

Virtual Target Formulation for Singularity-Free Visual Control Using the Trifocal Tensor

H.M. Becerra¹, J.B. Hayet¹, and C. Sagüés²

¹ Centro de Investigación en Matemáticas (CIMAT),
C.P. 36240, Guanajuato, Gto., Mexico
{hector.becerra, jbhayet}@cimat.mx

² Instituto de Investigación en Ingeniería de Aragón,
Universidad de Zaragoza, C/ María de Luna 1, E-50018, Zaragoza, Spain
csagues@unizar.es

Abstract. We present a novel approach for visual control of wheeled mobile robots, extending the existing works that use the trifocal tensor as source for measurements. In our approach, singularities typically encountered in this kind of methods are removed by formulating the control problem based on the trifocal tensor and by using a virtual target vertical translated from the real target. A *single* controller able to regulate the robot pose towards the desired configuration without local minima is designed. Additionally, the proposed approach is valid for perspective cameras as well as catadioptric systems obeying a central camera model. All these contributions are supported by convincing simulations.

Keywords: Visual control, virtual target, trifocal tensor.

1 Introduction

The use of machine vision in robotics is still a challenging task. In this context, the problem of visual control (visual servoing) of robots is an interesting application of computer vision techniques, control theory and robotics. This problem consists in computing suitable velocities to drive a robot to a desired location associated to a target image previously acquired. Feedback is obtained from visual data by comparing current and target images. Visual servoing (VS) schemes are classified as image-based when image data is used directly in the control loop, or position-based, when pose parameters are needed [1]. Classical approaches use image points as visual features, given that they are easily extracted. Advanced approaches use other geometrical primitives (lines, moments, etc.) or geometric constraints to improve robustness of the control scheme [2–4].

Recently, multiple-view geometry have been exploited for the visual control of mobile robots [5–8]. When designed properly, these schemes avoid the local minima problems of the classical schemes where overdetermined solutions are obtained. The homography-based control relies on planar scenes [5, 6], so that, more general constraints like the ones induced by epipolar and trifocal geometries have been preferred. However, the epipolar geometry is ill-conditioned with short

baseline. Moreover, controllers based on the epipolar constraint are subject to singularities. This has been tackled by using control schemes that switch between two different controllers when degeneracies and singularities appear [3, 7].

The present work has been inspired by [9], where the authors suggest the use of a virtual target in order to avoid some degeneracies of the essential matrix and singularities of an epipolar visual control. In that work, a virtual target is generated relying on the transfer relations given by the essential matrix. However, the transfer of points can fail for certain configurations, like collinear projection centers [10]. In the work herein, we propose the use of the trifocal tensor (TT), which is known to be more robust and better defined than the epipolar geometry. The transfer relations associated with the TT can be used to transfer points and lines from two source images into a third one without degenerate cases. In the literature, the TT has been exploited for image-based visual servoing but with some limitations related to local minima and switching control policies [4, 8].

The contribution of this paper is the formulation to generate and use a virtual target (virtual image) from the 2D TT in the context of visual control of a wheeled mobile robot. The virtual target provides additional information that avoids the need of switching to a different controller in contrast to [3, 7, 8]. A single controller, free of singularities, is derived in order to show the feasibility of using the virtual target. This controller achieves regulation of the robot pose, i.e., position and orientation are simultaneously driven to their desired values. An additional benefit of the proposed VS scheme is that it is valid for different types of cameras, in particular, those obeying the unified projection model [11].

The paper is organized as follows. Section 2 describes the mathematical modeling of the robot and the vision system. Section 3 details the generation of the virtual target relying on the TT. Section 4 describes the controller design from the TT using the virtual target. Section 5 shows the performance of the proposed approach through realistic simulations and Section 6 summarizes the conclusions.

