Citation: H. M. Becerra, G. López-Nicolás, and C. Sagüés. Omnidirectional Visual Control of Mobile Robots based on the 1D Trifocal Tensor. Robotics and Autonomous Systems, vol. 58, no. 6, pp. 796-808, 2010

Omnidirectional Visual Control of Mobile Robots based on the 1D Trifocal Tensor

H. M. Becerra[1], G. López-Nicolás and C. Sagüés

Dept. Informática e Ingeniería de Sistemas - Instituto de Investigación en Ingeniería de Aragón, Universidad de Zaragoza, 50018 Zaragoza, Spain
{*hector.becerra, gonlopez, csagues* }*@unizar.es*

Abstract

The precise positioning of robotic systems is of great interest particularly in mobile robots. In this context, the use of omnidirectional vision provides many advantages thanks to a wide field of view. This paper presents an image-based visual control to drive a mobile robot to a desired location, which is specified by a target image previously acquired. It exploits the properties of omnidirectional images to preserve bearing information by using the 1D trifocal tensor. The main contribution of the paper is that the elements of the tensor are introduced directly in the control law and neither any a prior knowledge of the scene nor any auxiliary image are required. Our approach can be applied with any visual sensor obeying approximately a central projection model, presents good robustness to image noise, and avoids the problem of short baseline by exploiting the information of three views. A sliding mode control law in a square system ensures stability and robustness for the closed loop. The good performance of the control system is proven via simulations and real world experiments with a hypercatadioptric camera.

Key words: Visual control, Omnidirectional images, 1D trifocal tensor, Mobile robots.

1 Introduction

Visual servoing is an interesting research field that involves computer vision and control theory in order to command the robot motion. In particular, visual servoing allows mobile robots to improve their navigation capabilities in a single robot task or in cooperative tasks. This paper describes an approach to drive a wheeled mobile robot equipped with an omnidirectional camera on board to a desired location,

[1] Corresponding author. E-mail: hector.becerra@unizar.es

which is specified by a target image previously acquired, i. e., using a teach-by-showing strategy. On visual servoing, a robot is steered to a desired location by minimizing an error function that relates visual data, typically from two images: the current and the target one. We propose to take advantage of more information by using three views.

Along the years, the research on visual servoing has dedicated important efforts to find suitable error functions in order to obtain a desired behavior of the robotic system in terms of stability and robustness of the closed loop control. The basic approaches are typically separated in image-based visual servoing (IBVS), in which the error function consists of a set of features that are directly available in the image data, and position-based visual servoing (PBVS), in which a set of 3D parameters must be estimated from image measurements [1]. Subsequently, many approaches have been proposed as hybrid schemes [2]. Among the advanced approaches, some schemes that are based on a geometric constraint have been proposed. A geometric constraint is a robust way to relate features that are observed in different views of the same scene. Nowadays, two geometric constraints have been exploited for mobile robots control: epipolar geometry and the homography model. Examples of epipolar visual control are [3] and [4]. In these works, the epipoles are directly used to compute the control inputs for the robot. The homography model has been used in several visual servoing schemes, for instance [5] and [6]. In the last, the elements of the homography matrix are used directly in a control law for mobile robots. However, it is known that these both geometric constraints have serious drawbacks. Epipolar geometry is ill-conditioned with short baseline and with planar scenes. The homography model is not well defined if there is no dominant planes in the scene.

Most of the visual servoing schemes have the drawback that the target may leave the camera field of view during the servoing, which leads to failure because the feedback error cannot be computed any more. In this context, the use of wide field of view cameras becomes a very good option to overcome this issue, although some strategies have been proposed for conventional cameras, for instance [7]. One effective way to enhance the field of view is to use mirrors in conjunction with lenses, i. e. catadioptric image formation systems. Some of the pioneer works proposing the use of catadioptric cameras for visual servoing are [8], [9]. The approach of exploiting a geometric constraint has been also explored in omnidirectional visual servoing using the epipolar geometry [10] and the homography model [11]. Additionally, there are different approaches for omnidirectional vision-based robot navigation that exploit particular properties of omnidirectional images, for instance [12] and [13]. More related work on omnidirectional vision can be found in publications of the Omnivis workshop.

In order to overcome the drawbacks of the typical geometric constraints, we propose a novel approach based on the 1D trifocal tensor (TT) which allows us to exploit the information given by three available images: the initial, the current and the target ones. The 1D TT completely describes the relative geometry of three views and is independent of the observed scene [14]. The first work that proposes a robotic application of a trilinear constraint is [15]. In that work, conventional perspective cameras are converted to 1D virtual cameras through a transformation of bearing measurements. In the context of computer vision, the same idea is introduced to wide-angle cameras as a tool for calibrating the radial distortion in [16]. The same authors present a general hybrid trifocal constraint by representing conventional and omnidirectional cameras as radial 1D cameras in [17]. They assert that the radial 1D camera model is sufficiently general to represent the great majority of omnidirectional cameras under the assumption of knowing the center of radial distortion. The effectiveness of applying the 1D TT to recover location information has been also proved in [18]. It uses the TT with both conventional and omnidirectional cameras for scene reconstruction, and proposes this approach for initialization of bearing-only SLAM algorithms. The radial TT has been also proposed for hierarchical localization exploting omnidirectional images in [19]. A recent work presents a visual control for mobile robots based on the elements of a 2D trifocal tensor constrained to planar motion [20].

We propose in this paper an image-based approach to perform visual servoing for mobile robots. The visual control is performed using the value of the elements of the 1D TT directly in the control law. The approach is suitable for all central catadioptric cameras and even for fisheye cameras, since all of these imaging systems present high radial distortion but they preserve the bearing information, which is the only required data in our approach. This paper is an extension of [21], where a visual control based on the 1D TT obtained from metric information is introduced for conventional cameras. However, since the constrained field of view of conventional cameras, it is a need to use omnidirectional images for this approach. The extension presented here is justified along the paper and as part of the results, realistic simulations with synthetic images, which are generated using the unified model for central catadioptric cameras [22], are reported. We have tested the robustness of the control law under image noise and the general performance is also analyzed through real world experiments with images of a hypercatadioptric camera. The approach does not require any a priori knowledge of the scene and does not need any auxiliary image. We propose a two-step control law, the first step performs position correction and the second one corrects orientation. Our approach ensures total correction of the robot pose even for initial locations where epipolar geometry or homography based approaches fail. In comparison with a typical IBVS approach, the proposed scheme allows us to prove stability of the closed loop on the basis of a square control system. Additionally, from a control theory point of view, we have incorporated robustness properties to the system by using sliding mode control. The field of application of this approach is referred to differential-drive robots constrained to planar motion. It results of great interest in many areas, specially for service robots, for which our approach could be applied for navigation together with a SLAM scheme like the ones in [23] and [24].

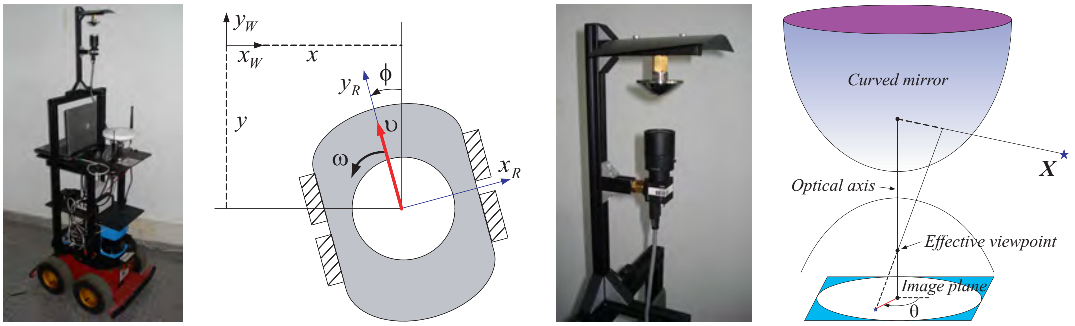
The paper is organized as follows. Section 2 specifies the mathematical modeling of the mobile robot and the 1D TT geometric constraint. Section 3 details the design procedure of the control law. Section 4 presents the stability analysis. Section 5 shows the performance of the control system via simulations with synthetic images, experimental analysis with real images and real world experiments in closed loop. Finally, Section 6 provides the conclusions.

2 Mathematical Modeling

2.1 Robot and Camera Modeling

This work focuses on controlling a wheeled mobile robot through the information given by an omnidirectional imaging system mounted onboard as shown in Fig. 1(a) and under the framework that is depicted in Fig. 1(b). The kinematic model of this kind of robot as expressed in state space is the following

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \left[\begin{array}{c|c} -\sin\phi & 0 \\ \hline \cos\phi & 0 \\ \hline 0 & 1 \end{array}\right] \begin{bmatrix} \upsilon \\ \omega \end{bmatrix}. \tag{1}$$



(a) (b) (c) (d)

Fig. 1. Description of the robot-camera configuration. (a) Our robotic platform with the omnidirectional camera on top. (b) Robot frame definition. (c) Omnidirectional vision system with a Sony camera and Neovision mirror. (d) A generic central catadioptric system.

Thus, $\mathbf{x} = (x, y, \phi)^T$ represents the state of the robotic platform, where x and y are the coordinates of the robot position in the plane and ϕ is the robot orientation. Additionally, υ and ω are the translational and rotational input velocities, respectively. From now on, we use the notation $s\beta = \sin\beta$, $c\beta = \cos\beta$. Note that the model (1) also describes the camera motion, because of the fact that the robot frame is defined in such a way that the optical axis coincides with the rotation axis of the robot, i.

e. the camera is looking upward. Fig. 1(c) shows a detailed view of our omnidirectional imaging system used in real experiments. It is a hypercatadioptric system, for which we can assume that there exist a unique effective viewpoint (Fig. 1(d)). This optic arrangement is popular for robotic applications because it is constructed just with a perspective camera and a hyperboloidal mirror. These systems, as well as those using paraboloidal and ellipsoidal mirrors, have been well studied in the field of computer vision [25], and according to this theory, all of them satisfie the fixed view point constraint. In practice, with a careful construction of the system, it is realistic to assume a central configuration and many robotic applications have proven its effectiveness [8], [9], [10], [18].

