the image Jacobians relating the linear and heading velocities on the image plane and on the workplane. The proposed control law for heading control is:

$$u_- =: \frac{J_\omega^{-1}}{K_\omega} \left[ T_\omega (\dot{\varphi}_r + k_2 \dot{\tilde{\varphi}} + k_3 \tilde{\varphi}) + \omega \right] \quad k_2, k_3 > 0 \quad (3.3)$$

where  $\varphi_r$  is the reference orientation of the robot on the image plane,  $\tilde{\varphi} = \varphi_r - \varphi$  is the heading error of the robot,  $J_\omega$  is the angular Jacobian, and the constants are design parameters and robot model constants. By equating (3.3) with the second of (2.8), the closed loop equation is obtained,

$$\ddot{\tilde{\varphi}} + k_2 \dot{\tilde{\varphi}} + k_3 \tilde{\varphi} = 0 \quad (3.4)$$

which implies that  $\tilde{\varphi}(t) \rightarrow 0$  exponentially as  $t \rightarrow \infty$ .

By considering the linear velocity control, the proposed common voltage command will be:

$$u_+ =: \frac{J_v^{-1}}{K_v} \left[ T_v (\dot{v}_r + k_1 \tilde{v} \cos \tilde{\varphi}) + v \right] \quad k_1 > 0 \quad (3.5)$$

$$\tilde{v} = v_r - v$$

where  $v_r$  is the module of the reference velocity on the image plane;  $\tilde{v} = v_r - v$  is the error of the velocity modules;  $J_v$  is the linear Jacobian, and the constants are design parameters and robot model constants. The  $\cos \varphi$  factor allows us to attenuate the velocity module correction when the orientation of the robot is wrong. By equating (3.5) to the first of (2.8), the closed loop equation is obtained,

$$\dot{\tilde{v}} + k_1 \tilde{v} \cos \tilde{\varphi} = 0. \quad (3.6)$$

Considering the following positive definite function,

$$V = \frac{1}{2} \tilde{v}^2$$

the time derivative is given by

$$\dot{V} = \tilde{v} \dot{\tilde{v}} = -k_1 \tilde{v}^2 \cos \tilde{\varphi}. \quad (3.7)$$

According to (3.4),  $\cos \tilde{\varphi}$  takes positive values in a finite time, while  $\tilde{v}$  remains finite.

Therefore, (3.7) is negative definite in a finite time, which allows us to conclude that

$$\tilde{v}(t) \rightarrow 0 \quad \text{as } t \rightarrow \infty.$$

The Jacobians needed for (3.3) and (3.5) are estimated on-line using the  $\alpha - \beta$  filter [7] as described in Section 3.2.

It can be concluded now that the control error  $\tilde{\mathbf{x}}(t)$  tends to zero asymptotically. In the controller design it has been proven that  $\dot{\mathbf{x}}_{cr}(t) - \dot{\mathbf{x}}_c(t) = \boldsymbol{\rho}(t)$ , with  $\boldsymbol{\rho}(t) \rightarrow \mathbf{0}$ . Now (3.1) can be written as

$$\dot{\tilde{\mathbf{x}}} + \mathbf{K}(\tilde{\mathbf{x}})\tilde{\mathbf{x}} = \boldsymbol{\rho}(t). \quad (3.8)$$

Considering the following Lyapunov candidate and its time derivative,

$$V = \frac{1}{2} \tilde{\mathbf{x}}^T \tilde{\mathbf{x}} \quad (3.9)$$

$$\dot{V} = -\tilde{\mathbf{x}}^T \mathbf{K}(\tilde{\mathbf{x}})\tilde{\mathbf{x}} + \tilde{\mathbf{x}}^T \boldsymbol{\rho}.$$

A sufficient condition for the second of (3.9) to be negative definite is

$$\frac{k}{a + \|\tilde{\mathbf{x}}\|} \|\tilde{\mathbf{x}}\|^2 > \|\boldsymbol{\rho}\| \|\tilde{\mathbf{x}}\|$$

$$(k - \|\boldsymbol{\rho}\|) \|\tilde{\mathbf{x}}\| > a \|\boldsymbol{\rho}\|$$

$$\|\tilde{\mathbf{x}}\| > \frac{a \|\boldsymbol{\rho}\|}{k - \|\boldsymbol{\rho}\|}, \quad \text{if } k > \|\boldsymbol{\rho}\| \text{ or it is verified in a finite time.}$$

As  $\boldsymbol{\rho}(t) \rightarrow \mathbf{0}$ , it implies that  $\|\tilde{\mathbf{x}}(t)\| \rightarrow 0$  as  $t \rightarrow \infty$ .

### 3.2. On-line estimation of Jacobian functions

The control (3.3) and (3.5) proposed so far require the use of the Jacobian functions  $J_v, J_\omega$ . To overcome this constraint, these Jacobians are estimated on-line using the  $\alpha - \beta$  filter [7]. In order to write the filter equations, the following state and measurement variables are defined for the linear Jacobian:

$$\mathbf{X}(k) = [J_v \quad J_\omega]^T \quad Z(k) = v. \quad (3.10)$$

The state-space stochastic dynamic model is now defined as:

$$\mathbf{X}(k+1) = \boldsymbol{\Phi} \mathbf{X}(k) + \boldsymbol{\psi} \mu(k)$$

$$Z(k) = \mathbf{U} \mathbf{X}(k) + \lambda(k) \quad (3.11)$$

where  $k$  is the discrete time instant,  $\mu, \lambda$  are the state and measurement noises and  $\boldsymbol{\Phi}, \boldsymbol{\psi}$  are transition matrices,

$$\boldsymbol{\Phi} = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} \quad \mathbf{U} = [v_w \quad 0]$$

