

# Visual Navigation by Means of Three View Geometry

G. López-Nicolás, H. M. Becerra, M. Aranda, and C. Sagüés

**Abstract**—We address the problem of driving autonomously a nonholonomic vehicle to a target location by using a purely vision-based control framework based on three view geometry. We present three different approaches based on the trifocal tensor. The first one is a control law defined by an exact input-output linearization of the trifocal tensor model, with the desired evolution of the system directly defined in terms of the trifocal tensor elements. The second proposal uses a simplified trifocal tensor, avoiding the need of a complete camera calibration, by means of a sliding mode control law ensuring stability and robustness for the closed loop. The third approach employs a reference set of images of the environment previously acquired at different locations. We define a purely angle-based approach, by means of multiple trifocal tensors computed across the image database to solve the homing task. The trifocal tensor presents important advantages for visual control purposes because distance information is not needed and the additional geometric constraints enforced by the tensor improve the robustness in the presence of mismatches. The good performance of the control system is proven via simulations and real world experiments.

## I. INTRODUCTION

Visual servoing is a growing 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 [4] or in cooperative tasks [5], [6]. In this work we tackle the problem of visual servo control of a nonholonomic vehicle with an on-board monocular vision system, which can be a conventional [7] or omnidirectional camera [8]. The goal is to design a control framework for autonomous navigation to a desired location, which is defined by a target image taken previously at that location. The control scheme is based on the trifocal tensor model, which is computed from feature correspondences across three views.

A direct way to face the problem of extracting information from the images is to rely on landmarks or particular features which can be extracted and tracked, and then, these features matched between images are directly included in the control scheme [9]. In favour of robustness, a good choice is to process the image information through a geometric model relating the acquired images. Then, from the set of point correspondences there is less probability that spurious matches

could reduce the control performance. An early work [10] based on the epipolar geometry, where the image information relies on the epipoles, has been followed by others [11]. Nevertheless, the epipolar geometry has main drawbacks. One is that the fundamental matrix is ill-conditioned with short baseline and therefore, a control based on it becomes unstable. Another issue is that the solution is degenerated when all the detected points belong to one plane. A natural way to overcome these drawbacks is using the homography defined by a plane of the scene. This geometric model is robust and well defined with short baseline. Some examples of visual control based on homographies are [12], [13]. However, the performance of a homography-based control can be affected if there is no dominant plane in the scene.

Here, we propose a new framework based on the trifocal tensor. This tensor comprises the intrinsic geometry between three views and it is independent of the observed scene [14]. This geometric model has several advantages: It is more robust than the two view geometry models as it involves the information given by a third view, and the set of matches obtained is more robust to outliers. Besides, the trifocal tensor is still useful with short baseline, whereas the epipolar geometry fails. The problem of localization has been discussed in [15] through the 1D trifocal tensor and with the 2D trifocal tensor [16], [17]. We propose and compare three different methods for solving the control task. In the first method [1], the control law is obtained by an exact input-output linearization of the 2D trifocal tensor. Then, the desired evolution of the system towards the target is directly defined in terms of the trifocal tensor elements by means of sinusoidal functions without needing metric or additional information from the environment. The trifocal tensor is computed from three views, and in our approach the three images are the initial, the current and the target images. So, at the start, the initial and current images are the same and, as the vehicle moves towards the target, the current and target images get similar.

On the one hand, the use of the 2D trifocal tensor has the advantage of providing of all the information available in the 2D images. On the other hand, the 1D trifocal tensor is more robust and it is easier to compute than the 2D trifocal tensor. The second method presented [2] takes advantage of the 1D trifocal tensor, with its elements introduced directly in the control law without requiring any prior knowledge of the scene. 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. The control design consists of

The authors are with Instituto de Investigación en Ingeniería de Aragón. Dept. Informática e Ingeniería de Sistemas, Universidad de Zaragoza, 50018 Zaragoza, Spain.

gonlopez@unizar.es, hector.becerra@unizar.es, marandac@unizar.es, csagues@unizar.es

This paper surveys the work presented in [1], [2] and [3].

This work was supported by project DPI2009-08126, project DPI2009-14664-C02-01, I3A fellowship Program, grant AP2009-3430 Ministerio de Educación and Grant of Banco Santander-Universidad de Zaragoza and CONACYT-México.

a sliding mode control (SMC) law in a square system that ensures stability and robustness for the closed loop. In these approaches, rather than decomposing the trifocal tensor to obtain pose information, the control is performed directly on the trifocal tensor elements.

A different approach for solving the visual control problem is the angle-based homing method using omnidirectional vision, being [18] an early work and [19], [20] more recent contributions. These are purely feature-based approaches where the angles of landmarks in the images are used to generate a homing vector. The third approach [3] presented is a homing method that makes use of the angular information between omnidirectional views extracted by means of the 1D trifocal tensor. This approach employs only the visual information provided by omnidirectional images to obtain the angles between the current position and a set of previously acquired reference images taken at different locations, any of which can be selected as goal position.

The proposed approach has been tested experimentally. The results show the performance of the three proposed methods via simulations with synthetic images, experimental analysis with real images and real-world experiments in closed loop.