2.2 The 1D Trifocal Tensor

Our approach is based on a direct feedback of the information given by a geometric constraint, the 1D trifocal tensor. A similar idea has been exploited for mobile robots control by using the epipolar geometry relating omnidirectional views [10]. The fundamental epipolar constraint is analogue for conventional perspective as that for central catadioptric cameras if it is formulated in terms of rays which emanate from the effective viewpoint [26]. In a similar way, the 1D TT estimation is basically the same for conventional and central catadioptric cameras. The 1D TT particularly adapts to the property of omnidirectional images to preserve bearing information regardless of the high radial distortion induced by lenses and mirrors. Fig. 1(d) shows the bearing angle of an observed feature, which is measured with respect to a frame centered in the principal point of the image. Thus, a bearing measurement θ can be converted to its projective formulation in a 1D virtual retina as $\mathbf{m} = (\sin\theta, \, \cos\theta)^T$. By relating this representation for three different views of a feature that is expressed in a 2D projective space, it results in a trifocal constraint

$$\sum_{i=1}^{2}\sum_{j=1}^{2}\sum_{k=1}^{2} T_{ijk}\mathrm{u}_i\mathrm{v}_j\mathrm{w}_k = 0 \tag{2}$$

where $\mathbf{u} = (\mathrm{u}_1{,}\mathrm{u}_2)^T$, $\mathbf{v} = (\mathrm{v}_1{,}\mathrm{v}_2)^T$ and $\mathbf{w} = (\mathrm{w}_1{,}\mathrm{w}_2)^T$ are the image coordinates of a feature projected in the 1D virtual retina of the first, second and third camera respectively, and T_{ijk} are the eight elements of the homogeneous trifocal tensor. The described representation of bearing measurements is sufficiently general to model from pin-hole cameras to omnidirectional ones, as shown in [17]. Moreover, it permits to compute a mixed trifocal constraint for heterogeneous cameras. In our case, the three images are captured by the same omnidirectional camera, i. e. the homogeneous case. In order to compute the eight elements of the 1D TT we have to solve the linear system of equations obtained from seven stacked trifocal constraints (2). Thus, in general, seven triples of matched features are required to solve for the 1D TT linearly.

Let us define a global reference frame as depicted in Fig. 2(a) with the origin in the third camera. Then, the camera locations with respect to that global reference are $\mathbf{C}_1 = (x_1, y_1, \phi_1)$, $\mathbf{C}_2 = (x_2, y_2, \phi_2)$ and $\mathbf{C}_3 = (x_3, y_3, \phi_3) = (0, 0, 0)$. We assume that the motion is constrained to be planar. The relative locations between cameras are defined by a local reference frame in each camera as shown in Fig. 2(b).

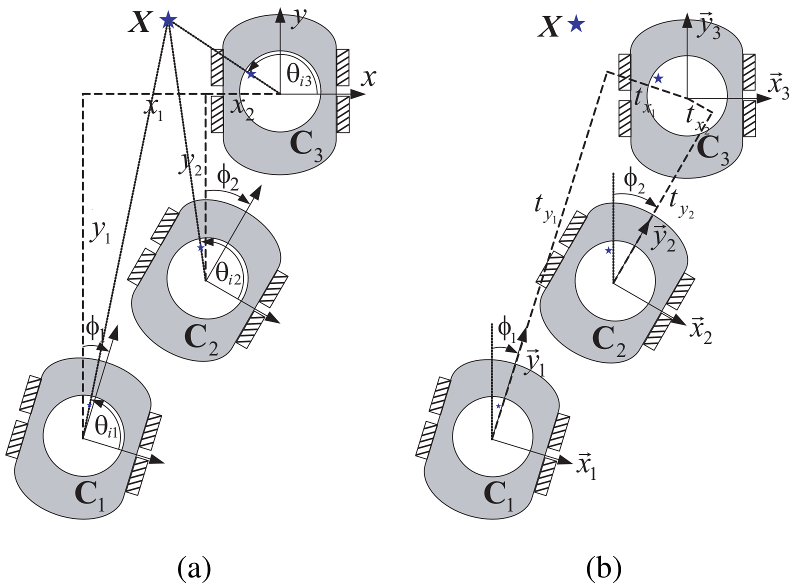


Fig. 2. Framework of the three-view geometry. (a) Global reference definition with origin in the third view and bearing measurements θ of a feature. (b) Relative location between cameras with a fixed reference frame on each view.

Under the framework previously described, the camera motion parameters are related to the 1D TT elements as follows

$$\mathbf{T}_1 = \begin{bmatrix} T_{111} & T_{112} \\ T_{121} & T_{122} \end{bmatrix} = \left[\begin{array}{c|c} t_{y_1} s\phi_2 - t_{y_2} s\phi_1 & -t_{y_1} c\phi_2 + t_{y_2} c\phi_1 \\ \hline t_{y_1} c\phi_2 + t_{x_2} s\phi_1 & t_{y_1} s\phi_2 - t_{x_2} c\phi_1 \end{array}\right], \tag{3}$$

$$\mathbf{T}_2 = \begin{bmatrix} T_{211} & T_{212} \\ T_{221} & T_{222} \end{bmatrix} = \left[\begin{array}{c|c} -t_{x_1} s\phi_2 - t_{y_2} c\phi_1 & t_{x_1} c\phi_2 - t_{y_2} s\phi_1 \\ \hline -t_{x_1} c\phi_2 + t_{x_2} c\phi_1 & -t_{x_1} s\phi_2 + t_{x_2} s\phi_1 \end{array}\right]$$

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 $i = 1, 2$. Some details on deducing the trifocal constraint (2) and the expressions in (3) can be seen in [18]. Two additional constraints accomplished when the radial TT is computed from a calibrated retina are $-T_{111}+T_{122}+T_{212}+T_{221} = 0$, and $T_{112}+T_{121}+T_{211}+T_{222} = 0$.

These calibration constraints allow us to estimate the 1D TT from only five triplets of point correspondences, which improves the estimation [18]. Fig. 3 shows an example of the SIFT [27] point matches (34 good matches) used to compute the tensor as will be described later in Section 5.

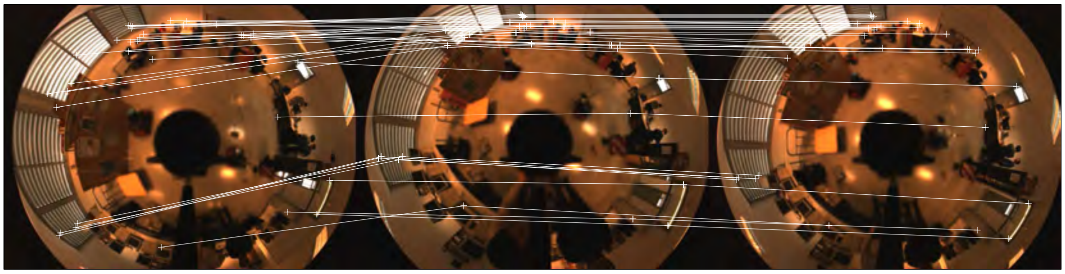


Fig. 3. Robust SIFT matching between three omnidirectional images with translation and rotation between them. The lines between images show 34 corresponding features, which have been extracted using SIFT and matched robustly to be the entries for the 1D TT estimation algorithm.

2.3 Values of the TT in Particular Locations

Let us define the initial location of the robot to be (x_1, y_1, ϕ_1), the target location $(x_3, y_3, \phi_3) = (0, 0, 0)$ and (x_2, y_2, ϕ_2) the current location, which varies as the robot moves. It is worth emphasizing that $\mathbf{C}_1$ could be the moving camera and similar overall results may be obtained. In order to design a controller to drive a robot to a target location using only the tensor elements, we have to consider the corresponding final tensor values as control objective. Initially, when the second camera is in the starting location then $\mathbf{C}_2 = \mathbf{C}_1$, i.e., $(x_2, y_2, \phi_2) = (x_1, y_1, \phi_1)$, the relative location between these cameras is $t_{x_2} = t_{x_1}$, $t_{y_2} = t_{y_1}$ and the values of the tensor elements produce the following relationships

$$T_{111} = 0,\ T_{112} = 0,\ T_{221} = 0,\ T_{222} = 0,\ T_{121} + T_{211} = 0,\ T_{122} + T_{212} = 0. \quad (4)$$

When the robot is in the goal location $\mathbf{C}_2 = \mathbf{C}_3$, i.e., $(x_2, y_2, \phi_2) = (0, 0, 0)$, the relative location between these cameras is $t_{x_2} = 0$, $t_{y_2} = 0$, and it yields the following relationships

$$T_{111} = 0,\ T_{122} = 0,\ T_{211} = 0,\ T_{222} = 0,\ T_{112} + T_{121} = 0,\ T_{212} + T_{221} = 0. \quad (5)$$