with  $T$  the sampling time. The filter equations are now written as:

$$\text{Predictor: } \hat{\mathbf{X}}(k/k-1) = \boldsymbol{\Phi} \hat{\mathbf{X}}(k-1/k-1) \quad (3.12)$$

$$\text{Filter: } \hat{\mathbf{X}}(k/k) = \hat{\mathbf{X}}(k/k-1) + \mathbf{K}[Z(k) - \mathbf{U} \hat{\mathbf{X}}(k/k-1)] \quad (3.13)$$

where  $\mathbf{K} = [\alpha \ \beta/T]^\top$  with  $\alpha, \beta$  selected by following [7]. Considering that the Jacobians change slowly – Property (b) – the following simple expressions for the filter can be written:

$$\text{Predictor: } \hat{J}_v(k/k-1) = \hat{J}_v(k-1/k-1) \quad (3.14)$$

$$\text{Filter: } \hat{J}_v(k/k) = \hat{J}_v(k/k-1) + \alpha[v - v_w \hat{J}_v(k/k-1)]. \quad (3.15)$$

Similarly, the angular Jacobian is estimated on-line using the following filter equations:

$$\text{Predictor: } \hat{J}_\omega(k/k-1) = \hat{J}_\omega(k-1/k-1) \quad (3.16)$$

$$\text{Filter: } \hat{J}_\omega(k/k) = \hat{J}_\omega(k/k-1) + \alpha[\omega - \omega_w \hat{J}_\omega(k/k-1)]. \quad (3.17)$$

It can be seen from (3.15) and (3.17) that the filter inputs are the linear and angular velocities on the image plane and on the workplane. The first ones are measured on the images captured by the camera, and the last ones are estimated using the dynamic model of the robot. This model has been identified by off-line experiments.

### 3.3. Effect of the errors in the Jacobians' estimation

The previous sections have shown how to derive a stable control law for the proposed control problem and the means for estimating the Jacobians online. The estimation of the Jacobian functions immediately raises the problem of analysing the effect of the estimation error on the control errors. The Jacobian function estimates are used to calculate the control actions, as done in (3.3) and (3.5). The differential control action of (3.3) is first considered with the estimated angular Jacobian,

$$u_- =: \frac{\hat{J}_\omega^{-1}}{K_\omega} \left[ T_\omega (\ddot{\varphi}_r + k_2 \dot{\varphi} + k_3 \tilde{\varphi}) + \omega \right]. \quad (3.18)$$

By equating (3.18) with the second of (2.8), the closed-loop expression is obtained,

$$\ddot{\varphi} + k_2 \dot{\varphi} + k_3 \tilde{\varphi} = \eta_1 + \eta_2 \quad (3.19)$$

where

$$\eta_1 = J_\omega \tilde{J}_\omega^{-1} \left[ \left( k_2 - \frac{1}{T_\omega} \right) \dot{\varphi} + k_3 \tilde{\varphi} \right] = J_\omega \tilde{J}_\omega^{-1} \mathbf{K} [\dot{\varphi} \ \tilde{\varphi}]^\top$$

$$\eta_2 = J_\omega \tilde{J}_\omega^{-1} \left( \ddot{\varphi}_r + \frac{1}{T_\omega} \dot{\varphi}_r \right) = J_\omega \tilde{J}_\omega^{-1} \Phi_r.$$

Eq. (3.19) can be expressed in state variables with state  $\mathbf{x}^\top = [\tilde{\varphi} \ \dot{\tilde{\varphi}}]$

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{N}_1(\mathbf{x}) + \mathbf{N}_2(t), \quad \mathbf{N}_1(\mathbf{0}) = \mathbf{0} \\ \begin{bmatrix} \dot{\tilde{\varphi}} \\ \ddot{\tilde{\varphi}} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ -k_3 & -k_2 \end{bmatrix} \begin{bmatrix} \tilde{\varphi} \\ \dot{\tilde{\varphi}} \end{bmatrix} + \begin{bmatrix} 0 \\ \eta_1 \end{bmatrix} + \begin{bmatrix} 0 \\ \eta_2 \end{bmatrix} \end{aligned} \quad (3.20)$$

with,  $\|\mathbf{N}_1(\mathbf{x})\| \leq \|J_\omega \tilde{J}_\omega^{-1}\| \|\mathbf{K}\| \|\mathbf{x}\| = \gamma_1 \|\mathbf{x}\|$  and  $\|\mathbf{N}_2(t)\| \leq \|J_\omega \tilde{J}_\omega^{-1}\| \|\Phi_r\| = \gamma_2$ .

By regarding the following Lyapunov function candidate [3],

$$V = \mathbf{x}^\top \mathbf{P}\mathbf{x} \quad (3.21)$$

its time derivative on the system's trajectories is given by,

$$\dot{V} = \left( \frac{\partial V}{\partial \mathbf{x}} \right) \dot{\mathbf{x}} = \left( \frac{\partial V}{\partial \mathbf{x}} \right) \mathbf{A}\mathbf{x} + \left( \frac{\partial V}{\partial \mathbf{x}} \right) \mathbf{N}_1(\mathbf{x}) + \left( \frac{\partial V}{\partial \mathbf{x}} \right) \mathbf{N}_2(t) \quad (3.22)$$

$$\dot{V} \leq -\alpha \|\mathbf{x}\|^2 + 2\lambda_{\max}(\mathbf{P}) \gamma_2 \|\mathbf{x}\| \quad (3.23)$$

where  $\alpha = \lambda_{\min}(\mathbf{Q}) - 2\lambda_{\max}(\mathbf{P})\gamma_1$  is positive by selecting  $\gamma_1 < \frac{\lambda_{\min}(\mathbf{Q})}{2\lambda_{\max}(\mathbf{P})}$ . It is concluded that the heading error is ultimately bounded [3] with a bound given by

$$\|\tilde{\varphi}\| < 2\lambda_{\max}(\mathbf{P}) \frac{\gamma_2}{\alpha} = \Phi. \quad (3.24)$$