2 Modeling for Visual Control of a Mobile Robot

2.1 Robot Kinematics

Let $\chi = (x, y, \phi)^T$ be the state vector of a differential drive robot shown in Fig. 1(a), where x and y are the robot position coordinates in the plane, and ϕ is its orientation. Assume that a central camera is fixed to the robot in such a way that the robot and camera reference frames coincide. The kinematic model of the camera-robot system expressed in state space can be written as follows:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} -\sin \phi & 0 \\ \cos \phi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \nu \\ \omega \end{bmatrix}, \quad (1)$$

being ν and ω the translational and angular velocities, respectively. In the sequel, the notation $s\phi = \sin \phi$ and $c\phi = \cos \phi$ is used.

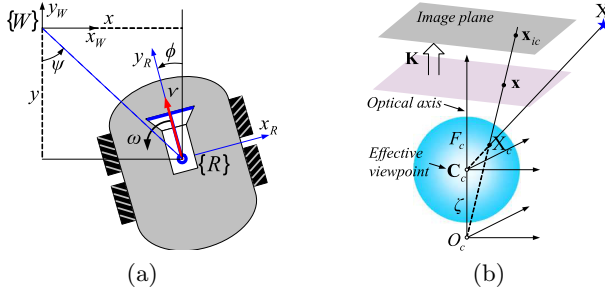


Fig. 1. Representation of the robot model and the camera model. (a) Robot frame definition. (b) Generic camera model of central cameras [11].

2.2 The Trifocal Tensor for Generic Cameras

A desirable feature for a VS scheme is its applicability for different types of cameras, e.g. conventional or omnidirectional. Cameras with wide field of view are preferred to avoid the loss of visual features during motion. Geometric constraints have provided a good strategy to achieve generic control schemes. Hence, we use the generic model of central cameras [11] to exploit the computation of a geometric constraint, like the TT, in the same way for any central vision system.

Let \mathbf{X} be the coordinates of a 3D point. Under the unified projection model, its corresponding point on the unit sphere \mathbf{X}_c can be computed from point \mathbf{x} on the normalized image plane (see Fig. 1(b)) and the sensor parameter ζ as:

$$\mathbf{X}_c = (\eta^{-1} + \zeta) \bar{\mathbf{x}}, \quad \bar{\mathbf{x}} = \left[\mathbf{x}^T \frac{1}{1+\zeta\eta} \right]^T \quad (2)$$

where $\eta = \frac{-\gamma - \zeta(x^2 + y^2)}{\zeta^2(x^2 + y^2) - 1}$, $\gamma = \sqrt{1 + (1 - \zeta^2)(x^2 + y^2)}$. We assume that the camera is calibrated if omnidirectional vision is used, which allows to exploit the representation of the points on the unit sphere. With a conventional camera, calibration is not needed and the TT can be computed from normalized points [10].

The TT encodes the geometry between three views, independently of the scene structure [10]. It has 27 elements (18 independent) and can be expressed using three 3×3 matrices $\mathbf{T} = \{\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3\}$. Here, we use points as image features: Consider three corresponding points projected on the unit sphere \mathbf{p} , \mathbf{p}' and \mathbf{p}'' in three views of a 3D scene, in homogeneous coordinates, i.e. $\mathbf{p} = (p^1, p^2, p^3)^T$. The incidence relation between them is given by

$$[\mathbf{p}']_{\times} \left(\sum_i p^i \mathbf{T}_i \right) [\mathbf{p}'']_{\times} = \mathbf{0}_{3 \times 3} \quad (3)$$

where $[\mathbf{p}]_{\times}$ is the common skew symmetric matrix.

Consider images taken from three different *coplanar* locations, i.e., with a camera moving at a fixed distance from the ground. In this case, several tensor elements are zero and only 12 elements are in general non-null. Fig. 2 depicts the upper view of three cameras with global reference frame in the third view,

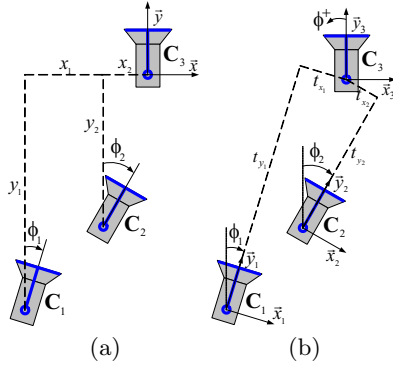


Fig. 2. Geometry between three camera locations in the plane. (a) Absolute locations with respect to a reference frame in \mathbf{C}_3 . (b) Relative locations.