2.4 Dynamic Behavior of the TT Elements

In order to carry out the control from the tensor elements, we have to obtain the dynamic system that relates the change in the tensor elements exerted by a change in the velocities of the robot. The dynamic system involves the robot model and is obtained by finding the time-derivatives of the tensor elements in (3) (Appendix A). In practice, the 1D TT has an unknown scale factor that varies as the robot moves. We can set a common scale during the navigation by normalizing each element of the tensor as follows

$$T_{ijk} = \frac{T_{ijk}^m}{T_N^m} \tag{6}$$

where T_{ijk}^m are the TT elements computed from point matches, T_{ijk} are the normalized elements and T_N^m is a suitable normalizing factor. Thus, the normalized dynamic system is the following

$$\begin{aligned}
\dot{T}_{111} &= \tfrac{s\phi_1}{T_N^m}\upsilon + T_{121}\omega, & \dot{T}_{211} &= \tfrac{c\phi_1}{T_N^m}\upsilon + T_{221}\omega,\\
\dot{T}_{112} &= -\tfrac{c\phi_1}{T_N^m}\upsilon + T_{122}\omega, & \dot{T}_{212} &= \tfrac{s\phi_1}{T_N^m}\upsilon + T_{222}\omega,\\
\dot{T}_{121} &= -T_{111}\omega, & \dot{T}_{221} &= -T_{211}\omega,\\
\dot{T}_{122} &= -T_{112}\omega, & \dot{T}_{222} &= -T_{212}\omega.
\end{aligned} \tag{7}$$

It is worth noting that in (7) there are four elements that are independent on the translational velocity (T_{121}, T_{122}, T_{221} and T_{222}). It means that a change in υ does not produce a variation in these tensor elements and consequently, only orientation correction can be performed using such elements. Additionally, the normalizing factor is a kind of gain for the translational velocity and such factor must not be zero. We can see from (3) that $T_{121} \neq 0$ if the initial robot location has nonzero longitudinal translation ($t_{y_1} \neq 0$) and $|\phi_2| < \pi/2$. So, the terms $\cos\phi_2 \to 1$ and $t_{x_2} \to 0$, and consequently, $T_{121} \to t_{y_1}$ as the robot reaches the target. Therefore, a good option for the normalizing factor is $T_N^m = T_{121}^m$.

2.5 Selecting Suited Outputs

The problem of taking three variables to desired values $(t_{x_2}, t_{y_2}, \sin\phi_2) = (0, 0, 0)$ may be completely solved with at least three outputs being controlled. However, it is also possible to find two outputs to take two variables to their desired values and then a third one is left as a DOF to be corrected a posteriori. We propose to use only two outputs, because defining more than two generates a non-square dynamic system, in which its non-invertibility makes difficult to prove stability of the control system.

Under the definition of a global frame in the target view, we can define the longitudinal error as the y coordinate and the lateral error as the x robot position. By taking into account three premises: 1) the values of the tensor elements in the final location, 2) the solution of the homogeneous linear system generated when the outputs are equal to zero, 3) the invertibility of the matrix relating the output dynamics with the inputs, we can state:

- It is possible to design a square control system which can correct orientation and longitudinal error. However, it leaves the lateral error as a DOF. This error cannot be corrected later considering the nonholonomic constraint of the robot.

Thus, this case does not have practical interest.

- It is not possible to design a square control system which allows us to correct orientation and lateral error, leaving the longitudinal error as a DOF.
- It is feasible to design a square control system which can correct both longitudinal and lateral error, leaving the orientation as a DOF. The orientation error can be corrected in a second step considering that the robot uses a differential drive. We concentrate in exploiting this option.

3 1D TT-Based Controller Design

We present the development of a two-step control law, which firstly drives the robot to a desired position and then corrects its orientation. The first step is based on solving a tracking problem for a nonlinear system in order to correct x and y positions. The second step uses direct feedback from one element of the tensor to correct orientation.

3.1 First-Step Controller - Position Correction

The initial location of the robot is (x_1, y_1, ϕ_1), the target location $(x_3, y_3, \phi_3) = (0, 0, 0)$ and the current location (x_2, y_2, ϕ_2), which varies as the robot moves. The goal is to drive the robot to the target location, i.e., to reach $(x_2, y_2, \phi_2) = (0, 0, 0)$. Now we define the control objective as a function of the 1D TT elements. When the robot reaches the target, it achieves the condition given in (5) and therefore, the following sum of normalized tensor elements are selected as outputs

$$\begin{aligned} \xi_1 &= T_{112} + T_{121}, \\ \xi_2 &= T_{212} + T_{221}. \end{aligned} \tag{8}$$

We can see that these outputs go to zero as the robot moves to the target. When $\xi_1 = 0$ and $\xi_2 = 0$ the following homogeneous linear system is given

$$\begin{bmatrix} T_{112} + T_{121} \\ T_{212} + T_{221} \end{bmatrix} = \left[\begin{array}{c|c} s\phi_1 & c\phi_1 \\ \hline c\phi_1 & -s\phi_1 \end{array}\right] \begin{bmatrix} t_{x_2} \\ t_{y_2} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}.$$

This system has unique solution $t_{x_2} = 0$, $t_{y_2} = 0$ for any value of ϕ_1 $(\det(\cdot) = -1)$. Thus, $(t_{x_2}, t_{y_2}, \sin\phi_2) = (0, 0, \sin\phi_2)$ is accomplished, which ensures position correction ($x_2 = 0$, $y_2 = 0$). A robust tracking controller is proposed to take the value of both outputs to zero in a smooth way. Let us define the tracking errors as $e_1 = \xi_1 - \xi_1^d$ and $e_2 = \xi_2 - \xi_2^d$. Thus, the error system is given as

$$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \end{bmatrix} = \left[\begin{array}{c|c} -\frac{c\phi_1}{T_N^m} & T_{122} - T_{111} \\ \hline -\frac{s\phi_1}{T_N^m} & T_{222} - T_{211} \end{array}\right] \begin{bmatrix} \upsilon \\ \omega \end{bmatrix} - \begin{bmatrix} \dot{\xi}_1^d \\ \dot{\xi}_2^d \end{bmatrix}. \tag{9}$$

This system has the form $\dot{\mathbf{e}} = \mathbf{M}(\mathbf{T}, \phi_1)\, \mathbf{u} - \dot{\xi}^d$, where $\mathbf{M}(\mathbf{T}, \phi_1)$ corresponds to the decoupling matrix and $\dot{\xi}^d$ represents a known disturbance. We need to invert the system in order to assign the desired dynamics using the inverse matrix

$$\mathbf{M}^{-1}(\mathbf{T}, \phi_1) = \tfrac{1}{\det(\mathbf{M})} \left[\begin{array}{c|c} T_{222} - T_{211} & T_{111} - T_{122} \\ \hline \frac{s\phi_1}{T_N^m} & -\frac{c\phi_1}{T_N^m} \end{array}\right] \tag{10}$$

where $\det(\mathbf{M}) = \frac{1}{T_N^m}\left[(T_{122} - T_{111})\, s\phi_1 + (T_{211} - T_{222})\, c\phi_1\right]$ and $T_N^m = T_{121}^m$. At the final location $T_{221} = -\alpha t_{x_1}$, $T_{212} = \alpha t_{x_1}$, $T_{121} = \alpha t_{y_1}$, $T_{112} = -\alpha t_{y_1}$, where α is an unknown scale factor, and the other tensor elements are zero. The proposed normalizing factor is never zero in our framework as described in Section 2.4; however, $\det(\mathbf{M}) = 0$ at the final location. This entails the problem that the rotational velocity (ω) increases to infinite as the robot reaches the target. We face this problem by switching to a bounded control law as described later.

We treat the tracking problem as the stabilization of the error system in (9). We propose a robust control law to solve the tracking problem using sliding mode control [28], which has been already applied in visual control [4]. A common way to define sliding surfaces in an error system is to take directly the errors as sliding surfaces, in such a way that, if there exist switched feedback gains that make the states to evolve in $\mathbf{s} = 0$, then the tracking problem is solved.

$$\mathbf{s} = \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} = \begin{bmatrix} e_1 \\ e_2 \end{bmatrix} = \begin{bmatrix} \xi_1 - \xi_1^d \\ \xi_2 - \xi_2^d \end{bmatrix}. \tag{11}$$

We use these sliding surfaces and the *equivalent control method* in order to find switched feedback gains to drive the state trajectory to $\mathbf{s} = 0$ and maintaining it there for future time. From the equation $\dot{\mathbf{s}} = 0$, the so-called equivalent control is

$$\mathbf{u}_{eq} = \mathbf{M}^{-1}\dot{\xi}^d. \tag{12}$$

A control law that ensures global stabilization of the error system has the form $\mathbf{u}_{sm} = \mathbf{u}_{eq} + \mathbf{u}_{disc}$, where $\mathbf{u}_{disc}$ is a two-dimensional vector containing switched feedback gains. We propose these gains as follows

$$\mathbf{u}_{disc} = \mathbf{M}^{-1} \begin{bmatrix} -k_1^{sm} sign(s_1) \\ -k_2^{sm} sign(s_2) \end{bmatrix} \tag{13}$$

where $k_1^{sm} > 0$ and $k_2^{sm} > 0$ are control gains. Although $\mathbf{u}_{sm}$ can achieve global stabilization of the error system, high gains may be needed, which can cause undesirable effects in real situations. We add a pole placement term in the control law to alleviate this issue

$$\mathbf{u}_{pp} = \mathbf{M}^{-1} \left[\begin{array}{c|c} -k_1 & 0 \\ \hline 0 & -k_2 \end{array} \right] \begin{bmatrix} s_1 \\ s_2 \end{bmatrix} \tag{14}$$

where $k_1 > 0$ and $k_2 > 0$ are control gains. Finally, a decoupling-based control law that achieves robust global stabilization of the system (9) is as follows

$$\mathbf{u}_{db} = \begin{bmatrix} \upsilon_{db} \\ \omega_{db} \end{bmatrix} = \mathbf{u}_{eq} + \mathbf{u}_{disc} + \mathbf{u}_{pp} = \mathbf{M}^{-1} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \tag{15}$$

where $u_1 = \dot{\xi}_1^d - k_1^{sm} sign\left(s_1\right) - k_1 s_1$, and $u_2 = \dot{\xi}_2^d - k_2^{sm} sign\left(s_2\right) - k_2 s_2$. Note that this control law depends on the orientation of the fixed auxiliary camera ϕ_1. This orientation has to be computed only in the initial location and can be obtained from the epipoles that relate the initial and target images. Any uncertainty in the estimation of the initial orientation can be overcome given the robustness properties of our control law, which justify the application of sliding mode control. Moreover, ϕ_1 can be fixed to zero as shown in Table 1 of Section 5.1.