## II. VISUAL CONTROL WITH THE 2D TRIFOCAL TENSOR

Three images can be geometrically linked by the trifocal tensor. This tensor only depends on the relative locations between the three views and the internal calibration parameters of the cameras, being independent of the observed scene [14]. Let us suppose that the three images are taken with the same calibrated camera, represented by the pinhole model. The global reference system is defined with the origin attached to the third camera. Then, the locations of the cameras in the global reference are  $\mathbf{C}_1 = (x_1, y_1, z_1)$ ,  $\mathbf{C}_2 = (x_2, y_2, z_2)$  and  $\mathbf{C}_3 = (0, 0, 0)$  with their respective orientations  $\phi_1$ ,  $\phi_2$  and  $\phi_3 = 0$ . Given that we consider planar motion we have  $y_1 = 0$ ,  $y_2 = 0$ . The camera translations in local coordinate systems can be expressed as  $\mathbf{t}_i = (t_{xi}, t_{yi}, t_{zi})^T = -\mathbf{R}_i \mathbf{C}_i$ , with  $i = 1, 2, 3$  and  $\mathbf{R}_i$  the rotation matrices. It can be deduced that the geometry constraints of the three views lead to the trilinear relations that define the trifocal tensor [14] as follows

$$\begin{aligned} T'_{111} &= -t_{x1} \cos \phi_2 + t_{x2} \cos \phi_1 \\ T'_{113} &= t_{x1} \sin \phi_2 + t_{z2} \cos \phi_1 \\ T'_{131} &= -t_{z1} \cos \phi_2 - t_{x2} \sin \phi_1 \\ T'_{133} &= t_{z1} \sin \phi_2 - t_{z2} \sin \phi_1 \\ T'_{212} &= -t_{x1} \\ T'_{221} &= t_{x2} \\ T'_{223} &= t_{z2} \\ T'_{232} &= -t_{z1} \\ T'_{311} &= -t_{x1} \sin \phi_2 + t_{x2} \sin \phi_1 \\ T'_{313} &= -t_{x1} \cos \phi_2 + t_{z2} \sin \phi_1 \\ T'_{331} &= -t_{z1} \sin \phi_2 + t_{x2} \cos \phi_1 \\ T'_{333} &= -t_{z1} \cos \phi_2 + t_{z2} \cos \phi_1 \end{aligned} \quad (1)$$

The other elements of the trifocal tensor are zero as a result of the planar motion constraint.

In our visual control framework,  $(x_1, z_1, \phi_1)$  is the initial location of the vehicle,  $(x_3, z_3, \phi_3) = (0, 0, 0)$  is the target location and  $(x_2, z_2, \phi_2)$  is the current location that varies as the vehicle moves. The goal is to drive the vehicle to the target location. Therefore the objective of the control law is to drive the vehicle to  $(x_2, z_2, \phi_2) = (0, 0, 0)$ . When the vehicle is in the desired target location we have the following values for the trifocal tensor elements

$$\begin{aligned} T'_{111} &= T'_{212} = T'_{313} = -t_{x1}, \\ T'_{131} &= T'_{232} = T'_{333} = -t_{z1}, \end{aligned} \quad (2)$$

with the rest of the trifocal elements equal to zero.

### • Control Law with the 2D trifocal tensor

In this section, the first proposal for vision-based control is presented. An overview of the control loop is as follows: Image features are extracted from the initial and target images, and matched with the features extracted from the current image. Then, the 2D trifocal tensor ( $T_{ijk}$ ) is computed. The input of the control is defined by  $v_{ijk}$  which depends on the trifocal tensor and its desired value ( $T_{ijk}^d$ ). The control law gives the velocities necessary to drive the vehicle to the target location. We consider the nonholonomic differential kinematics of the vehicle modeled with the unicycle equations expressed in state space form  $(x, z, \phi)$  as a function of the translation and rotation velocities of the robot  $(v, \omega)$ .

In the model of the system, the output (the trifocal tensor) is only indirectly related to the input (the robot velocities). Therefore, it is not easy to see how the input can be designed to control the desired evolution of the system. Then, we carry out an exact input-output linearization to find a direct relation between the system variables. So, we transform the problem of nonlinear control into a tracking problem where the desired evolutions of the trifocal tensor elements are defined. The input-output linearization is obtained by means of the derivatives of all the trifocal tensor elements as detailed in [1]. After the first derivative we have already obtained a linear relation between the system input and output. Then, the system can be modeled as follows,

$$\begin{pmatrix} \dot{T}'_{111} \\ \dot{T}'_{113} \\ \dot{T}'_{131} \\ \dot{T}'_{313} \\ \dot{T}'_{331} \\ \dot{T}'_{333} \end{pmatrix} = \begin{bmatrix} 0 & T_{113} \\ \frac{-\cos \phi_1}{T'_N} & -T_{111} \\ 0 & T_{133} \\ \frac{-\sin \phi_1}{T'_N} & -T_{311} \\ 0 & T_{333} \\ \frac{-\cos \phi_1}{T'_N} & -T_{331} \end{bmatrix} \begin{pmatrix} v \\ \omega \end{pmatrix}, \quad (3)$$

where the previous decoupling matrix is denoted with  $\mathbf{L}$ .

From the derivative of the available trifocal tensor elements (1), six of them have been selected for the control of the system. Two velocities of the system are controlled and, in principle, two elements of the trifocal tensor would be enough. However, a two-element based control would fail in solving the control task from some location of the workspace (i.e. there are locations of the workspace in which the robot is uncontrollable with these particular elements). Additionally, with the selection of more elements we can guarantee no singularity of the control. The selection of the elements has been

studied experimentally and the best elements to work with have been found to be the selected ones. Note that no metric information or depth estimation is used in our approach. Then, we define a common scale for tensor by normalizing their elements with  $T'_N = \text{sign}(T'_{232})\sqrt{(T'_{212})^2 + (T'_{232})^2}$ , and  $T_{ijk}$  are the normalized trifocal tensor elements.