in such a way that the camera locations are $\mathbf{C}_1 = (x_1, y_1, \phi_1)$, $\mathbf{C}_2 = (x_2, y_2, \phi_2)$ and $\mathbf{C}_3 = (0, 0, 0)$. The TT non-null elements can be analytically deduced [4] as:

$$\begin{aligned}
 T_{111}^m &= -t_{x_1}c\phi_2 + t_{x_2}c\phi_1, T_{112}^m = t_{x_1}s\phi_2 + t_{y_2}c\phi_1, T_{121}^m = -t_{y_1}c\phi_2 - t_{x_2}s\phi_1, \\
 T_{122}^m &= t_{y_1}s\phi_2 - t_{y_2}s\phi_1, T_{211}^m = -t_{x_1}s\phi_2 + t_{x_2}s\phi_1, T_{212}^m = -t_{x_1}c\phi_2 + t_{y_2}s\phi_1, \\
 T_{221}^m &= -t_{y_1}s\phi_2 + t_{x_2}c\phi_1, T_{222}^m = -t_{y_1}c\phi_2 + t_{y_2}c\phi_1, \\
 T_{313}^m &= -t_{x_1}, T_{323}^m = -t_{y_1}, T_{331}^m = t_{x_2}, T_{332}^m = t_{y_2}
 \end{aligned} \tag{4}$$

where $t_{x_i} = -x_i c\phi_i - y_i s\phi_i$, $t_{y_i} = x_i s\phi_i - y_i c\phi_i$ for $i = 1, 2$ and where the superscript m indicates metric information. The estimated tensor has an unknown scale factor, changing as the robot moves. We set a common scale by normalizing each tensor element as $T_{ijk} = T_{ijk}^e / T_N$, where T_{ijk}^e are the estimated TT elements obtained, T_{ijk} are the normalized elements, and T_N a suitable normalizing factor. We can see from Eq. 4 that T_{313} and T_{323} are constant and non-null, if $\mathbf{C}_1 \neq \mathbf{C}_3$. Hence, any of these two elements can serve as normalizing factor.

3 A Virtual Target from the TT

In the sequel, as described in Fig. 2, \mathbf{C}_1 , $\mathbf{C}_2(t)$, and \mathbf{C}_3 are respectively the initial, current (at time t) and target camera-robot locations. Notice that \mathbf{C}_1 and \mathbf{C}_3 remain fixed during the motion. The pose regulation problem consists in driving the robot to $\mathbf{C}_2(t) = \mathbf{C}_3$, where the current image observed by the camera (corresponding to $\mathbf{C}_2(t)$) is the same as the previously acquired target image. On the one hand, it is numerically troublesome to estimate the TT if two images are the same and some elements must be discarded for control purposes in that case [4]. On the other hand, the use of the radial TT (first 8 expressions of Eq. 4) has resulted in the need of a few controllers in order to accomplish the pose regulation task [8]. Inspired by [9], we use a TT relative to a new, virtual target location $\bar{\mathbf{C}}_3$. This location is the same as before but *shifted* by a vertical

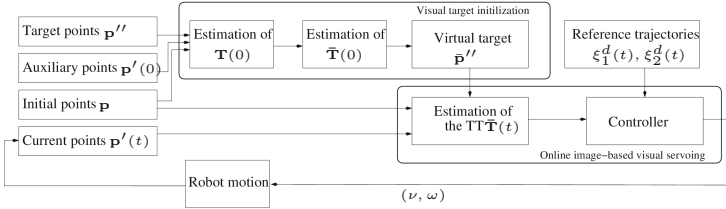


Fig. 3. Overview of the TT-based visual servoing methodology with virtual targets

translation ($t_z > 0$) as shown in Fig. 4. Thus, the *planar* geometry among \mathbf{C}_1 , $\mathbf{C}_2(t)$ and $\bar{\mathbf{C}}_3$ stays the same as in Fig. 2 and the image at the end of the motion is different to the virtual target image ($\bar{\mathbf{C}}_3$). The benefits of using a virtual target are that any non-null element of the new tensor $\bar{\mathbf{T}}(t)$ (corresponding to views at \mathbf{C}_1 , $\mathbf{C}_2(t)$, and $\bar{\mathbf{C}}_3$) may be chosen for control purposes, as they are well-defined.