3.1.1 Solving the Singularity

We use the inverse of the decoupling matrix (10) to compute the control inputs, which causes a singularity problem at the final condition. The singularity affects the computation of both velocities, however υ tends to zero as the robot reaches the target. To keep ω bounded and the outputs tracking their references, we propose the commutation to a direct sliding mode controller when $\det(\mathbf{M})$ is near to zero. This kind of controller has been studied for output tracking through singularities [29], and has been previously applied in the context of visual servoing [4]. For this case, a bounded sliding mode controller is as follows

$$\mathbf{u}_b = \begin{bmatrix} \upsilon_b \\ \omega_b \end{bmatrix} = \begin{bmatrix} M \; sign\left(s_1\right) \\ -N \; sign(s_2 \; g(\mathbf{T})) \end{bmatrix} \tag{16}$$

where M and N are suitable gains, and $g(\mathbf{T})$ will be defined through the stability analysis in Section 4, and it is found by achieving the negativeness of a Lyapunov function derivative. The control law in (16) locally stabilizes the system (9) and is always bounded.

3.1.2 Desired Trajectories

The goal of the reference tracking is to take the outputs to zero in a smooth way in such a way that the robot performs a smooth motion in a desired time. We propose the following references

$$
\begin{aligned}
\xi_1^d &= \tfrac{T_{112}^{ini}+T_{121}^{ini}}{2}\left(1+\cos\left(\frac{\pi}{\tau}t\right)\right), 0 \le t \le \tau \\
\xi_1^d &= 0, \qquad\qquad\qquad\qquad\quad t > \tau \\
\xi_2^d &= \tfrac{T_{212}^{ini}+T_{221}^{ini}}{2}\left(1+\cos\left(\frac{\pi}{\tau}t\right)\right), 0 \le t \le \tau \\
\xi_2^d &= 0, \qquad\qquad\qquad\qquad\quad t > \tau
\end{aligned}
\tag{17}
$$

where τ is the time to reach the target and $T_*^{ini} = T_*(t = 0)$. The choice of these trajectories obeys just to the requirement of a smooth zeroing of the outputs along a fix temporal horizon. Indeed, a parabolic function may be used without difference in the resulting behavior. By defining τ, we fix the duration of the first part of the control and the time to switch to correct orientation. Note that, although initially the current image is the same than the starting one, there is enough information in the 1D TT (4) to have well defined references.

3.2 Second-Step Controller - Orientation Correction

Once position correction has been reached in $t = \tau$, we can use any single tensor element whose dynamics depends on ω and with desired final value zero to correct orientation. We select the dynamics $\dot{T}_{122} = -T_{112}\omega$. A suitable input ω that yields T_{122} exponentially stable is

$$\omega = k_\omega \tfrac{T_{122}}{T_{112}}, \qquad t > \tau \tag{18}$$

where $k_\omega > 0$ is a control gain. This rotational velocity assigns the following dynamics to T_{122}, which is clearly exponentially stable

$$\dot{T}_{122} = -T_{112}\left(k_\omega \tfrac{T_{122}}{T_{112}}\right) = -k_\omega T_{122}. \tag{19}$$

Note that (18) never becomes singular because $T_{112} = -t_{y_1}\cos\phi_2$ for $t = \tau$ and it tends to $-t_{y_1} \neq 0$ as final value. Although only a rotation is carried out in this second step, we keep the translational velocity υ_b given in (16) in order to have closed loop control along the whole motion.

4 Stability Analysis

The control action in the first step is based on zeroing the defined outputs. So, when these outputs reach zero, the so-called *zero dynamics* in the robot system is achieved. Zero dynamics is described by a subset of the state space which makes the output to be identically zero [30]. In the particular case of the robot system (1) with output vector (8), this set is given as follows

$$Z^* = \left\{ (x_2, y_2, \phi_2)^T \mid \xi_1 \equiv 0,\ \xi_2 \equiv 0 \right\} = \left\{ (0, 0, \phi_2)^T,\ \phi_2 \in \mathbb{R} \right\}.$$

Zero dynamics in this control system means that, when the chosen outputs are zero, the x and y-coordinates of the robot are corrected, but orientation may be different to zero. This zero dynamics yields $T_{122} = t_{y_1} \sin \phi_2$ and, therefore, when we make $T_{122} = 0$ then $\phi_2 = n\pi$ with $n \in \mathbb{Z}$, and the orientation is corrected. It is clear the exponential stability of T_{122} in the second step (19) for any $k_\omega > 0$, and we focus on proving stability for the tracking control law.

Proposition 1. Global stabilization of the system in (9) is achieved with a commuted control law applied for $t \leq \tau$, which starts with the decoupling-based control in (15) and switches to the bounded control in (16) if

$$\left|\det \left(\mathbf{M} \left(\mathbf{T}, \phi_1 \right) \right)\right| < T_h \tag{20}$$

where T_h is a suitable threshold value.

Proof. As mentioned above, the switching between the decoupling-based control to the bounded one happens only when the robot is near to the target location. For a sliding mode controller we have to prove the existence of sliding modes. This means to develop a stability proof to know if the sliding surfaces can be reached in a finite time and the state trajectory can be maintained there. Let us use the natural Lyapunov function for a sliding mode controller

$$V = V_1 + V_2,\ V_1 = \tfrac{1}{2} s_1^2,\ V_2 = \tfrac{1}{2} s_2^2 \tag{21}$$

which accomplishes $V(s_1 = 0, s_2 = 0) = 0$ and $V > 0$ for all $s_1 \neq 0, s_2 \neq 0$.

$$\dot{V} = \dot{V}_1 + \dot{V}_2 = s_1 \dot{s}_1 + s_2 \dot{s}_2. \tag{22}$$

Now, we analyze each term of (22) for the decoupling based controller (15). After some simple mathematical simplifications we have

$$\begin{aligned}
\dot{V}_1 &= s_1 \left(u_1 - \dot{\xi}_1^d \right) = s_1 \left(\dot{\xi}_1^d - k_1^{sm} sign \left(s_1 \right) - k_1 s_1 - \dot{\xi}_1^d \right) = -k_1^{sm} \left| s_1 \right| - k_1 s_1^2, \\
\dot{V}_2 &= s_2 \left(u_2 - \dot{\xi}_2^d \right) = s_2 \left(\dot{\xi}_2^d - k_2^{sm} sign \left(s_2 \right) - k_2 s_2 - \dot{\xi}_2^d \right) = -k_2^{sm} \left| s_2 \right| - k_2 s_2^2.
\end{aligned}$$

$\dot{V}_1$ and $\dot{V}_2$ are negative definite iff the following inequalities are guaranteed for all $s_1 \neq 0, s_2 \neq 0$.

$$k_1^{sm} > 0, k_1 \geq 0, k_2^{sm} > 0, k_2 \geq 0. \tag{23}$$

Therefore, $\dot{V} < 0$ iff both inequalities in (23) are fulfilled. So, global convergence to the sliding surfaces is achieved.

Now, let us develop the existence conditions of sliding modes for the bounded controller (16). The same Lyapunov function in (21) is used, and for each term of (22) we have

$$\dot{V}_1 = -\tfrac{M\cos\phi_1}{T_N^m}\,|s_1| + s_1\left((T_{122}-T_{111})\,(-N\ sign(s_2\,g(\mathbf{T}))) - \dot{\xi}_1^d\right),$$
$$\dot{V}_2 = s_2\left(-\tfrac{M\sin\phi_1}{T_N^m}\ sign\,(s_1) - \dot{\xi}_2^d\right) - N\ |s_2|\ \ (T_{222}-T_{211})\,sign\,(g(\mathbf{T}))\,.$$

Let us define $A = -N\,(T_{122}-T_{111})\ sign(s_2\ g(\mathbf{T})) - \dot{\xi}_1^d$ and $B = -\frac{M\sin\phi_1}{T_N^m}$ $sign\,(s_1) - \dot{\xi}_2^d$. In order to enforce negativeness of $\dot{V}_2$ for some value of N, the function $g(\mathbf{T})$ has to be $g(\mathbf{T}) = T_{222} - T_{211}$. Hence, we have

$$\dot{V}_1 = -\tfrac{M\cos\phi_1}{T_N^m}\,|s_1| + s_1 A,\, \dot{V}_2 = -N\ |s_2|\ \ |T_{222}-T_{211}| + s_2 B.$$

We can see that

$$\dot{V}_1 \leq -\left(\tfrac{M\cos\phi_1}{T_N^m} - |A|\right)|s_1|\,, \dot{V}_2 \leq -\left(N\ |T_{222}-T_{211}| - |B|\right)|s_2|\,.$$

$\dot{V}_1$ and $\dot{V}_2$ are negative definite iff the following inequalities are assured for all $s_1 \neq 0$, $s_2 \neq 0$.

$$M > \tfrac{T_N^m |A|}{\cos\phi_1},\, N > \tfrac{|B|}{|T_{222}-T_{211}|}. \tag{24}$$

Therefore, $\dot{V} < 0$ iff both inequalities in (24) are fulfilled. The bounded controller does not need any information of system parameters and thus, its robustness is implicit.