The Jacobian estimation error is now considered, in relation to the common control action of (3.5),

$$u_+ =: \frac{\hat{J}_v^{-1}}{K_v} \left[ T_v (\dot{v}_r + k_1 \tilde{v} \cos \tilde{\varphi}) + v \right] \quad (3.25)$$

by equating (3.25) with the first of (2.8), the closed loop dynamics is obtained,

$$\dot{\tilde{v}} + k_1 \tilde{v} \cos \tilde{\varphi} = \eta_1(\tilde{v}, \varphi, t) + \eta_2(t) \quad (3.26)$$

with

$$\eta_1(\tilde{v}, \varphi, t) = J_v \tilde{J}_v^{-1} \left( k_1 \cos \tilde{\varphi} + \frac{1}{T_v} \right) \tilde{v}$$

$$\eta_2(t) = J_v \tilde{J}_v^{-1} \left( \dot{v}_r + \frac{v_r}{T_v} \right) = J_v \tilde{J}_v^{-1} \Omega_r.$$

By taking the positive definite function,

$$V = \frac{1}{2} \tilde{v}^2 \quad (3.27)$$

the time derivatives in the system's trajectories is given by

$$\dot{V} = \tilde{v} \dot{\tilde{v}} = \tilde{v} \left[ -k_1 \tilde{v} \cos \tilde{\varphi} + \eta_1(\tilde{v}, \varphi, t) + \eta_2(t) \right]. \quad (3.28)$$

Recalling that  $\tilde{\varphi}$  is an ultimately bounded variable,

$$\begin{aligned} \dot{V} &\leq -k_1 \cos \Phi \|\tilde{v}\|^2 + \|J_v \tilde{J}_v^{-1}\| \left( k_1 + \frac{1}{T_v} \right) \|\tilde{v}\|^2 \\ &\quad + \|J_v \tilde{J}_v^{-1}\| \|\Omega_r\| \|\tilde{v}\| = -\delta \|\tilde{v}\|^2 + \beta \|\tilde{v}\|. \end{aligned} \quad (3.29)$$

If the condition

$$k_1 \cos \Phi > \|J_v \tilde{J}_v^{-1}\| \left( k_1 + \frac{1}{T_v} \right)$$

is fulfilled, then  $\delta > 0$ . Under this condition,  $\dot{V} < 0$  for  $\delta \|\tilde{v}\|^2 > \beta \|\tilde{v}\|$ , which implies that  $\tilde{v}$  is ultimately bounded by

$$\|\tilde{v}\| \leq \frac{\beta}{\delta} = \Theta. \quad (3.30)$$

Since it has been proven that the velocity and heading errors are ultimately bounded by (3.24) and (3.30), the system's error equation (3.8) can now be expressed as

$$\dot{\tilde{\mathbf{x}}} + \mathbf{K}(\tilde{\mathbf{x}})\tilde{\mathbf{x}} = \boldsymbol{\rho}(t) \quad \|\boldsymbol{\rho}(t)\| \rightarrow R \quad (3.31)$$

with

$$R = \frac{\|\dot{\mathbf{x}}_{cr}\| \Phi}{\cos \varepsilon}, \quad \varepsilon = \arctan \frac{\theta}{\|\dot{\mathbf{x}}_{cr}\| \Phi}.$$

By considering the following Lyapunov candidate and its time derivative,

$$V = \frac{1}{2} \tilde{\mathbf{x}}^T \tilde{\mathbf{x}} \quad (3.32)$$

$$\dot{V} = -\tilde{\mathbf{x}}^T \mathbf{K}(\tilde{\mathbf{x}}) \tilde{\mathbf{x}} + \tilde{\mathbf{x}}^T \boldsymbol{\rho}(t) \leq -\tilde{\mathbf{x}}^T \mathbf{K}(\tilde{\mathbf{x}}) \tilde{\mathbf{x}} + R \|\tilde{\mathbf{x}}\|$$

the condition for  $\dot{V} < 0$  can be expressed as

$$\frac{k}{a + \|\tilde{\mathbf{x}}\|} \|\tilde{\mathbf{x}}\|^2 > R \|\tilde{\mathbf{x}}\| \quad (3.33)$$

which implies that  $\tilde{\mathbf{x}}$  is ultimately bounded, with bound  $\|\tilde{\mathbf{x}}\| \leq \frac{aR}{k-R}$ ,  $k > R$ .

### 3.4. Obstacle avoidance

It is important to endow the robot with the capability of avoiding any obstacles that may appear on the trajectory. Any strategy developed to this aim should be integrated with the proposed tracking controller described above, which operates with measurements on the image plane. The concept of visual impedance with fictitious forces [9,13,5,12] is used here to modify the desired velocity, thus deviating the desired robot trajectory to avoid the obstacle. The obstacle is detected by the vision system, and the fictitious forces are generated as a function of the relative posture of the robot and the obstacle on the image plane.

In robotics, the concept of impedance control aims at establishing the dynamic regulation between the motion and the interacting force of the robot with the environment. The linear impedance can be expressed as

$$\mathbf{f}(t) = \mathbf{Z}(p) \tilde{\mathbf{x}}(t)$$

with  $p = d/dt$  the time derivative operator,  $\mathbf{f}(t)$  is the interacting force of the robot with the environment;  $\tilde{\mathbf{x}}(t) = \mathbf{x}_d(t) - \mathbf{x}_c(t)$  represents the robot motion error in relation to the specified trajectory, and  $\mathbf{Z}(p) = \mathbf{I}p^2 + \mathbf{B}p + \mathbf{K}$  is the impedance function. In relation to the force vector  $\mathbf{f}(t)$ , physical forces are considered when trying to regulate the mechanical interaction of a robot by using information from force sensors. Fictitious forces are used instead, when a non-contact regulation is preferred, i.e., a virtual interaction captured by the image in a visually controlled system. The control objective is then defined as the regulation of the dynamic relationship between the fictitious forces and the position error on the image plane.