As depicted in Fig. 3, we define points configurations \mathbf{p} (initial) and \mathbf{p}'' (target). Then, the robot moves to a second configuration from which the control loop starts ($t = 0$). From this configuration, the points \mathbf{p} have been tracked into points $\mathbf{p}'(0)$, and the TT $\mathbf{T}(0)$ (at time 0) is computed. From section 3.1, we deduce the TT associated to the virtual target, $\bar{\mathbf{T}}(0)$, and from section 3.2, we estimate the virtual target position $\bar{\mathbf{p}}''$ to be used in the control loop. Then, for $t > 0$, the control loop uses the entries of the TT computed from \mathbf{p} , $\bar{\mathbf{p}}''$ and $\mathbf{p}'(t)$ (current image) to drive the robot to the target location (see Section 4).

3.1 From Real TT to Virtual TT

At time 0, the TT, $\bar{\mathbf{T}}(0)$ (relating \mathbf{p} , $\mathbf{p}'(0)$, $\bar{\mathbf{p}}''$) differs from $\mathbf{T}(0)$ (as in Eq. 4, computed from \mathbf{p} , $\mathbf{p}'(0)$, \mathbf{p}'') by the following elements:

$$\begin{aligned} \bar{T}_{113}^m(0) &= t_{z_2} c\phi_1, \bar{T}_{123}^m(0) = -t_{z_2} s\phi_1, \bar{T}_{131}^m(0) = -t_{z_1} c\phi_2(0), \bar{T}_{132}^m(0) = t_{z_1} s\phi_2(0), \\ \bar{T}_{213}^m(0) &= t_{z_2} s\phi_1, \bar{T}_{223}^m(0) = t_{z_2} c\phi_1, \bar{T}_{231}^m(0) = -t_{z_1} s\phi_2(0), \bar{T}_{232}^m = -t_{z_1} c\phi_2(0) \end{aligned} \quad (5)$$

where $t_{z_1} = t_{z_2} = t_z$, as the global reference frame is now $\bar{\mathbf{C}}_3$. The distance t_z is arbitrary but we recommend the unity. The angles ϕ_1 and $\phi_2(0)$ (evolving as $\phi_2(t)$ in the control loop) are estimated from $\mathbf{T}(0)$ (Eq. 4) as follows:

$$\phi_1 = \arcsin\left(\frac{T_{323}T_{212} - T_{313}T_{121}}{T_{323}T_{332} + T_{313}T_{331}}\right), \quad \phi_2(0) = \arccos\left(\frac{T_{332}T_{121} + T_{331}T_{212}}{T_{323}T_{332} + T_{313}T_{331}}\right). \quad (6)$$

Hence, by using both Eq. 5 and Eq. 6, we can deduce $\bar{\mathbf{T}}(0)$.

3.2 Generating the Virtual Target

Generating the virtual target implies transferring the points seen at \mathbf{C}_1 and $\mathbf{C}_2(0)$ into the view corresponding to $\bar{\mathbf{C}}_3$. If \mathbf{p} and $\mathbf{p}'(0)$ are two such points at

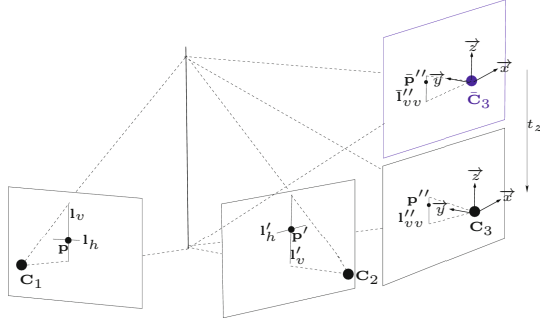


Fig. 4. Generation of a virtual target. From \mathbf{T} , associated to \mathbf{C}_1 , \mathbf{C}_2 , \mathbf{C}_3 , and t_z , the tensor $\bar{\mathbf{T}}$ associated to \mathbf{C}_1 , \mathbf{C}_2 , $\bar{\mathbf{C}}_3$ is estimated and used for point transfer.