According to the existence conditions of sliding modes, the bounded controller (16) is able to locally stabilize the system (9). Its attraction region is bigger as long as the control gains M and N are higher. Because of the bounded control law is also a switching one, the commutation from the decoupling-based control to the bounded one does not affect the stability of the closed loop system. The first controller ensures entering to the attraction region of the second one. Once the sliding surfaces are reached for any case of control law, the system's behavior is independent of matched uncertainties and disturbances [28]. Uncertainties in the system (9) due to ϕ_1 fulfill the so-called *matching condition*, and as a result, robustness of the control system is accomplished.

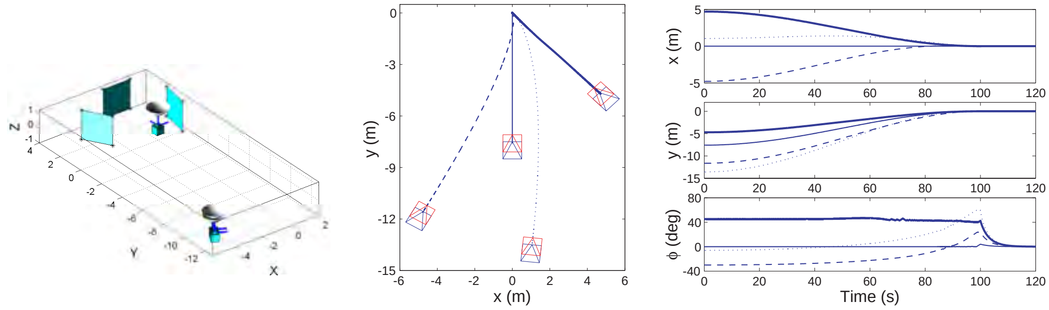
5 Experimental Evaluation

5.1 Simulation Results

In this section, we present some simulations of the overall control system as established in the Proposition 1 for the first step, and using ω (18) and υ_b (16) for the second one. Simulations have been performed in Matlab. The results show that the main objective of driving the robot to a desired pose ((0,0,0^o) in all the cases) is attained just from image measurements and even with noise in the images. The 1D TT is estimated from more than five point correspondences in virtual omnidirectional images of size 1024×768. These images have been generated from a 3D scene (Fig. 4(a)) through the generic model for catadioptric central cameras [22]. We report results with hypercatadioptric, paracatadioptric and also fisheye cameras, which can be approximately represented with the same model [31]. Besides, the computation of the 1D TT has been studied for fisheye cameras in [16], which supports the claim that our approach is robust to small deviations of the central camera configuration. It is worth noting that, although analytically we can deduce values of the tensor elements by substituting in (3) the relative location between cameras, in practice, it is troublesome when the image coordinates of two images are exactly the same. It causes that the linear estimation of the trifocal constraint degenerates for such condition. We avoid this issue by moving the robot forward for a short time before to start the control. When the robot reaches the target, there is always a minimum error between image coordinates that is enough to prevent numeric problems to solve for the 1D TT even in simulations. Without loss of generality, the projection center is zero for all the simulations. For the controllers, the time to reach the target position τ is fixed to 100 s, the threshold to switch to the bounded control T_h (20) is fixed to 0.04, and the control gains are set to $k_1 = 1$, $k_2 = 2$, $k_1^{sm} = 0.02$, $k_2^{sm} = 0.02$, $k_\omega = 0.3$, $M = 0.1$, $N = 0.05$.

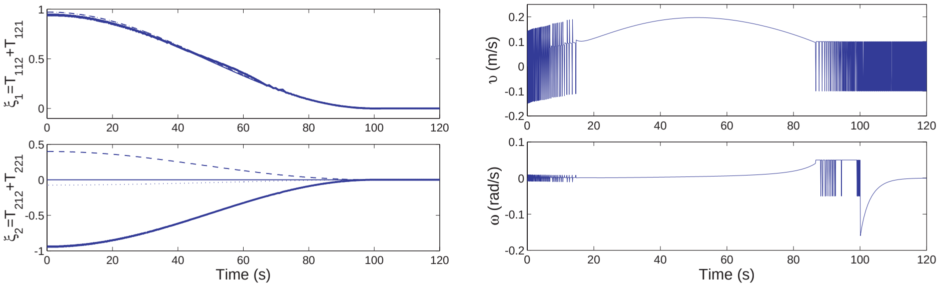
Fig. 4 shows the paths traced by the robot and the state variables evolution from four different initial locations. The thick solid line starts from (5,-5,45^o), the long dashed line from (-5,-12,-30^o), the solid line from (0,-8,0^o), and the short dashed line from (1,-14,-6^o). In the paths of Fig. 4(b) we can differentiate between three kind of autonomously performed robot motion. The solid lines correspond to a rectilinear motion to the target, while the long dashed line and the short dashed line both describe an inner curve and an outer curve before reaching the target respectively. The rectilinear motion is obtained when the initial rotation is such that $t_{x_1} = t_{x_2} = 0$, which implies that the robot is pointing toward the target. The inner curve is generated when the initial rotation is such that $t_{x_1} = t_{x_2} > 0$ and the outer curve when the initial rotation is such that $t_{x_1} = t_{x_2} < 0$. In both later cases the robot rotation increases autonomously, and it is efficiently corrected in the second step after 100 s, as shown in Fig. 4(c).

We can see in Fig. 5(a) that both outputs are driven to zero in 100 s for all the cases. This is achieved by using bounded inputs, which are presented in Fig. 5(b) for the case (-5,-12,-30^o). Both control inputs commute to a bounded value around 86 seconds because the determinant of the decoupling matrix falls under the fixed threshold. We can also see how the rotational velocity presents an exponential decay after 100 s, which takes the element T_{122} to zero as can be seen in Fig. 6. This forces the orientation to decrease with a fixed exponential rate, whose settling time is approximately 16.7 s ($5/k_\omega$). This time or a threshold for T_{122} may be used to stop both of the control inputs and finish the task.



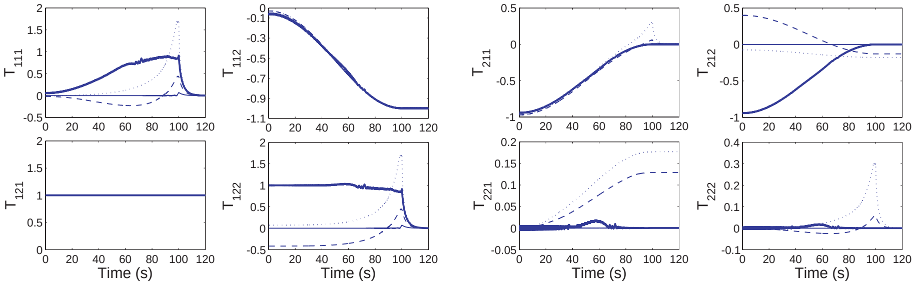
(a) (b) (c)

Fig. 4. Simulation results with synthetic images. (a) 3D scene. (b) Paths on the plane. (c) State variables of the robot.



(a) (b)

Fig. 5. Control law performance. (a) Controlled outputs for the four cases of Fig. 4. (b) Example of the computed velocities for initial location (-5,-12,-30^o).

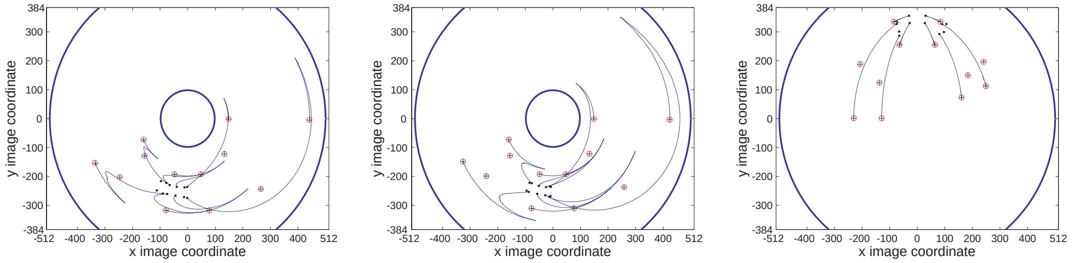


(a) (b)

Fig. 6. Tensor elements evolution for the four cases of Fig. 4. (a) Behavior of the elements of $\mathbf{T}_1$. (b) Behavior of the elements of $\mathbf{T}_2$.

The previous results have been obtained for three different kind of omnidirectional cameras. Fig. 7(a) shows the motion of the image points for the case (-5,-12,-30^o), in which a hypercatadioptric camera is simulated. Fig. 7(b) corresponds to the case (1,-14,-6^o) with a paracatadioptric camera and Fig. 7(c) is a fisheye camera for the initial location (0,-8,0^o).

Table 1 shows that the target location is reached with good accuracy. The results in the first part of the table are obtained considering that the initial orientation ϕ_1



(a) (b) (c)

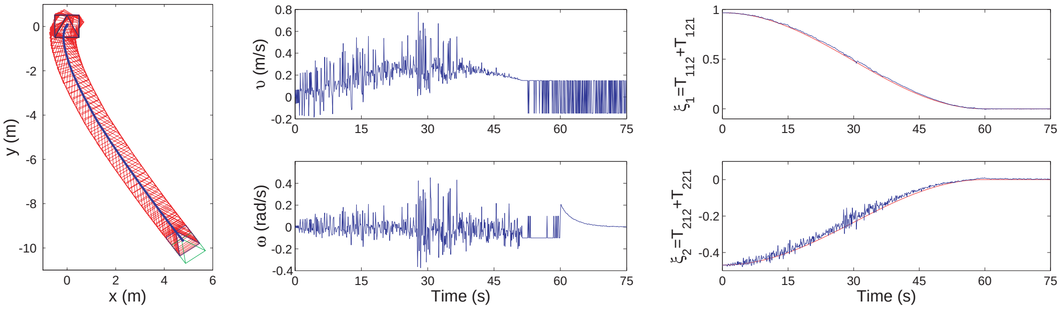
Fig. 7. Motion of the points in the image plane for three different kind of omnidirectional virtual images. (a) Hypercatadioptric. (b) Paracatadioptric. (c) Fisheye. The images depict the point features from the initial, current and target views.