Regarding the obstacle avoidance strategy developed here for the visual tracking control of cellular robots, the following impedance relation is considered,

$$f = (Ip^2 + Bp + K)x_a = Zx_a \quad (3.34)$$

$$x_a = Z^{-1} f \cos \beta$$

where  $I$ ,  $B$  and  $K$  are positive constants;  $f$  is the fictitious force module;  $f_t$  is the tangential component of the fictitious force as

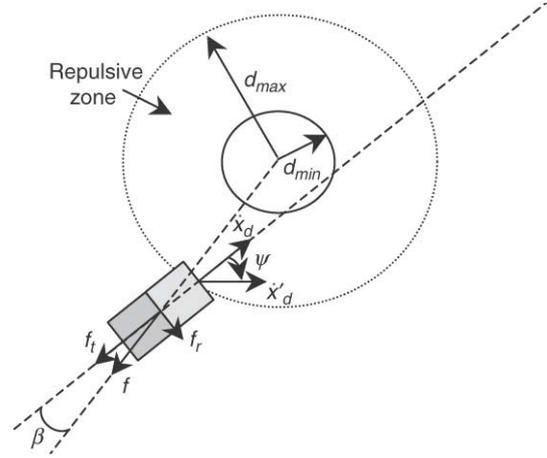


Fig. 5. Rotation of the velocity vector due to the fictitious force effect.

shown in Fig. 5, and  $x_a$  represents an error which will be used to deviate the robot from the desired trajectory. Constants  $I$ ,  $B$  and  $K$  represent the inertial, damping and spring effect of the impedance interaction.

The fictitious force module is calculated as,

$$f(t) = a - b \cdot (d(t) - d_{min})^n \quad (3.35)$$

where  $a$ ,  $b$  are positive constants such that

$$a - b \cdot (d_{max} - d_{min})^n = 0. \quad (3.36)$$

In the above equations,  $d(t)$  is the distance between the robot and the obstacle ( $d_{min} < d(t) < d_{max}$ );  $d_{max}$  defines a repulsive zone within which the obstacle avoidance strategy becomes active, see Fig. 5. Finally,  $d_{min}$  represents the minimum distance for non-contact of the robot with the obstacle. The choice of the value for the exponent  $n$  will depend on how fast it is required that the fictitious force increase while the robot approaches the obstacle.

Now, a modified desired velocity vector  $\dot{\mathbf{x}}'_d$  is calculated, by rotating an angle  $\psi$

$$\dot{\mathbf{x}}'_d = \begin{bmatrix} \cos \psi & -\sin \psi \\ \sin \psi & \cos \psi \end{bmatrix} \dot{\mathbf{x}}_d. \quad (3.37)$$

The rotation angle  $\psi$  is calculated as a function of the error signal  $x_a$ , and of the angle  $\beta$  of incidence of the robot in the repulsive zone, Fig. 5,

$$\psi = x_a \text{sign}(f_r) \quad (3.38)$$

$$f_r = f \sin(\beta).$$

## 4. Experimental results

The controller proposed in the previous section has been implemented and tested on experimental cellular robots developed at the Instituto de Automática, National University of San Juan, Argentina. The system consists of a robot controlled via a radio link from a PC. A host computer executes the tasks of determining the robot position and orientation on the image coordinates, and generates the proper control signals.

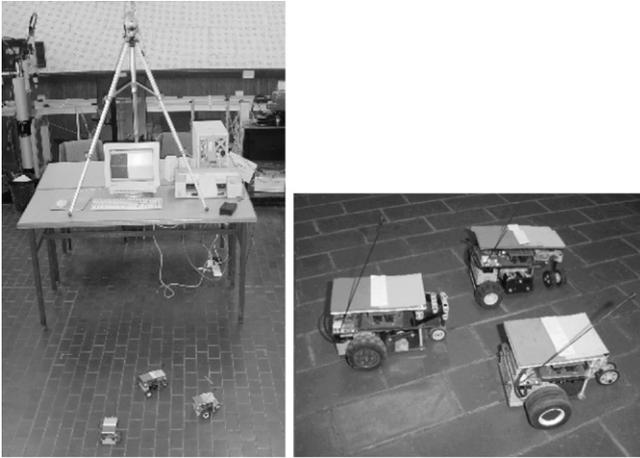


Fig. 6. Experimental setup and cellular robots.

The visual tracking system runs at 20 Hz, without using any special processing hardware. The reference trajectories and the saturation function in (3.1) have been selected such that the available sampling rate is adequate for the experiments. The communication is ensured via serial port, with a simple protocol to send the control commands. The robots are assembled from LEGO components and they contain a CPU-board consisting of a radio receiver, the decoding logic and PWM generation for the differential control of the robot. Each robot is identified by a four bit address and, consequently, up to 16 cellular vehicles can be controlled. There are no on-board sensors. The experimental setup and the cellular robots are shown in Fig. 6.

To localize the robot on the image plane, a tracking system was developed which estimates the robot position and heading direction over time from the video stream. The video camera uses a RGB representation, allowing colour detection for robot segmentation in the image plane. The robot position on the image plane is thus estimated by calculating the centre of mass on the binary image obtained from the colour segmentation. In a similar way, the robot heading direction is estimated using the angle of the vector passing through the centre of mass of two distinct colour bars located at the front and back of the robot. These variables and its derivatives required in the control laws of (3.3) and (3.5) are estimated using the  $\alpha - \beta$  filter [7].