\mathbf{C}_1 and $\mathbf{C}_2(0)$, detected independently, then in general, they do not satisfy the epipolar constraint induced by the first two views. Hence, Eq. 3 has no solution.

A simple workaround uses the fact that the TT can give a unique transferred straight line for the third view, given two straight lines equations in the first and second views. As depicted in Fig. 4, consider a pair of horizontal/vertical lines $(\mathbf{l}_h, \mathbf{l}_v)$ through \mathbf{p} , and a second pair of lines $(\mathbf{l}'_h, \mathbf{l}'_v)$ through $\mathbf{p}'(0)$. By construction, transferred points $\bar{\mathbf{p}}''$ should belong to any straight line made from pairs $(\mathbf{l}_h, \mathbf{l}'_h)$, $(\mathbf{l}_h, \mathbf{l}'_v)$, $(\mathbf{l}_v, \mathbf{l}'_h)$, $(\mathbf{l}_v, \mathbf{l}'_v)$, each of which being computed with $\bar{\mathbf{T}}(0)$. In Fig. 4, the process is illustrated for the pair $(\mathbf{l}_v, \mathbf{l}'_v)$, which image is given by [10]: $\bar{\mathbf{l}}''_{vv} = (\mathbf{l}_v^T \bar{\mathbf{T}}_1(0) \mathbf{l}'_v, \mathbf{l}_v^T \bar{\mathbf{T}}_2(0) \mathbf{l}'_v, \mathbf{l}_v^T \bar{\mathbf{T}}_3(0) \mathbf{l}'_v)^T$. Each of these straight line pairs leads to a linear constraint on the coordinates of $\bar{\mathbf{p}}''$. Hence, a linear system can be formed to determine the coordinates of $\bar{\mathbf{p}}'' \in \mathbb{R}^3$ in a least square sense $\bar{\mathbf{p}}'' = \min_{\|\bar{\mathbf{q}}''\|=1} \|\mathbf{L}\bar{\mathbf{q}}''\|^2$ where \mathbf{L} is a 4×3 matrix containing the line equations. Then, $\bar{\mathbf{p}}''$ is extracted as the singular vector of \mathbf{L} with the smallest singular value.

4 Visual Control Using the Virtual Target

In the literature, switching control laws have been proposed to solve the pose regulation problem of mobile robots by exploiting geometric constraints [3, 7, 8]. At least two different controllers with an appropriate switching policy are used in such approaches to deal with degeneracies and singularities. In this section, we present a single controller capable to drive the robot to the target pose (position and orientation). Hereafter, we denote $\mathbf{T} = \bar{\mathbf{T}}(t)$ for clarity.

4.1 Input-Output Linearization

After analysis of the information provided by the TT estimated from the virtual target, we have chosen the following measurements as outputs of our system:

$$\xi_1 = T_{332}, \quad \xi_2 = \frac{T_{132}}{T_{131}}. \quad (7)$$

Note that the first tensor element comes from the “true” trifocal tensor, while the last two elements come from the virtual one. Hereafter, we will denote $x = x_2$, $y = y_2$ and $\phi = \phi_2$, the robot pose in the plane w.r.t. the reference frame attached to $\bar{\mathbf{C}}_3$. The outputs are related to the camera-robot state as follows:

$$\xi_1 = \alpha (x s \phi - y c \phi), \quad \xi_2 = -\tan \phi$$

where α is an unknown scale factor. An important consideration is that these outputs are valid in the range $|\phi| < \pi/2$, so that, we assume that the initial orientation of the robot accomplish such condition. Notice that both outputs are null if the robot has reached the target location. It can be seen that $\xi_1 = 0$ and $\xi_2 = 0$ imply that $\phi = 0$, $y = 0$ and x is a degree of freedom of the solution, which means that the orientation and longitudinal error are corrected while the lateral error may be different from zero (zero dynamics from the control theory point of view [12]). However, given that ξ_1 is related to the longitudinal position and ξ_2 depends directly on the orientation, the lateral deviation can be corrected with the tracking of an adequate trajectory for ξ_2 as the robot moves forward. It is desired to drive the outputs to zero in a fixed time horizon, which is a trajectory tracking control problem. Let us define the following error functions:

$$e_1 = \xi_1 - \xi_1^d, \quad e_2 = \xi_2 - \xi_2^d$$

where ξ_1^d and ξ_2^d are smooth desired trajectories with null final value. The tracking problem can be faced by using the input-output linearization technique [12]. It needs the time derivatives of the error functions, given by:

$$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \end{bmatrix} = \begin{bmatrix} -\alpha & -T_{331} \\ 0 & -\frac{T_{131}^2 + T_{132}^2}{T_{131}^2} \end{bmatrix} \begin{bmatrix} \nu \\ \omega \end{bmatrix} + \begin{bmatrix} \dot{\xi}_1^d \\ \dot{\xi}_2^d \end{bmatrix}. \quad (8)$$

This system can be written as $\dot{\mathbf{e}} = \mathbf{J}\mathbf{u} + \dot{\boldsymbol{\xi}}^d$, where \mathbf{J} is the interaction matrix that relates the robot velocities to the rate of change of the visual measurements of Eq. 7. In order to find out adequate robot velocities to track the desired trajectories, the error system must be inverted, which is possible given that $\det(\mathbf{J}) = \alpha \frac{T_{131}^2 + T_{132}^2}{T_{131}^2} = \frac{\alpha}{\cos^2 \phi} \neq 0$. Hence, the robot velocities are given by:

$$\begin{bmatrix} \nu \\ \omega \end{bmatrix} = \begin{bmatrix} -\frac{1}{\alpha} & \frac{T_{331} T_{131}^2}{\alpha(T_{131}^2 + T_{132}^2)} \\ 0 & -\frac{T_{131}^2}{T_{131}^2 + T_{132}^2} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \quad (9)$$

where v_1, v_2 are auxiliary control inputs that define the error functions convergence. Different options exist to assign the auxiliary controls but their derivation is not in the scope of the paper. In the results section, the super-twisting control technique [13] is used to show the applicability of the controller of Eq. 9.

4.2 Desired Trajectories

The desired trajectories must drive smoothly the outputs of the system from their initial values to zero in a fixed time horizon. Thus, the desired trajectory for ξ_1 is always defined as follows:

$$\xi_1^d = \frac{\xi_1(0)}{2} \left(1 + \cos \left(\frac{\pi t}{\tau} \right) \right), \quad 0 \leq t \leq \tau$$

$$\xi_1^d = 0, \quad t > \tau$$

where τ is a user-defined time horizon in which the robot reaches the target location. Given that the orientation control also must drive the robot to correct the lateral deviation, a desired trajectory ξ_2^d related to x is proposed. Let us define the angle ψ as in Fig. 1(a), related to the lateral deviation:

$$\psi = \arctan \left(-\frac{T_{332}s\phi - T_{331}c\phi}{T_{332}c\phi + T_{331}s\phi} \right)$$

where ϕ is given by Eq. 6. Then, the desired trajectory for ξ_2^d is given by:

$$\xi_2^d = \xi_2(0) \frac{\psi(t)}{\psi(0)}, \quad 0 \leq t \leq \tau$$

$$\xi_2^d = 0, \quad t > \tau. \quad (10)$$

This trajectory is used if $|\phi(0)| > |\psi(0)|$, otherwise, an initial motion is induced by smoothly increasing the desired value ξ_2^d for an initial period $\tau_1 < \tau$.

5 Simulation Results

The performance of our virtual target-based approach is evaluated via simulations. The results have been obtained by using Matlab scripting with a closed loop time of 0.3s. The TT is estimated from synthetic images of size 800×600 pixels. Omnidirectional images are generated through the generic camera model [11]. The time to complete the regulation task is set to $\tau = 90$ s.