Table 1
Final location for the paths in Fig. 4.

	(5,-5,45) (m,m,^{o})	(-5,-12,-30) (m,m,^{o})	(0,-8,0) (m,m,^{o})	(1,-14,-6) (m,m,^{o})
Final locations considering the initial orientation ϕ_1 as known.				
x (cm)	-0.28	0.85	0	0.91
y (cm)	0.59	0.71	0.11	-0.47
ϕ (^{o})	0.10	0.02	0	0.08
Final locations fixing $\phi_1 = 0$ in the controller.				
x (cm)	-0.51	0.77	0	0.98
y (cm)	0.86	0.39	0.11	-0.25
ϕ (^{o})	0.11	0.01	0	0.07

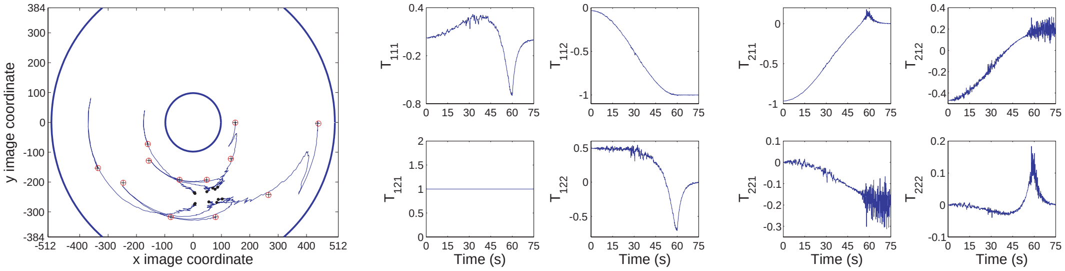
is known for each case. On the other hand, the second part of the table shows that the precision is preserved even if the initial orientation is fixed to $\phi_1 = 0$ in the controller for all the cases. We can assert that similar accuracy is obtained by fixing ϕ_1 in the range $-30 \leq \phi_1 \leq 30$, since that the sliding mode control law is robust to parametric uncertainty. For all the experiments, the mean squared tracking error is very low, in the order of 1×10^{-5}.

Fig. 8(a) shows the good performance of the approach under image noise for initial pose (5,-10,35^o). The added noise has a standard deviation of 1 pixel and the time to reach the target (τ) is set to 60 s. The control inputs are affected directly by the noise, as can be seen in Fig. 8(b). Nevertheless, the outputs are regulated properly to the desired reference as shown in Fig. 8(c). The presence of the noise can be observed in the image points motion of Fig. 9(a), which results in the behavior of the tensor elements that is presented in Fig. 9(b)-(c).



(a) (b) (c)

Fig. 8. Control law performance with image noise. (a) Resultant robot path. (b) Control inputs. (c) Controlled outputs.



(a) (b) (c)

Fig. 9. Visual measurements from synthetic images with image noise. (a) Motion of the image points. (b) Behavior of $\mathbf{T}_1$. (c) Behavior of $\mathbf{T}_2$.

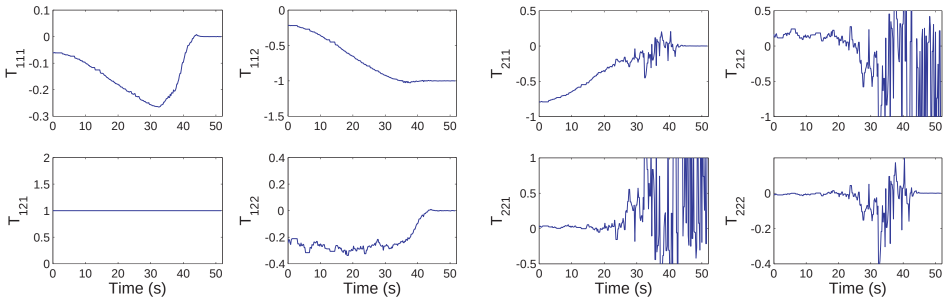
5.2 Experiments with Real Data

This section describes an analysis of the behavior of the proposed control scheme through experiments with omnidirectional images. Two techniques of extracting the required features are employed. The first case corresponds to the use of the well known SIFT features [27] and, in the second case, we use the Lucas-Kanade pyramidal algorithm [32], [33]. These experiments are performed off-line, which means that a sequence of images were taken and then used to compute the 1D TT and the control inputs to analyze the effect of the feature extraction. This analysis is a key factor toward the real world experimentation of the next section. Sliding mode control requires a relatively high closed loop frequency, around 10 Hz as minimum, and consequently, the computational cost of the feature extraction and the matching process becomes very important. For these experiments, we use an omnidirectional system with a camera Sony XCD-X7101CR and a mirror Neovision H3S (Fig. 1(c)) to capture images of size 1024×768. The image data is acquired using the free software tool Player. The commanded robot motion is a slight curve going forward and finishing with a rotation.

An important parameter required to obtain the bearing measurements is the projection center. We have tested the singleness of this point in our imaging system by estimating the projection center along a sequence. Like in [34], the center is robustly estimated using a RANSAC approach from 3D vertical lines, which project in radial lines for central imaging systems. Results have shown that our imaging system properly approximates a single view point configuration, with standard deviation of around 1 pixel for each image coordinate of the estimated center. For the size of images that we are using, these deviations have a negligible effect in the computation of bearing measurements and thus, we have fixed the projection center to ($x_0 = 541$, $y_0 = 405$).

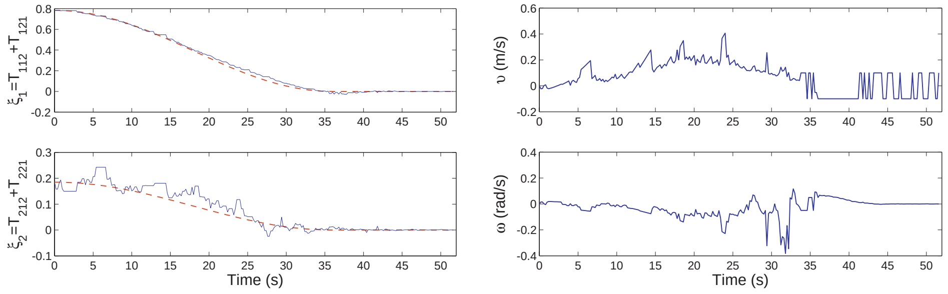
5.2.1 Behavior using SIFT Features

We have implemented a 1D TT estimation algorithm by solving the trifocal constraint for at least five points that are extracted using SIFT and robustly matched using RANSAC. The five-point method reduces the number of iterations required for the robust estimation, however, the computation time of the 1D TT with this method is still very high (approximately 5 seconds per iteration). Moreover, as can be seen in Fig. 10, the 1D TT estimation is very unstable even having correct matches. It happens because in some cases the matches are concentrated in a region of the image. Besides, due to the property of SIFT features of being a region in the image, the effective coordinates of the features may change discontinuously along the sequence. We can see that the elements of T_2 are the most unstable, in particular when the current image is close to the target image (around 35 seconds), however, after this time the first control step is finishing and the noisy elements are not used any more. Fig. 11(a) shows how the reference trajectory for the first output is well approximated while output two is not close to its reference. Fig. 11(b) presents the computed control inputs. The translational velocity approximately describes the forward motion; however, the rotational velocity is very noisy.



(a) (b)

Fig. 10. Performance of the 1D TT estimation using SIFT features. (a) Behavior of the elements of $\mathbf{T}_1$. (b) Behavior of the elements of $\mathbf{T}_2$.



(a) (b)

Fig. 11. Behavior of the control law using SIFT features. (a) Outputs and their references. (b) Computed velocities.

5.2.2 Behavior using Tracking of Features

In order to achieve an adequate closed loop frequency, we evaluate the strategy of tracking a set of chosen points using the Lucas-Kanade algorithm [32], [33]. The tracking of features has been extensively applied for visual servoing purposes [35]. It allows us to have the matching between features for each iteration without additional computations, which makes the scheme feasible for real world experimentation. Additionally, the smooth motion of the image features with the Lucas-Kanade tracker results in a stable tensor estimation. We have defined 12 point features to be tracked along the same image sequence and then, the corresponding point coordinates are used to estimate the 1D TT and the velocities as given for our control law. Fig. 12 displays some of these tracked points and their motion in the image. The resulting behavior of the TT elements (Fig. 13) shows that they are more stable than in the case of SIFT features, however, a similar behavior is obtained at the end for the elements T_{212} and T_{221}. According to Fig. 14 both of the outputs are close to their reference trajectories, and consequently, the computed velocities in Fig. 14(b) actually describe the real motion of the camera.