In a first experiment, the reference point moves on the image plane describing a straight line with velocity defined by the relation  $v_x/v_y = 0.81$  with  $v_x = 13$  pix/s. The controllers' design constants are set to  $k_1 = 3$ ;  $k_2 = 4$ ;  $k_3 = 8$ ;  $k = 60$ ;  $a = 50$ . Fig. 7 shows the robot evolution on the image plane. Fig. 8 represents the evolution of the tracking error; Fig. 9, the evolution of the linear and angular velocities and, Fig. 10, the evolution of the estimated Jacobians.

In a second experiment, the target point moves on the image plane describing a circular trajectory, with a 75 pixels radius and angular velocity  $\omega = 0.18$  rad/s. The parameters of the controller are set to:  $k_1 = 2$ ;  $k_2 = 5$ ;  $k_3 = 8$ ;  $k = 70$ ;  $a = 50$ . Fig. 11 shows the robot evolution on the image plane. Fig. 12, represents the evolution of the tracking error; Fig. 13, the magnitude of the angular and lineal velocities, and Fig. 14,

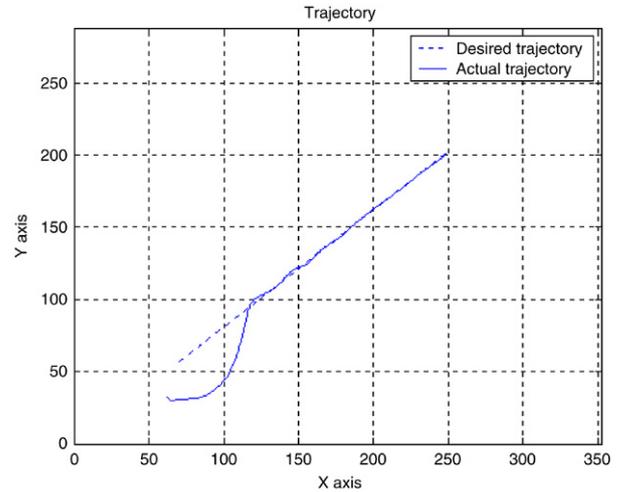


Fig. 7. Robot evolution on the image plane.

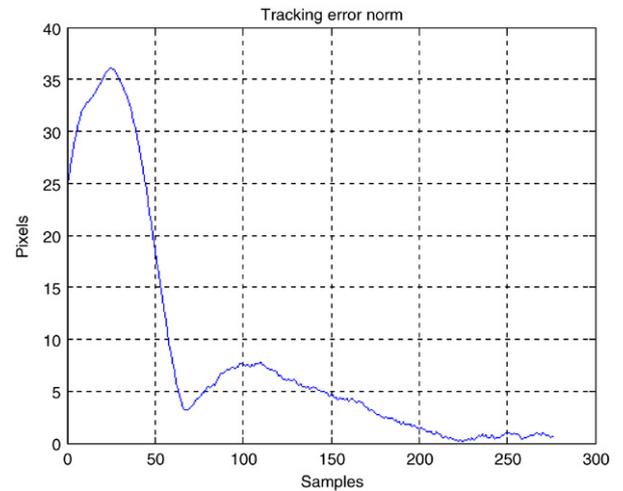


Fig. 8. Evolution of the tracking error.

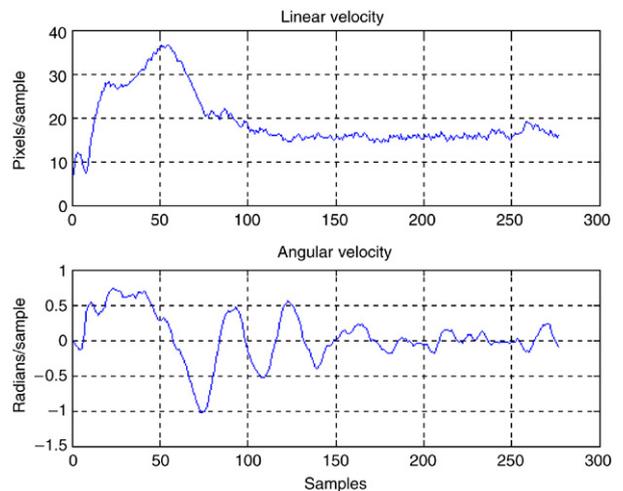


Fig. 9. Evolution of linear and angular velocities.

the behavior of the estimated linear and angular Jacobians for this experiment.

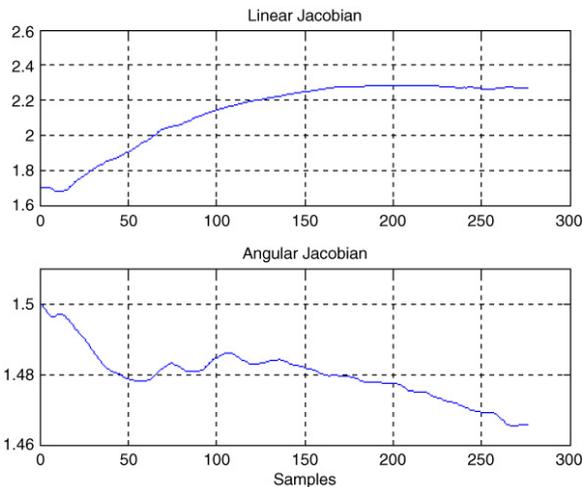


Fig. 10. Evolution of the estimated Jacobians.