Solving (3) for the control outputs we have

$$(v, \omega)^T = \mathbf{L}^+ \nu_{ijk}, \quad ijk = 111, 113, 131, 313, 331, 333 \quad (4)$$

where  $\mathbf{L}^+$  is the pseudo-inverse of  $\mathbf{L}$  and  $\nu_{ijk}$  are the new inputs to be defined. The tracking error is defined as  $(T_{ijk}(t) - T_{ijk}^d(t))$  and we choose

$$\nu_{ijk} = \dot{T}_{ijk}^d - k(T_{ijk} - T_{ijk}^d), \quad (5)$$

$k > 0$  being constant gains.

Once the input-output linearization is carried out and the control law is obtained, the desired evolution of the input control in order to reach the target has to be defined. The input control consists of the tensor elements  $T_{ijk}$ . Then, the objective is to define smooth functions which lead from the initial to the final desired values of the trifocal tensor elements. In [1] we proposed a continuous and time differentiable function for each element of the input control to be tracked as follows:

$$\begin{cases} T_{111}^d(t) = (T_{111}(0) - T_{212}) \frac{\psi(t)}{\psi(0)} + T_{212} \\ T_{113}^d(t) = \frac{T_{113}(0)}{2} + \frac{T_{113}(0)}{2} \cos\left(\frac{\pi t}{t_b}\right) \\ T_{131}^d(t) = (T_{131}(0) - T_{232}) \frac{\psi(t)}{\psi(0)} + T_{232} \\ T_{313}^d(t) = \frac{T_{313}(0) + T_{212}}{2} + \frac{T_{313}(0) - T_{212}}{2} \cos\left(\frac{\pi t}{t_b}\right) \\ T_{331}^d(t) = T_{331}(0) \frac{\psi(t)}{\psi(0)} \\ T_{333}^d(t) = \frac{T_{333}(0) + T_{232}}{2} + \frac{T_{333}(0) - T_{232}}{2} \cos\left(\frac{\pi t}{t_b}\right) \end{cases} \quad (6)$$

where  $\psi$  can be computed from the trifocal tensor as

$$\psi(t) = \arctan\left(\frac{T_{223} \sin \phi_2 - T_{221} \cos \phi_2}{T_{223} \cos \phi_2 + T_{221} \sin \phi_2}\right). \quad (7)$$

If  $(0 \leq t \leq t_b)$  and the final desired values of the trifocal tensor elements, taken from (2) and (1), if  $(t_b < t < \infty)$ . More details about the design of the desired evolution of the trifocal tensor entries as well as the stability analysis of the system under the proposed control law can be found in [1].

### III. VISUAL CONTROL WITH THE 1D TRIFOCAL TENSOR

The 1D Trifocal Tensor is a simplified tensor that relates three views in the frame of planar motion, which is the typical situation in the context of mobile robots. This tensor provides the advantage of being estimated from bearing visual measurements avoiding the need of complete camera calibration. In general, the point features have to be converted to their projective formulation in a 1D virtual retina in order to estimate the 1D trifocal tensor. The computation of this geometric constraint is basically the same for conventional cameras and for central catadioptric systems assuming that all of them approximately obey the generic central camera model. For omnidirectional cameras, a bearing measurement  $\theta$  of an image point, measured with respect to a frame

centered in the principal point of the image, can be converted to its 1D projection as  $\mathbf{p} = (\sin \theta, \cos \theta)$ . By relating this representation for three different views of a feature that is expressed in a 2D projective space, it results in the simplified trifocal constraint

$$\sum_{i=1}^2 \sum_{j=1}^2 \sum_{k=1}^2 T_{ijk} \mathbf{u}_i \mathbf{v}_j \mathbf{w}_k = 0 \quad (8)$$

where  $\mathbf{u} = (\mathbf{u}_1, \mathbf{u}_2)^T$ ,  $\mathbf{v} = (\mathbf{v}_1, \mathbf{v}_2)^T$  and  $\mathbf{w} = (\mathbf{w}_1, \mathbf{w}_2)^T$  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 1D trifocal tensor. Let us define a global reference frame in the plane analogue as in section II with the origin in the third camera. The geometry across the three views in 1D coordinates leads to a similar result to the 2D trifocal tensor as follows:

$$\begin{cases} T'_{111} = t_{z_1} \sin \phi_2 - t_{z_2} \sin \phi_1 \\ T'_{112} = -t_{z_1} \cos \phi_2 + t_{z_2} \cos \phi_1 \\ T'_{121} = t_{z_1} \cos \phi_2 + t_{x_2} \sin \phi_1 \\ T'_{122} = t_{z_1} \sin \phi_2 - t_{x_2} \cos \phi_1 \\ T'_{211} = -t_{x_1} \sin \phi_2 - t_{z_2} \cos \phi_1 \\ T'_{212} = t_{x_1} \cos \phi_2 - t_{z_2} \sin \phi_1 \\ T'_{221} = -t_{x_1} \cos \phi_2 + t_{x_2} \cos \phi_1 \\ T'_{222} = -t_{x_1} \sin \phi_2 + t_{x_2} \sin \phi_1 \end{cases} \quad (9)$$

where  $t_{x_i} = -x_i \cos \phi_i - z_i \sin \phi_i$ ,  $t_{z_i} = x_i \sin \phi_i - z_i \cos \phi_i$  for  $i = 1, 2$ . In order to fix a common scale during the navigation, each estimated element of the tensor is normalized ( $\mathbf{T}_{ijk} = \mathbf{T}'_{ijk}/T'_N$ ) with  $T'_N = T'_{121}$ .

To design a controller for solving the pose regulation problem using only the tensor elements, we have to consider the corresponding final tensor values as control objective, analyze the dynamic behavior of the tensor elements and select an adequate set of them as outputs to be controlled.