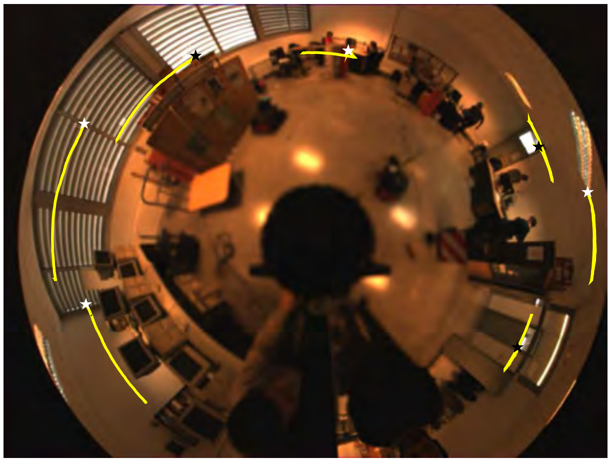
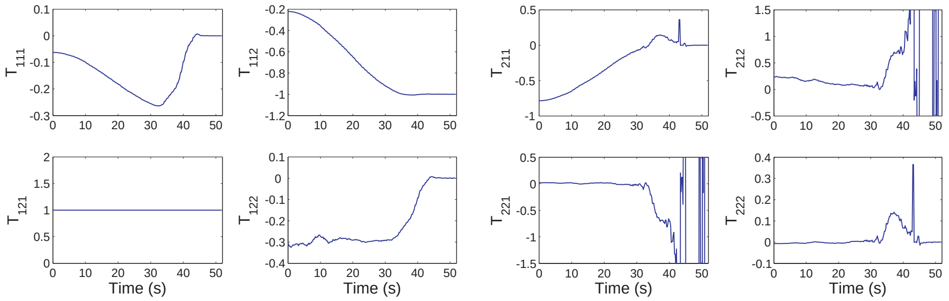
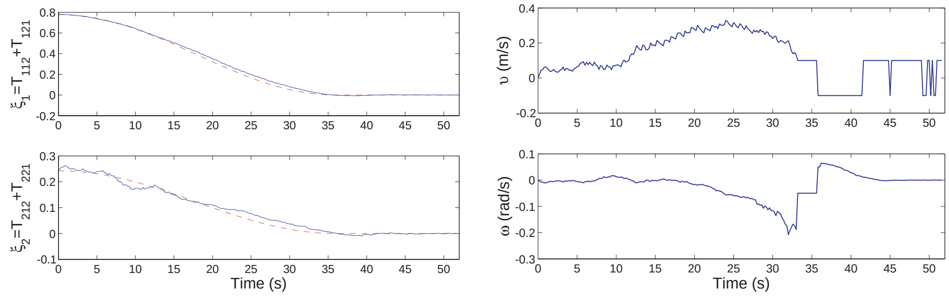


Fig. 12. Some of the tracked points (stars) and their motion in the image along a sequence.



(a) (b)

Fig. 13. Performance of the 1D TT estimation using tracking of point features. (a) Behavior of the elements of $\mathbf{T}_1$. (b) Behavior of the elements of $\mathbf{T}_2$.

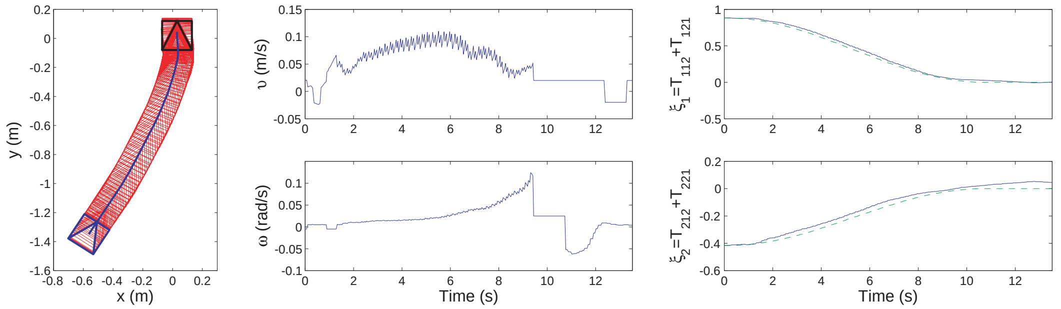


(a) (b)

Fig. 14. Behavior of the control law using tracking of point features. (a) Outputs and their references. (b) Computed velocities.

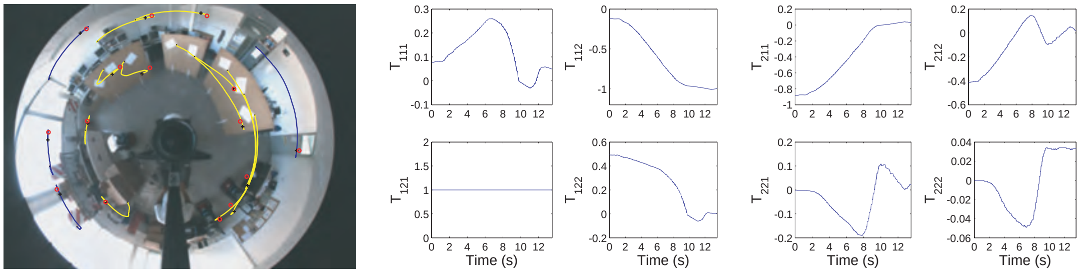
5.3 Real World Experiments

The proposed approach has been tested in closed loop with real conditions using the Pioneer 3-AT robot that is shown in Fig. 1(a). The same hypercatadioptric imaging system of the previous section is used, but now the images are acquired at a size of 800×600 pixels. The projection center has been fixed according to a calibration process to ($x_0 = 404$, $y_0 = 316$). The observed scene has been set up with features on three different planes in order to ensure a sufficient number of points in the scene. However, points not belonging to these planes are also used to achieve a total of 15 points, which are manually matched in the three initial images. We have implemented these experiments using the tracking of features because its low computational cost, giving good frequency, leads to the good behavior in the 1D TT estimation, as described in the previous section. Fig. 15(a) presents the resultant path, given by odometry, of the closed loop control from the initial location (-0.55 m,-1.35 m,-35^o) for one of the experimental runs. The duration of the task is almost 14 s, the final longitudinal error is around 2 cm, the lateral error around 3 cm and the orientation error is practically negligible. The time τ for the execution of the first step is set to 9.4 s through fixing a number of iterations in our control software. Before that, we can see in Fig. 15(b) that the bounded sliding mode control law is applied due to the singularity of the decoupling-based controller. Fig. 15(c) shows that the behavior of the outputs is always close to the desired one but with a small error. The reason of the remaining error is that our robotic platform is not able to execute commands at a frequency higher than 10 Hz, and consequently the performance of the sliding mode control is not the optimum. According to Fig. 16(a) the motion of the image points along the sequence does not exhibit a damaging noise, in such a way that the tensor elements evolve smoothly during the task, as presented in Fig. 16(b)-(c). Fig. 17 shows a sequence of some images taken by the robot camera.



(a) (b) (c)

Fig. 15. Experimental results with the control law in closed loop. (a) Resultant path. (b) Computed velocities. (c) Controlled outputs. The data to plot the path is given by the robot odometry.



(a) (b) (c)

Fig. 16. Behavior of the visual measurements for the real experiments. (a) Motion of the image points. (b) Evolution of $\mathbf{T}_1$. (c) Evolution of $\mathbf{T}_2$.

In accordance to the results and the methodology presented along the paper, we can state that the main advantages of using the 1D TT on visual servoing are that the geometric constraint improves the robustness to image noise by filtering the data, allows applying the control approach with any visual sensor obeying approximately a central projection model and avoids the problem of short baseline by exploiting the information of three views. Thus, total correction of both position and orientation is ensured without switching to any visual constraint other than the 1D trifocal tensor. Since we assume planar motion and the camera is placed looking upward, the use of the 1D TT particularly adapts to the central omnidirectional case. Due to the property of this imaging process of preserving bearing information, we have

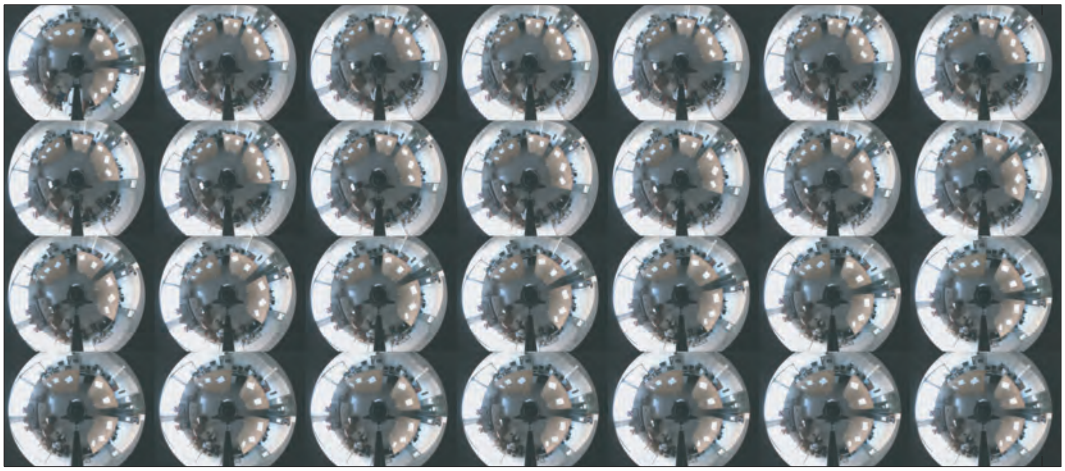


Fig. 17. Sequence of some of the omnidirectional images taken from the hypercatadioptric robot camera during the real experiments. The first is the target image, the second is the initial and the last is the image at the end of the motion.

achieved an algorithm that is independent of the radial distortion induced by lenses and mirrors.

From a control theory point of view, an additional advantage of our approach with respect to the basic IBVS schemes is that the selected outputs allow us to prove stability on the basis of a square control system. Additionally, we have incorporated robustness properties to the closed loop by using sliding mode control. The lack of clear stability properties and robustness has been a serious concern in IBVS approaches [1]. However, the cost to prove stability is to be limited to the application of our approach on differential drive robots, for which the final remaining orientation can be corrected.