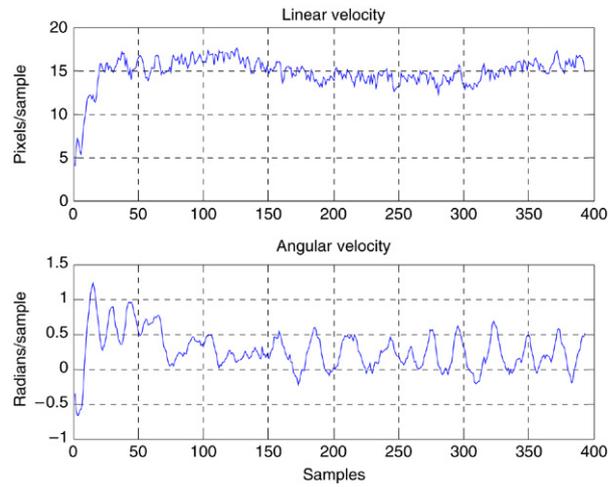


Fig. 13. Evolution of linear and angular velocities.

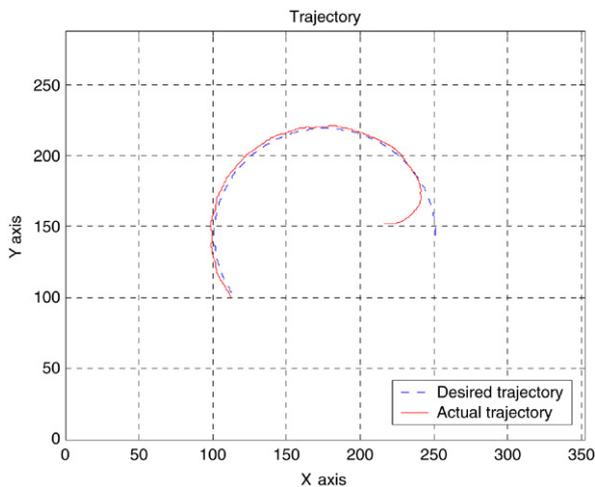


Fig. 11. Robot evolution on the image plane.

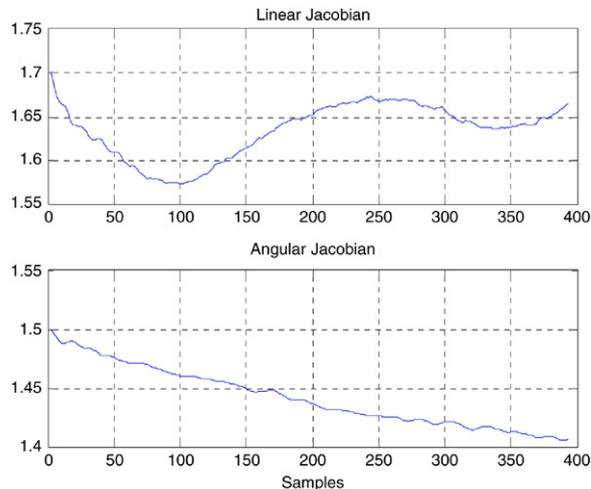


Fig. 14. Evolution of the estimated Jacobians.

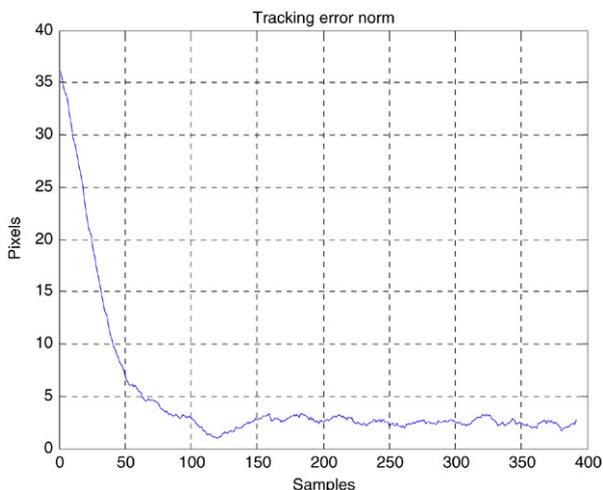


Fig. 12. Evolution of the tracking error.

$k_2 = 6; k_3 = 8; k = 60; a = 50$ . Fig. 15 shows the robot evolution on the image plane. Fig. 16 represents the evolution of the tracking error; Fig. 17 the magnitude of the angular and lineal velocities and Fig. 18, the behavior of the estimated linear and angular Jacobians of this experiment. All the above experiments show a good performance of the designed tracking controller.

A fourth and final experiment is presented to show the performance of the obstacle avoidance strategy described in Section 3.4. To this aim, a straight trajectory similar to the one used in the first experiment has been defined through a circular obstacle. Fig. 19 shows the robot evolution on the image plane while avoiding the obstacle. Fig. 20 represents the evolution of the angle of rotation of the desired velocity vector  $\dot{\mathbf{x}}_d$  and the evolution of the fictitious force.

### 5. Conclusions

This paper has presented a tracking controller for cellular robots and a fixed camera configuration by using direct visual feedback, without knowledge about camera parameters and configuration. The robot's kinematics and dynamics have

In a third experiment, the reference point moves on the image plane describing a third grade polynomial. The tangential velocity is  $v = 12$  pix/s, and the parameters are set to:  $k_1 = 3$ ;

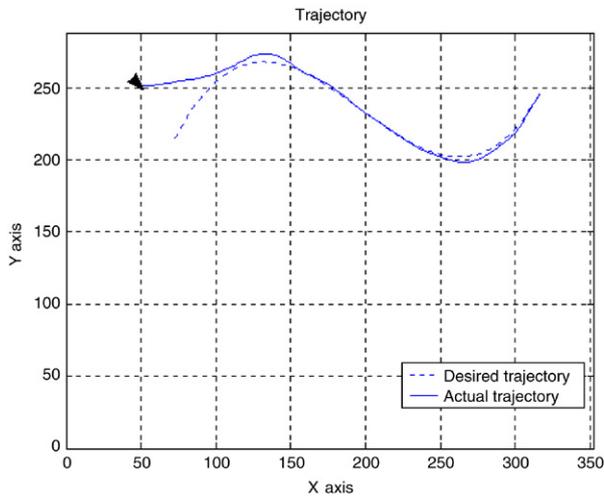


Fig. 15. Robot evolution on the image plane.