Next, the values of the 1D trifocal tensor in particular locations is obtained. Initially, when the second camera is in the starting location then  $\mathbf{C}_2 = \mathbf{C}_1$ , the relative location between these cameras is  $t_{x_2} = t_{x_1}$ ,  $t_{z_2} = t_{z_1}$  and the values of the tensor elements produce the following relationships

$$T_{121} + T_{211} = 0, \quad T_{122} + T_{212} = 0. \quad (10)$$

with the rest of the entries equal to zero. When the robot is in the goal  $\mathbf{C}_2 = \mathbf{C}_3$ , i.e.,  $(x_2, z_2, \phi_2) = (0, 0, 0)$ , the relative location between these cameras is  $t_{x_2} = 0$ ,  $t_{z_2} = 0$ , and it yields the following relationships

$$T_{112} + T_{121} = 0, \quad T_{212} + T_{221} = 0. \quad (11)$$

with the rest of the entries equal to zero.

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 normalized time-derivative of the 1D trifocal

tensor elements yields

$$\begin{aligned}\dot{T}_{111} &= \frac{\sin \phi_1}{T_N} v + T_{121} \omega, & \dot{T}_{211} &= \frac{\cos \phi_1}{T_N} v + T_{221} \omega, \\ \dot{T}_{112} &= -\frac{\cos \phi_1}{T_N} v + T_{122} \omega, & \dot{T}_{212} &= \frac{\sin \phi_1}{T_N} v + T_{222} \omega, \\ \dot{T}_{121} &= -T_{111} \omega, & \dot{T}_{221} &= -T_{211} \omega, \\ \dot{T}_{122} &= -T_{112} \omega, & \dot{T}_{222} &= -T_{212} \omega.\end{aligned}\quad (12)$$

The derivative of the normalizing parameter  $T_N$  is nearly zero in practice. Therefore, given that this tensor element changes slightly as the robot moves, we have assumed that it is constant in the previous derivatives.

- *Control law with the 1D Trifocal Tensor*

We present in this section 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  $z$  positions. The second step uses direct feedback on the tensor to correct orientation.

When the robot reaches the target, it achieves the condition given in (11) and therefore, the following sum of normalized tensor elements are selected as outputs

$$\begin{aligned}\xi_1 &= T_{112} + T_{121}, \\ \xi_2 &= T_{212} + T_{221}.\end{aligned}\quad (13)$$

Then, 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

$$\begin{pmatrix} \dot{e}_1 \\ \dot{e}_2 \end{pmatrix} = \begin{bmatrix} -\frac{\cos \phi_1}{T_N} & T_{122} - T_{111} \\ -\frac{\sin \phi_1}{T_N} & T_{222} - T_{211} \end{bmatrix} \begin{pmatrix} v \\ \omega \end{pmatrix} - \begin{pmatrix} \dot{\xi}_1^d \\ \dot{\xi}_2^d \end{pmatrix}\quad (14)$$

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 treat the tracking problem as the stabilization of the error system (14). We propose a robust control law to solve the tracking problem using SMC, which provides good properties to the control system.

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.

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

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

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

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

$$\mathbf{u}_{disc} = \mathbf{M}^{-1} \begin{pmatrix} -\kappa_1 \text{sign}(s_1) \\ -\kappa_2 \text{sign}(s_2) \end{pmatrix},$$

where  $\kappa_1 > 0$  and  $\kappa_2 > 0$  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 problem

$$\mathbf{u}_{pp} = \mathbf{M}^{-1} \begin{bmatrix} -\lambda_1 & 0 \\ 0 & -\lambda_2 \end{bmatrix} \begin{pmatrix} s_1 \\ s_2 \end{pmatrix},$$

where  $\lambda_1 > 0$  and  $\lambda_2 > 0$  are control gains. Finally, a decoupling-based control law that achieves robust global stabilization of the system (14) is as follows

$$\mathbf{u}_{db} = \begin{pmatrix} v_{db} \\ \omega_{db} \end{pmatrix} = \mathbf{u}_{eq} + \mathbf{u}_{disc} + \mathbf{u}_{pp} = \mathbf{M}^{-1} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix},\quad (15)$$

where  $u_1 = \dot{\xi}_1^d - \kappa_1 \text{sign}(s_1) - \lambda_1 s_1$ , and  $u_2 = \dot{\xi}_2^d - \kappa_2 \text{sign}(s_2) - \lambda_2 s_2$ . 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. Thus, suitable trajectories are defined for  $\xi_1^d$  and  $\xi_2^d$  (see [2] for details). We use the inverse of the decoupling matrix and, to avoid the singularity problem at the final condition, we propose the commutation to a direct sliding mode controller when  $\det(\mathbf{M})$  is near to zero. This issue is detailed in [2]

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

$$\omega = \lambda_\omega \frac{T_{122}}{T_{112}}, \quad t > \tau\quad (16)$$

where  $\lambda_\omega > 0$  is a control gain. Stability analysis of this control scheme is provided in [2], and as a result, robustness of the control system is accomplished.

#### IV. VISUAL HOMING USING THE 1D TRIFOCAL TENSOR

Next, we propose a different approach to take advantage of the information provided by the 1D trifocal tensor. This approach employs only the visual information provided by omnidirectional images to obtain the angles between the current position and a set of previously acquired reference images taken at different locations, any of which can be selected as the goal position.