The resulting paths, from three different initial locations, can be seen in Fig. 5(a). The case of the initial location at $\mathbf{L}_1 = (0, -11, 0^\circ)$ is special, given that $\phi(0) = \psi(0) = 0$. In such a case, $\xi_2^d = 0$ during the navigation. This case is also special because an epipolar control is not able to solve it. In the case of the initial location at $\mathbf{L}_2 = (4, -9, 40^\circ)$, ξ_2^d is defined by Eq. 10 and similarly for $\mathbf{L}_3 = (-5, -13, 0^\circ)$, by including an initial increment of ξ_2^d . In all cases, the robot reaches the target with good precision and carries out a smooth motion, as shown in Fig. 5(b). This behavior is obtained with the velocities of Fig. 5(c), given by the trajectory tracking scheme of Eq. 9. In Fig. 5(d), the evolution of the visual measurements of Eq. 7, taken as outputs of the system, are shown. Due to the normalization of ξ_1 , the plots for the three cases look similar.

In order to validate the proposed VS scheme for different cameras, the motion of the point features is also presented in Fig. 5(e) for the control from \mathbf{L}_2 with a hypercatadioptric projection. For the control from \mathbf{L}_3 , a paracatadioptric vision system is used and the motion of the image points is shown in Fig. 5(f). Note that the points at the end of the motion, marked with “×”, are overlapped to the real target points, marked with “O”. The difference between the former points

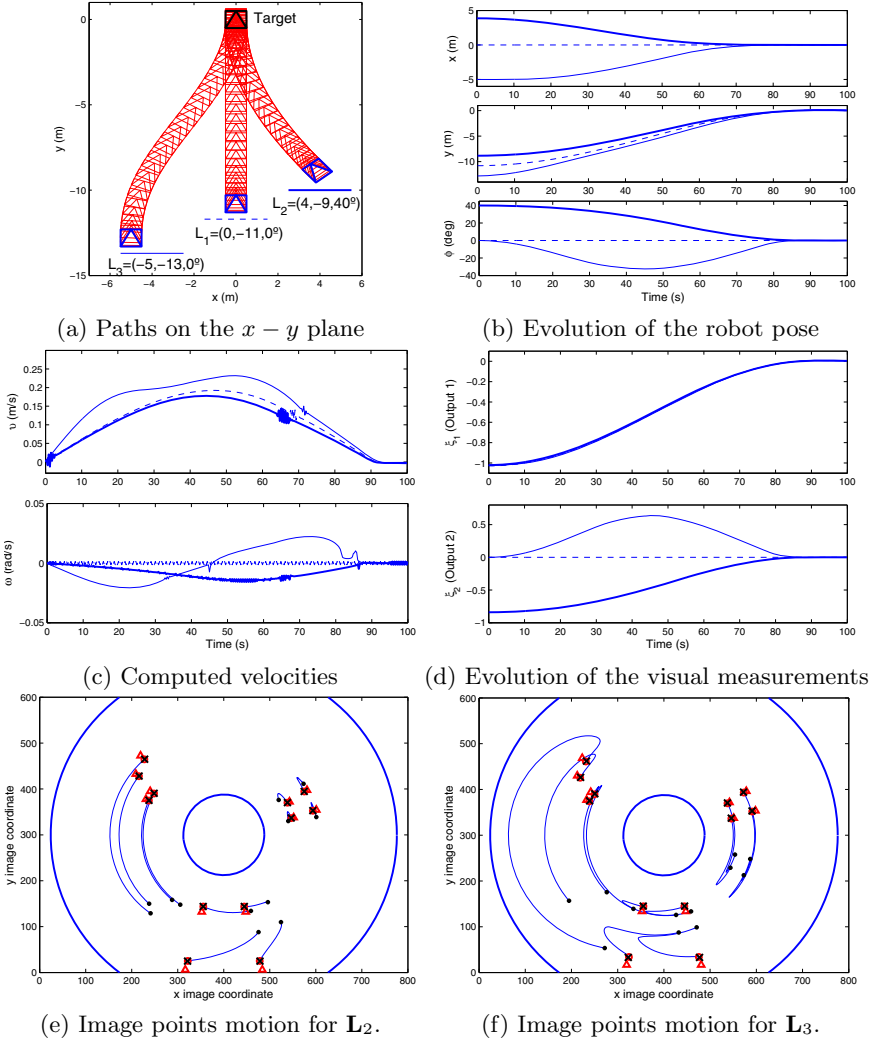


Fig. 5. Simulation results for vision-based pose regulation using images from hypercatadioptric (e) and paracatadioptric (f) vision systems. The markers represent: “.”=initial image, “O”=real target image, “ Δ ”=virtual target image and “ \times ”=image at the end.