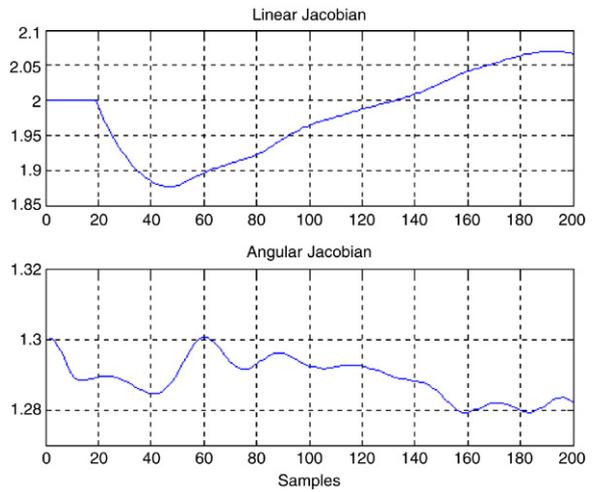


Fig. 18. Evolution of the estimated Jacobians.

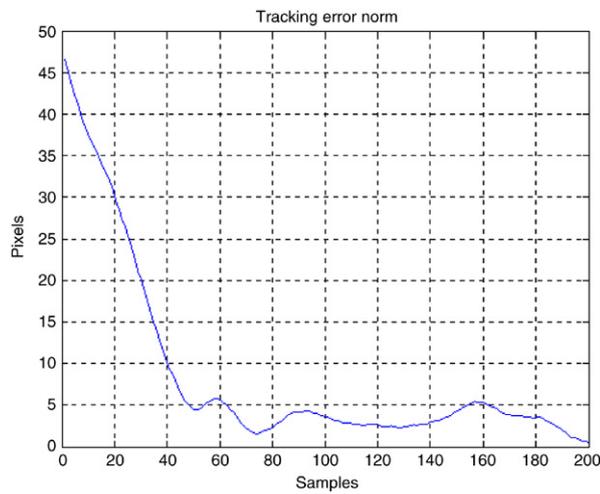


Fig. 16. Evolution of the tracking error.

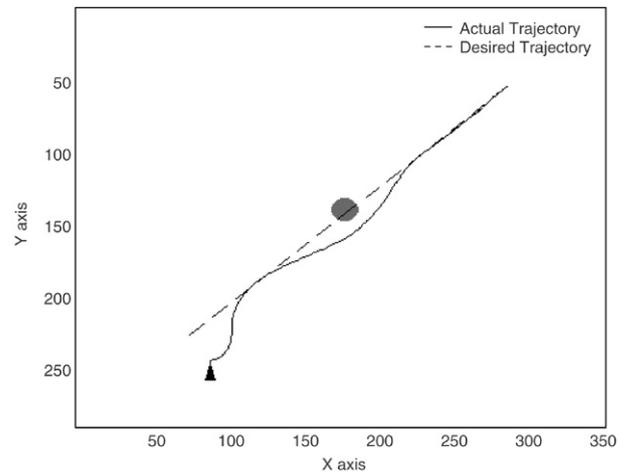


Fig. 19. Robot evolution on the image plane.

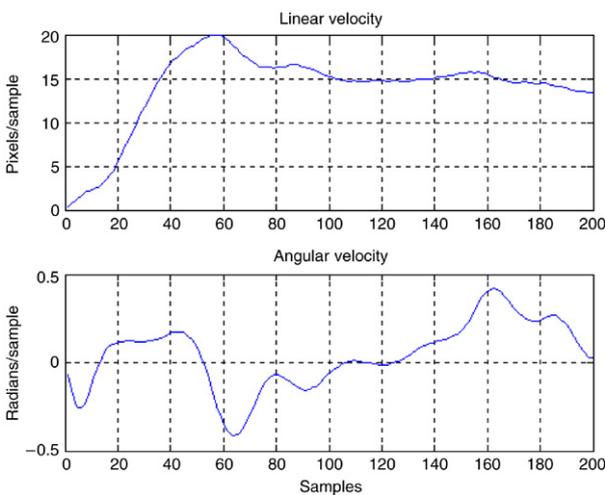


Fig. 17. Evolution the linear and angular velocities.

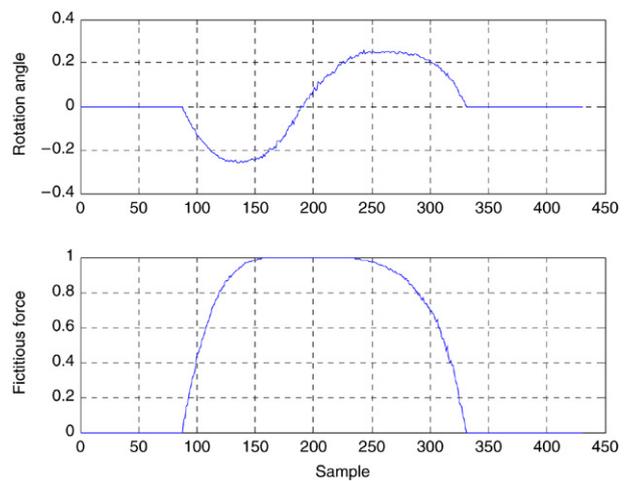


Fig. 20. Evolution of the rotation angle and the fictitious force.

been considered for the design of a non-linear controller. The controller requires the estimation of the linear and angular image Jacobians. A proof of the stability properties

of the proposed control system has been provided, including the effects of the Jacobian estimation errors in the control errors. In addition, an obstacle avoidance strategy has been designed by considering fictitious forces on the image plane.