After the 1D trifocal tensor has been estimated, we can directly extract what are known as the *intrinsic homographies* from the coefficients of the trifocal tensor. Then, the epipoles are extracted from them as their eigenvectors, obtaining the six epipoles of each set of three views. However, there are three ambiguities that need to be resolved in order to determine the correct values of the angles of the 2D epipoles from the values of the epipoles extracted using the 1D trifocal tensor. In [3], we proposed a method for the resolution of

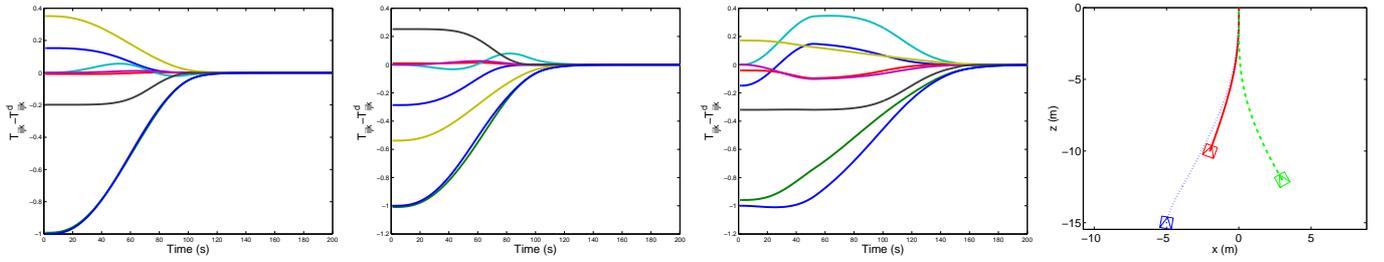


Fig. 2. Three simulations with initial locations at  $(-2, -10, -20^\circ)$ ,  $(3, -12, 30^\circ)$  and  $(-5, -15, -10^\circ)$ . The target location is  $(0, 0, 0^\circ)$ . The three left plots represent the evolution of the 2D trifocal tensor entries error for each example, respectively. Robot paths are plot on the right.

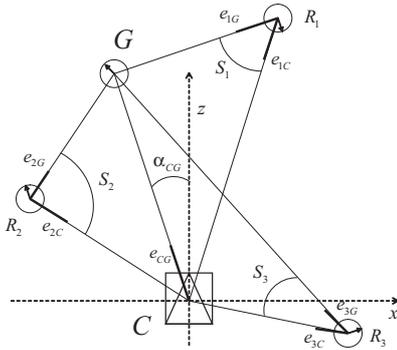


Fig. 1. Elements involved and angles employed in the homing strategy.  $C$  is the robot's current localization, at the coordinate origin  $(0, 0, 0)$ .  $G$  is the goal location.  $R_i$  are reference views. Three of the  $n$  views on the reference set are depicted as example.

the ambiguities in the estimation of the angles, so we can disambiguate the assignment of the complete set of epipoles.

The initial stage of our method involves the calculation of the angular relations between the images on the reference set. This processing can be done off-line and therefore its time consumption is not a critical issue. The aim is to build and store a matrix containing the angles of the epipoles between every pair of reference views, for their use during homing. The procedure for the computation of all the angular information needed for the homing task is explained in detail in [3].

#### • Homing strategy

We now describe the strategy designed in order for the mobile robot to perform homing. The homing method is based solely on the computation of the angles between the locations in which a series of omnidirectional images of the environment were obtained. This group of snapshots consists of the image taken by the robot from its current position and a set of previously acquired reference images, which includes an image obtained at the desired target location. The angles between the views on the reference set have been previously computed and stored, as described in [3]. Therefore only the angles between the robot and the reference views must be worked out during homing.

For every reference view  $R_i(x_i, z_i, \phi_i)$  (where  $x_i, z_i$  and  $\phi_i$  define its position and orientation in the ground plane), the difference between the angles of its epipoles with respect to the current and goal locations defines an angular sector

of size  $S_i = |\alpha_{iC} - \alpha_{iG}|$ , as illustrated in Fig. 1. We use the average value of the angular sizes of these sectors to set the linear velocity at which the robot will move toward the target position

$$v = k_v \text{sign}(\cos \alpha_{CG}) \cdot \frac{1}{n} \sum_{i=1}^n S_i, \quad (17)$$

where  $k_v > 0$  is a control gain. When the target is behind the robot,  $\text{sign}(\cos \alpha_{CG})$  will be negative, therefore generating backward motion. As the robot moves closer to the goal, the mean size of the angular sectors seen from the reference positions will become smaller; thus, the robot's linear velocity will gradually decrease and eventually become zero when the target is reached.

The direction in which the robot travels is determined by the angle at which the goal position is seen from the current location, i.e. the angle  $\alpha_{CG}$  of the epipole  $e_{CG}$ . The angular velocity of the control law is given by

$$\omega = k_\omega (\alpha_{CG} - \alpha_{CG}^d), \quad (18)$$

$$\alpha_{CG}^d = \begin{cases} 0 & \text{if } |\alpha_{CG}| \leq \frac{\pi}{2} \\ \pi & \text{if } |\alpha_{CG}| > \frac{\pi}{2} \end{cases}, \quad (19)$$

where  $k_\omega > 0$  is a control gain. Before tuning the algorithm on a real system, it is strongly recommended to first perform some simulations, and tune the observer gains in such a way that the response of the system is fast enough without saturation. The value of these gains is more related with the distance to be covered as well as the time required to perform the task rather than the images acquired or the environment. From a minimum number of four reference views, one of which would be the view from the target location, the robot will navigate to the home position. Note that the orientation in which the robot reaches the target position is not controlled, since, by definition, the purpose of the homing task is getting to the goal location. The stability of this control law is studied in [3], and the conditions in which the system under the proposed control is globally asymptotically stable are given.