6 Conclusions

Along this paper we have presented a novel image-based approach to perform visual control for mobile robots using the elements of the 1D trifocal tensor (TT) directly in the control law. It allows exploiting the properties of omnidirectional images of preserving bearing information. In this sense, our approach is valid for any visual sensor obeying approximately a central projection model. To the authors' knowledge, this is the first application of the 1D TT for visual servoing. The visual control utilizes the usual teach-by showing strategy without requiring any a priori knowledge of the scene and does not need any auxiliary image. The proposed two-step control law ensures total correction of position and orientation without need of switching to any visual constraint other than the 1D TT. In the first step, we correct position by solving a tracking problem for a non-linear square system using sliding mode control. This provides stability and robustness properties to the closed loop. In the second step, a single tensor element is used to perform orientation correction. The effectiveness of our approach is tested through simulations and real world experiments with a hypercatadioptric camera.

7 Acknowledgment

This work was supported by project DPI 2009-08126 and grants of Banco Santander-Universidad de Zaragoza and CONACYT-México.

A Derivation of the Dynamic Behavior of the 1D TT Elements

We show two examples of the procedure to obtain the time-derivatives of the tensor elements presented in (7). The unnormalized tensor is denoted by T_*^m. From (3) and using the rates of change of the state variables in (1) for T_{111} we have

$$\begin{aligned}
T_{111}^m &= t_{y_1} s\phi_2 - \left(x_2 s\phi_2 - y_2 c\phi_2\right) s\phi_1 \\
\dot{T}_{111}^m &= t_{y_1} \dot{\phi}_2 c\phi_2 - \left(\dot{x}_2 s\phi_2 + x_2 \dot{\phi}_2 c\phi_2 - \dot{y}_2 c\phi_2 + y_2 \dot{\phi}_2 s\phi_2\right) s\phi_1 \\
&= t_{y1} \omega c\phi_2 - \left(-\upsilon s\phi_2 s\phi_2 + x_2 \omega c\phi_2 - \upsilon c\phi_2 c\phi_2 + y_2 \omega s\phi_2\right) s\phi_1 \\
&= \upsilon s\phi_1 + \omega \left(t_{y1} c\phi_2 + t_{x2} s\phi_1\right) = \upsilon s\phi_1 + T_{121}^m \omega
\end{aligned}$$

By applying (6) in both sides of the equation, it results in the normalized time-derivative of T_{111}

$$\dot{T}_{111} = \tfrac{s\phi_1}{T_N^m} \upsilon + T_{121} \omega$$

The same procedure is carried out for each element, for instance for T_{121}

$$\begin{aligned}
T_{121}^m &= t_{y_1} c\phi_2 + \left(-x_2 c\phi_2 - y_2 s\phi_2\right) s\phi_1 \\
\dot{T}_{121}^m &= -t_{y_1} \dot{\phi}_2 s\phi_2 + \left(-\dot{x}_2 c\phi_2 + x_2 \dot{\phi}_2 s\phi_2 - \dot{y}_2 s\phi_2 - y_2 \dot{\phi}_2 c\phi_2\right) s\phi_1 \\
&= -t_{y1} \omega s\phi_2 + \left(\upsilon s\phi_2 c\phi_2 + x_2 \omega s\phi_2 - \upsilon c\phi_2 s\phi_2 - y_2 \omega c\phi_2\right) s\phi_1 \\
&= \omega \left(-t_{y1} s\phi_2 + t_{y2} s\phi_1\right) = -T_{111}^m \omega
\end{aligned}$$

By normalizing, the result is $\dot{T}_{121} = -T_{111}\omega$.

References

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

[2] F. Chaumette and S. Hutchinson. Visual servo control, part II: Advanced approaches. *IEEE Robotics and Automation Magazine*, 14(1):109–118, 2007.

[3] G. López-Nicolás, C. Sagüés, J.J. Guerrero, D. Kragic, and P. Jensfelt. Switching visual control based on epipoles for mobile robots. *Robotics and Autonomous Systems*, 56(7):592–603, 2008.

[4] H. M. Becerra and C. Sagues. A sliding mode control law for epipolar visual servoing of differential-drive robots. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3058–3063, 2008.

[5] S. Benhimane and E. Malis. Homography-based 2D visual servoing. In *IEEE International Conference on Robotics and Automation*, pages 2397–2402, 2006.

[6] G. López-Nicolás, C. Sagüés, and J.J. Guerrero. Homography-based visual control of nonholonomic vehicles. In *IEEE International Conference on Robotics and Automation*, pages 1703–1708, 2007.

[7] N. R. Gans and S. A. Hutchinson. A stable vision-based control scheme for nonholonomic vehicles to keep a landmark in the field of view. In *IEEE International Conference on Robotics and Automation*, pages 2196–2200, 2007.

[8] Y. Mezouar, H. H. Abdelkader, P. Martinet, and F. Chaumette. Central catadioptric visual servoing from 3D straight lines. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 343–349, 2004.

[9] H. H. Abdelkader, Y. Mezouar, N. Andreff, and P. Martinet. Image-based control of mobile robot with central catadioptric cameras. In *IEEE International Conference on Robotics and Automation*, pages 3522–3527, 2005.

[10] G. L. Mariottini and D. Prattichizzo. Image-based visual servoing with central catadioptric cameras. *The International Journal of Robotics Research*, 27(1):41–56, 2008.

[11] S. Benhimane and E. Malis. A new approach to vision-based robot control with omni-directional cameras. In *IEEE International Conference on Robotics and Automation*, pages 526–531, 2006.

[12] E. Menegatti, T. Maeda, and H. Ishiguro. Image-based memory for robot navigation using properties of omnidirectional images. *Robotics and Autonomous Systems*, 47(4):251–267, 2004.

[13] A. A. Argyros, K. E. Bekris, S. C. Orphanoudakis, and L. E. Kavraki. Robot homing by exploiting panoramic vision. *Autonomous Robots*, 19(1):7–25, 2005.

[14] R. I. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, second edition, 2004.

[15] F. Dellaert and A. W. Stroupe. Linear 2D localization and mapping for single and multiple robot scenarios. In *IEEE International Conference on Robotics and Automation*, pages 688–694, 2002.

[16] S. Thirthala and M. Pollefeys. The radial trifocal tensor: A tool for calibrating the radial distortion of wide-angle cameras. In *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 321–328, 2005.

[17] S. Thirthala and M. Pollefeys. Trifocal tensor for heterogeneous cameras. In *Proc. of 6th Workshop on Omnidirectional Vision, Camera Networks and Non-classical Cameras (OMNIVIS)*, 2005.

[18] J.J. Guerrero, A.C. Murillo, and C. Sagüés. Localization and matching using the planar trifocal tensor with bearing-only data. *IEEE Transactions on Robotics*, 24(2):494–501, 2008.

[19] A.C. Murillo, C. Sagüés, J.J. Guerrero, T. Goedemé, T. Tuytelaars, and L. Van Gool. From omnidirectional images to hierarchical localization. *Robotics and Autonomous Systems*, 55(5):372–382, 2007.

[20] G. López-Nicolás, J.J. Guerrero, and C. Sagüés. Visual control through the trifocal tensor for nonholonomic robots. *Robotics and Autonomous Systems*, doi:10.1016/j.robot.2009.09.005.

[21] H. M. Becerra and C. Sagues. A novel 1D trifocal-tensor-based control for differential-drive robots. In *IEEE International Conference on Robotics and Automation*, pages 1104–1109, 2009.

[22] C. Geyer and K. Daniilidis. Catadioptric projective geometry. *International Journal of Computer Vision*, 45(3):223–243, 2001.

[23] J. Tardif, Y. Pavlidis, and K. Daniilidis. Monocular visual odometry in urban environments using an omnidirectional camera. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2531–2538, 2008.

[24] D. Scaramuzza, F. Fraundorfer, and R. Siegwart. Real-time monocular visual odometry for on-road vehicles with 1-point RANSAC. In *IEEE International Conference on Robotics and Automation*, pages 4293–4299, 2009.

[25] S. Baker and S. K. Nayar. A theory of single-viewpoint catadioptric image formation. *International Journal of Computer Vision*, 35(2):175–196, 1999.

[26] T. Svoboda and T. Padjla. Epipolar geometry for central catadioptric cameras. *International Journal of Computer Vision*, 49(1):23–37, 2002.

[27] D. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.

[28] V. Utkin, J. Guldner, and J. Shi. *Sliding Mode Control in Electromechanical Systems*. CRC Press, Boca Raton, 1999.

[29] R. M Hirschorn. Output tracking through singularities. In *IEEE Conference on Decision and Control*, pages 3843–3848, 2002.

[30] S. Sastry. *Nonlinear Systems: Analysis, Stability and Control*. Springer, New York, 1999.

[31] J. Courbon, Y. Mezouar, L. Eck, and P. Martinet. A generic fisheye camera model for robotic applications. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1683–1688, 2007.

[32] B. D. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proc. of 7th International Joint Conference on Artificial Intelligence*, pages 674–679, 1981.

[33] Intel's OpenCV [Online]. Available: http://sourceforge.net/projects/opencvlibrary/

[34] C. Sagüés, A.C. Murillo, J.J. Guerrero, T. Goedemé, T. Tuytelaars, and L. Van Gool. Localization with omnidirectional images using the radial trifocal tensor. In *IEEE International Conference on Robotics and Automation*, pages 551–556, 2006.

[35] E. Marchand and F. Chaumette. Feature tracking for visual servoing purposes. *Robotics and Autonomous Systems*, 52(1):53–70, 2005.