Finally, representative experiments were presented to show the performance of the proposed control system. As regards the difficulties found implementing the proposed controller, it can be noted that the controller includes many parameters that need to be tuned before operating the system to get a good tracking performance. In addition, it should be emphasized that the trajectory control is performed on the image plane, which does not need any transformation onto the working plane, but requires the definition of some landmarks whose projection on the image plane allows a proper definition of the desired trajectory. For future work, the proposed controller can be easily applied to the problem of controlling multiple cellular robots performing cooperative tasks. Besides, the use of multiple cameras observing the scene, and a proper commutation between them, can contribute to the enlargement of the workspace. More confident measurements can also be achieved if a data fusion from redundant images is applied.

## References

- [1] C. Canudas de Wit, B. Siciliano, G. Bastin, *Theory of Robot Control*, Springer-Verlag, 1996.
- [2] R. Hartley, A. Zisserman, *Multiple View Geometry in Computer Vision*, Cambridge University Press, 2000.
- [3] H.K. Khalil, *Nonlinear Systems*, 2nd ed., Prentice-Hall, New Jersey, 1996.
- [4] J.-J. Slotine, W. Li, *Applied Non Linear Control*, Prentice-Hall, 1991.
- [5] R. Carelli, E. Oliva, C. Soria, O. Nasisi, Combined force and visual control of an industrial robot, *Robotica* 22 (2004) 163–171.
- [6] W. Dixon, D. Dawson, E. Zergeroglu, A. Behal, Adaptive tracking control of a wheeled mobile robot via an uncalibrated camera system, *IEEE Transactions on Systems, Man and Cybernetics* (2001) 31–33.
- [7] P. Kalata, The tracking index: A generalized parameter for  $\alpha - \beta$  and  $\alpha - \beta - \gamma$  target trackers, *IEEE Transactions on Aerospace and Electronic Systems* 20 (2) (1994) 174–182.
- [8] R. Kelly, Robust asymptotically stable visual servoing of planar robots, *IEEE Transactions on Robotics and Automation* 12 (10) (1996).
- [9] Y. Nakabo, M. Ishikawa, Visual impedance using 1ms. visual feedback system, in: *Proc. of the 1998 IEEE Int. Conference of Robotic & Automation*, 1998, pp. 2333–2338.
- [10] J. Santos Victor, Vision-based remote control of cellular robots, *Robotics and Autonomous Systems* (23) (1998) 221–234.
- [11] J. Santos Victor, R. Carelli, S. Van der Zwaan, Nonlinear visual control of remote cellular robots, in: *Mediterranean Control Conference*, Lisbon, 2002.
- [12] H. Secchi, R. Carelli, V. Mut, An experience on stable control of mobile robots, *Latin American Applied Research* 33 (4) (2003) 379–386.
- [13] T. Tsuji, A. Hiromasa, M. Kaneko, Non-contact impedance control for redundant manipulators using visual information, in: *Proc. of the 1997 IEEE Int. Conference of Robotic & Automation*, 1997, pp. 2571–2576.
- [14] C. Zanardi, J.-Y. Herve, P. Cohen, Mutual learning of unsupervised interaction between mobile robots, in: *Proc. Int. Conference on Pattern Recognition, ICPR, Vienna, 1996*.



**Ricardo Carelli** was born in San Juan, Argentina. He graduated in Engineering from the National University of San Juan, Argentina, and obtained a Ph.D. degree in Electrical Engineering from the National University of México (UNAM). He is presently full professor at the National University of San Juan and Senior Researcher at the National Research Council of Argentina (CONICET), and he is also Director of the Instituto de Automática, National University of San Juan. His research interests are in Robotics, Manufacturing Systems, Adaptive Control and Artificial Intelligence applied to Automatic Control. Professor Carelli is a Senior Member of IEEE and a Member of AADECA-IFAC.



**José Santos-Victor** received the Ph.D. degree in Electrical and Computer Engineering in 1995 from Instituto Superior Técnico (IST — Lisbon, Portugal), in the area of Computer Vision and Robotics.

He is an Associate Professor at the Department of Electrical and Computer Engineering of IST and a researcher of the Institute of Systems and Robotics (ISR), at the Computer and Robot Vision Lab — VisLab. (<http://vislab.isr.ist.utl.pt>)

He is the scientist responsible for the participation of IST in various European and national research projects in the areas of Computer Vision and Robotics. His research interests are in the areas of Computer and Robot Vision, particularly in the relationship between visual perception and the control of action, biologically inspired vision and robotics, cognitive vision and visually controlled (land, air and underwater) mobile robots.

Professor Santos-Victor is an IEEE member and an Associate Editor of the *IEEE Transactions on Robotics*.



**Flavio Roberti** was born in Buenos Aires, Argentina. He graduated in Electronic Engineering from the National University of San Juan, Argentina in 2003. He is presently a Ph.D. student in Control Systems Engineering at the Instituto de Automática, National University of San Juan, with a research scholarship from the National Research Council of Argentina (CONICET). His research interests are in Robotics, 3D Visual Servoing and Omnidirectional Vision.



**Santiago Tosetti** was born in San Luis, Argentina. He received the Bachelor's degree in Electronic Engineering from the National University of San Juan, Argentina, in 2003. He is currently a Ph.D. Student in Control Systems Engineering at the Instituto de Automática, National University of San Juan. His research interests include Optimal Control, Adaptive Critic Design, Hybrid Control, Production Systems Control.