## V. EXPERIMENTAL VALIDATION

In the following, different experiments are presented to show the validity of the approaches and their performance.

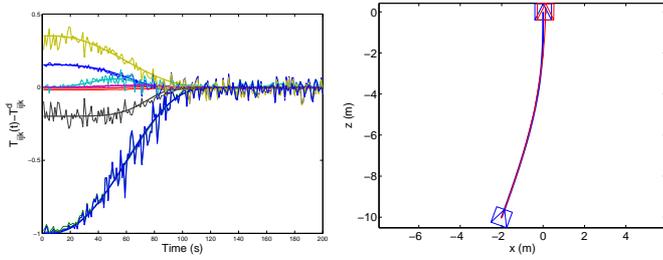


Fig. 3. Simulations with (thin line) and without (thick line) image noise of  $\sigma = 1$  pixel. The initial location is  $(-2, -10, -20^\circ)$ . (Left) Evolution of  $T_{ijk}(t) - T_{ijk}^d$ . (Right) Robot paths.

#### A. Experiments with the 2D trifocal tensor

The virtual scene consists of a random set of 3D points. The points of the scene are projected into the image plane through a pin-hole camera model. The image size is  $640 \times 480$  pixels. The 2D trifocal tensor between the initial, current and target images is computed from the point correspondences. Three simulations from different initial locations are presented in Fig. 2. It can be seen that the evolution of the 2D trifocal elements error along time converges to zero. The motion of the vehicle is also shown in Fig. 2. As it can be seen, the resultant motion is smooth and converges properly to the target location.

The robustness of the 2D trifocal-based control has been tested in presence of image noise. Simulations with and without image noise are superposed in Fig. 3. The image noise of the point correspondences consist in Gaussian noise with a standard deviation of  $\sigma = 1$  pixel in point coordinates. The results show that the control law can cope with image noise converging to the target location successfully.

Before computing the trifocal tensor from the point correspondences they have to be transformed to calibrated coordinates by means of the internal camera calibration parameters. As expected, the simulations carried out show that the final location error increases with the camera calibration parameter errors. However, it is stable and the performance is still acceptable with small calibration errors.

#### B. Experiments with the 1D trifocal tensor

1) *Simulation results:* We present some simulations of the overall control system. The 1D trifocal tensor is estimated from a minimum of five point correspondences in virtual omnidirectional images of size  $1024 \times 768$ . These images have been generated from a 3D scene (Fig. 4) through the generic model for central catadioptric cameras. We report results with hypercatadioptric, paracatadioptric and also fisheye cameras, which can be approximately represented with the same model. Figure 4 shows the paths traced by the robot from four different initial locations.

We can see in Fig. 5(left) 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(right) for the case  $(-5, -12, -30^\circ)$ . 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

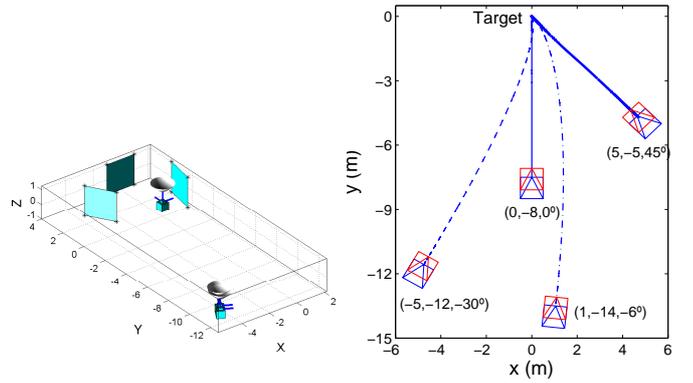


Fig. 4. Simulation results with synthetic images. (Left) 3D scene. (Right) Paths on the plane.

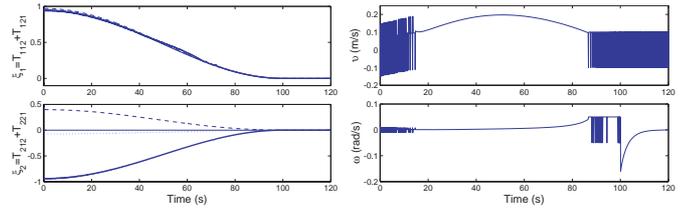


Fig. 5. Control law performance. (Left) Controlled outputs for the four cases of Fig. 4. (Right) Example of the computed velocities for initial location  $(-5, -12, -30^\circ)$ .

also see how the rotational velocity presents an exponential decay after 100 s, which takes the element  $T_{122}$  to zero. This forces the orientation to decrease with a fixed exponential rate, whose settling time is approximately 16.7 s ( $5/\lambda_\omega$ ).

The previous results have been obtained for three different kind of omnidirectional cameras. Figure 6 shows the motion of the image points for the case in which a hypercatadioptric, paracatadioptric and fisheye camera are simulated. For all the experiments, the mean squared tracking error is very low, in the order of  $1 \times 10^{-5}$ .

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. 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. Analysis and discussion of the behavior of the proposed control scheme using SIFT

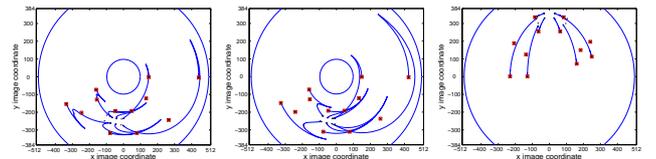


Fig. 6. Motion of the points in the image plane for three different kind of omnidirectional virtual images. (Left) Hypercatadioptric. (Center) Paracatadioptric. (Right) Fisheye. The images depict the point features from the initial, current and target views.