and the virtual target points, marked with “ Δ ”, avoids numerical problems at the end of the task in the estimation of the TT and makes possible the derivation of our singularity-free controller. Note that we use a calibrated camera but our control scheme is also valid for *uncalibrated* conventional cameras.

6 Conclusions

We have proposed a novel image-based visual servoing scheme that relies on the evaluation of the trifocal tensor computed by using an automatically generated virtual target. The pose regulation problem of a wheeled mobile robot is solved efficiently using the new control scheme, which is based on a few elements of the trifocal tensor relating the initial view from the robot, the current one and the target view. Contrary to other multiple view geometry-based approaches, we avoid control singularities by using the virtual target, easily deduced from the real one by a vertical translation. Thus, a single controller, tracking adequate trajectories, solves the pose regulation problem without the need of switching to a different controller as in previous works. We have illustrated the pertinence of this approach with very promising simulation results. Our ongoing work focuses on real-world experiments with this approach, with different camera models.

References

1. Chaumette, F., Hutchinson, S.: Visual servo control part I: Basic approaches. *IEEE Robotics and Aut. Mag.* 13(14), 82–90 (2006)
2. Chaumette, F., Hutchinson, S.: Visual servo control, part II: Advanced approaches. *IEEE Robotics and Aut. Mag.* 14(1), 109–118 (2007)
3. Mariottini, G.L., Oriolo, G., Prattichizzo, D.: Image-based visual servoing for nonholonomic mobile robots using epipolar geometry. *IEEE Transactions on Robotics* 23(1), 87–100 (2007)
4. López-Nicolás, G., Guerrero, J.J., Sagüés, C.: Visual control through the trifocal tensor for nonholonomic robots. *Robotics and Auton. Syst.* 58(2), 216–226 (2010)
5. Fang, Y., Dixon, W.E., Dawson, D.M., Chawda, P.: Homography-based visual servo regulation of mobile robots. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 35(5), 1041–1050 (2005)
6. Usai, A., Di Giamberardino, P.: Visual feedback for nonholonomic mobile robots: Homography based approach. In: Rigatos, G. (ed.) *Intelligent Industrial Systems*, pp. 152–181. IGI Global (2010)
7. Becerra, H.M., López-Nicolás, G., Sagüés, C.: A sliding mode control law for mobile robots based on epipolar visual servoing from three views. *IEEE Transactions on Robotics* 27(1), 175–183 (2011)
8. Becerra, H.M., López-Nicolás, G., Sagues, C.: Omnidirectional visual control of mobile robots based on the 1D trifocal tensor. *Robotics and Autonomous Systems* 58(6), 796–808 (2010)
9. López-Nicolás, G., Sagüés, C., Guerrero, J.J.: Parking with the essential matrix without short baseline degeneracies. In: *IEEE Int. Conf. on Robotics and Automation*, pp. 1098–1103 (2009)
10. Hartley, R.I., Zisserman, A.: *Multiple View Geometry in Computer Vision. 2*. Cambridge University Press (2004)
11. Geyer, C., Daniilidis, K.: A unifying theory for central panoramic systems and practical implications. In: Vernon, D. (ed.) *ECCV 2000*. LNCS, vol. 1843, pp. 445–461. Springer, Heidelberg (2000)
12. Slotine, J.J.E., Li, W.: *Applied nonlinear control*. Prentice Hall (1991)
13. Levant, A., Fridman, L.: Higher order sliding modes. In: Perruquetti, W., Barbot, J. (eds.) *Sliding Mode Control in Engineering*, Marcel Dekker, N.Y., USA, pp. 53–101 (2002)