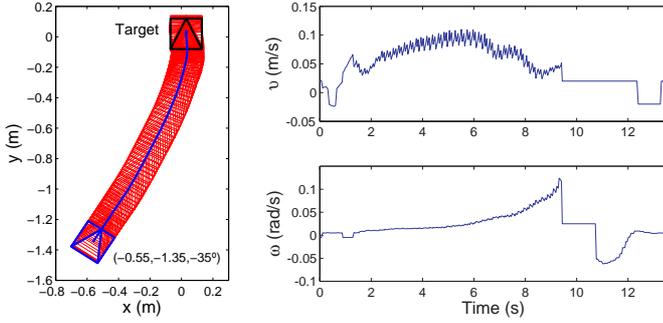


Fig. 7. Experimental results with the control law in closed loop. (Left) Resultant path. (Right) Computed velocities. The data to plot the path is given by the robot odometry.

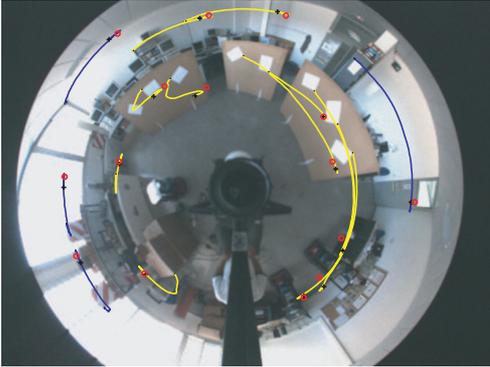


Fig. 8. Motion of the image points to illustrate the behavior of the visual measurements for the real experiments.

features and the Lucas-Kanade pyramidal algorithm can be found in [2].

2) *Real-world experiments*: The proposed approach has been tested in closed loop with real conditions using a Pioneer 3-AT robot. For these experiments, we use an omnidirectional system with a camera Sony XCD-X7101CR and a mirror Neovision H3S to capture images of size  $800 \times 600$  pixels. We have implemented these experiments using the tracking of features because its low computational cost. It gives good closed loop frequency, which leads to a good behavior in the 1D trifocal tensor estimation. Figure 7(left) presents the resultant path, given by odometry, of the closed loop control from the initial location  $(-0.55 \text{ m}, -1.35 \text{ m}, -35^\circ)$  for one of the experimental runs. The duration of the task is almost 14 s, the final position error is around 2 cm and the orientation error is practically negligible. We can see in Fig. 7(right) that the bounded SMC law is applied due to the singularity of the decoupling-based controller.

According to Fig. 8, 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.

In accordance to the results and the methodology presented, we can state that the main advantages of using the 1D trifocal tensor 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

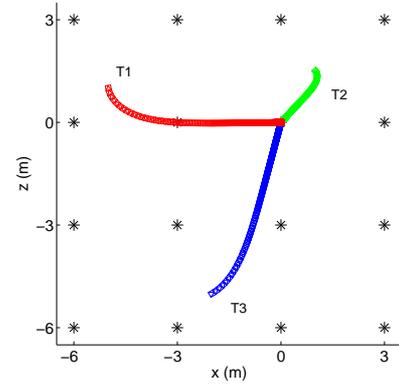


Fig. 9. Robot path of three sample simulated homing trajectories.



Fig. 10. Example image (left), omnidirectional camera (center) and complete setup (right) used for the experiments.

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 commuting to any visual constraint other than the 1D trifocal tensor.

### C. Experiments of homing with the 1D trifocal tensor

The performance of the third proposed method (section IV) has been tested both in simulation and with real images. For the simulations, the reference views were positioned forming a square grid, although any arbitrary distribution guaranteeing sufficient geometric diversity on the plane could be chosen. A randomly distributed cloud of 200 points in 3D was generated and projected in each camera. Three sample homing trajectories with a 16-view reference set are displayed in Fig. 9.

The setup for the real experiments consisted of an ActivMedia Pioneer nonholonomic unicycle robot base with a catadioptric vision system mounted on top. The resolution of the employed images was  $800 \times 600$  pixels. The imaging system is used without specific calibration other than the assumption that the camera and mirror axis are vertically aligned. The images were obtained in an indoor, laboratory setting. The experimental setup is illustrated in Fig. 10. To generate the reference set of views, 20 images were acquired from locations forming a  $5 \times 4$  rectangular grid with a spacing of 1.2 m., thus covering a total area of  $4.8 \times 3.6 \text{ m}^2$ . Features in the images were extracted and matched, and a RANSAC robust estimation was used to calculate the 1D trifocal tensors between the views. The number of three-

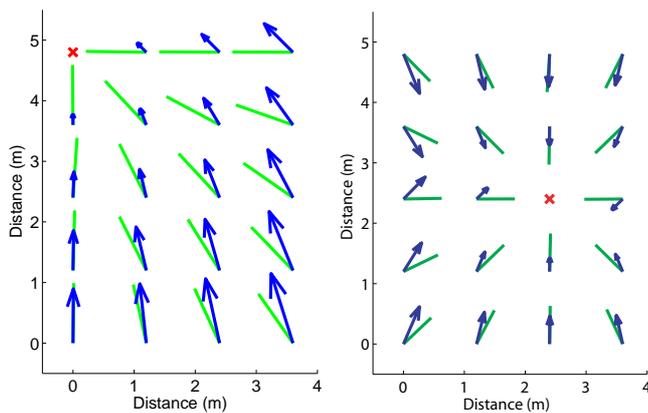


Fig. 11. Displacement vectors (arrows) and directions of the epipoles (line segments) with respect to the goal estimated at every reference position for two different goal locations (marked with a cross) in real setting.

view correspondences employed to obtain the trifocal tensor estimations lied in the range of 30 to 70. Although images taken on opposite sides of the room could not be matched, the connections between adjacent or close sets of views were sufficient to recover the relative angles of the complete reference set. Vector field representations for two different goal locations within the grid are displayed in Fig. 11. The arrows at each location represent the displacement vectors associated with the motion that a vertically oriented robot with nonholonomic constraints would perform from that spot, according to the proposed control law (section IV). The results show good accuracy despite the presence of outliers in the putative matches.

## VI. CONCLUSIONS

We have presented three different vision-based control approaches which are based on the trifocal tensor. The first control law is defined by the exact input-output linearization of the system through the 2D trifocal tensor. Some advantages of the trifocal tensor based approach are that it is more robust than two view geometry thanks to additional information of a third view and that the problem of short baseline with epipolar geometry is overcome. The second control law follows a similar design but taking advantage of the robustness of the 1D trifocal tensor using sliding mode control techniques. From a control theory point of view, an advantage of this approach is that the selected outputs allow us to prove stability on the basis of a square control system. The third controller relies on the 1D trifocal tensor to obtain the angular information provided by a set of omnidirectional reference images, being this information very precise. The designed control law employs these angular relations to guide the robot to the target location. The computational cost of the proposed method is low and it can be directly applied in settings where stored image databases are available. Our proposal assumes that the optical center of the camera is located in the robot rotation axis, and this issue has been further considered in [21], where the camera is translated a distance from the robot rotation axis. Experimental validation

showing the performance of these approaches shows that the methods perform properly.

## REFERENCES

- [1] G. López-Nicolás, J. J. Guerrero, and C. Sagüés, "Visual control through the trifocal tensor for nonholonomic robots," *Robotics and Autonomous Systems*, vol. 58, no. 2, pp. 216–226, 2010.
- [2] 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.
- [3] M. Aranda, G. López-Nicolás, and C. Sagüés, "Omnidirectional visual homing using the 1D trifocal tensor," in *IEEE International Conference on Robotics and Automation*, May 2010, pp. 2444–2450.
- [4] G. N. DeSouza and A. C. Kak, "Vision for mobile robot navigation: A survey," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 2, pp. 237–267, 2002.
- [5] A. K. Das, R. Fierro, V. Kumar, J. P. Ostrowski, J. Spletzer, and C. J. Taylor, "A vision-based formation control framework," *IEEE Transactions on Robotics and Automation*, vol. 18, pp. 813–825, 2002.
- [6] N. Moshtagh, N. Michael, A. Jadbabaie, and K. Daniilidis, "Vision-based, distributed control laws for motion coordination of nonholonomic robots," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 851–860, 2009.
- [7] G. Blanc, Y. Mezouar, and P. Martinet, "Indoor navigation of a wheeled mobile robot along visual routes," in *Proceedings of the IEEE International Conference on Robotics and Automation*, April 2005, pp. 3365–3370.
- [8] A. A. Argyros, K. E. Bekris, S. C. Orphanoudakis, and L. E. Kavraki, "Robot homing by exploiting panoramic vision," *Autonomous Robots*, vol. 19, no. 1, pp. 7–25, 2005.
- [9] 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*, Apr. 2007, pp. 2196–2200.
- [10] R. Basri, E. Rivlin, and I. Shimshoni, "Visual homing: Surfing on the epipoles," *International Journal of Computer Vision*, vol. 33, no. 2, pp. 117–137, 1999.
- [11] P. Rives, "Visual servoing based on epipolar geometry," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, 2000, pp. 602–607.
- [12] Y. Fang, W. E. Dixon, D. M. Dawson, and P. Chawda, "Homography-based visual servo regulation of mobile robots," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 35, no. 5, pp. 1041–1050, 2005.
- [13] S. Benhimane, E. Malis, P. Rives, and J. R. Azinheira, "Vision-based control for car platooning using homography decomposition," in *IEEE International Conference on Robotics and Automation, Barcelona, Spain*, April 2005, pp. 2173–2178.
- [14] R. I. Hartley and A. Zisserman, *Multiple View Geometry in Computer Vision*, 2nd ed. Cambridge University Press, 2004.
- [15] K. Åström and M. Oskarsson, "Solutions and ambiguities of the structure and motion problem for 1D retinal views," *Journal of Mathematical Imaging and Vision*, vol. 12, pp. 121–135, 2000.
- [16] 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*, May 2002, pp. 688–694.
- [17] L. Quan and M. Lhuillier, "Structure from motion from three affine views," in *Proceedings of the 16th International Conference on Pattern Recognition*, vol. IV, Aug. 2002, pp. 1–6.
- [18] J. Hong, X. Tan, B. Pinette, R. Weiss, and E. M. Riseman, "Image-based homing," *Control Systems Magazine, IEEE*, vol. 12, no. 1, pp. 38–45, Feb. 1992.
- [19] A. A. Argyros, K. E. Bekris, and S. C. Orphanoudakis, "Robot homing based on corner tracking in a sequence of panoramic images," in *IEEE Conference on Computer Vision and Pattern Recognition*, 11–14 Dec. 2001, pp. 3–10.
- [20] K. E. Bekris, A. A. Argyros, and L. E. Kavraki, "Angle-based methods for mobile robot navigation: Reaching the entire plane," in *Int. Conference on Robotics and Automation*, 2004, pp. 2373–2378.
- [21] H. M. Becerra and C. Sagüés, "Dynamic pose-estimation from the epipolar geometry for visual servoing of mobile robots," in *IEEE International Conference on Robotics and Automation*, May 2011, pp. 417–422.