Citation: M. Aranda, G. López-Nicolás, and C. Sagüés. Omnidirectional visual homing using the 1D trifocal tensor. In IEEE International Conference on Robotics and Automation, Anchorage, Alaska, USA, pp. 2444-2450, May 2010

Omnidirectional Visual Homing Using the 1D Trifocal Tensor

M. Aranda, G. López-Nicolás and C. Sagüés
DIIS - I3A, University of Zaragoza, Spain
Email: {marandac, gonlopez, csagues}@unizar.es

***Abstract*— This paper presents a new method for visual homing to be used on a robot moving on the ground plane. A relevant issue in vision-based navigation is the field-of-view constraints of conventional cameras. We overcome this problem by means of omnidirectional vision and we propose a vision-based homing control scheme that relies on the 1D trifocal tensor. The technique employs a reference set of images of the environment previously acquired at different locations and the images taken by the robot during its motion. In order to take advantage of the qualities of omnidirectional vision, we define a purely angle-based approach, without requiring any distance information. This approach, taking the planar motion constraint into account, motivates the use of the 1D trifocal tensor. In particular, the additional geometric constraints enforced by the tensor improve the robustness of the method in the presence of mismatches. The interest of our proposal is that the designed control scheme computes the robot velocities only from angular information, being this very precise information; in addition, we present a procedure that computes the angular relations between all the views even if they are not directly related by feature matches. The feasibility of the proposed approach is supported by the stability analysis and the results from simulations and experiments with real images.**

I. Introduction

Vision sensors have been widely used for robot navigational tasks [1] due to the high amount of information that can be extracted from them. In particular, visual homing is often inspired by the mechanisms that certain animal species, such as insects [2], [3], utilize to return to their known home location. Due to the advantages provided by their wide field of view, omnidirectional cameras are being increasingly employed in robot navigation. In contrast to what occurs with their radial information, which is strongly affected by distortion, the angular information provided by these cameras is rich and precise.

Angle-based homing methods using omnidirectional vision have been proposed, being [4] an early work and [5], [6] more recent contributions. These are purely feature-based approaches where the angles of landmarks in the images are used to generate a homing vector. An alternative is to employ the geometric models that relate the views of a scene. Visual control methods have been presented using the epipolar geometry, which expresses the relations between two views [7], and the trifocal tensor, which encapsulates the three-view geometric constraints [8]. In particular, robot navigation on the ground plane lends itself to the use of the 1D trifocal tensor, the unique matching constraint between 1D views of a 2D scene [9]. The 1D trifocal tensor can be computed linearly from point correspondences, and employed to perform 2D projective reconstruction. In robotics, it has been used for 2D localization tasks [10], [11].

This work was supported by projects DPI2009-08126 and DPI2009-14664-C02-01.

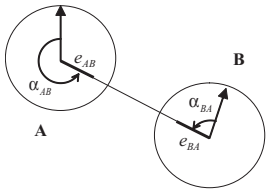

Fig. 1. Nomenclature and conventions used throughout the paper. e_{AB} is the epipole (i.e. the projection of the optical center) of view B in view A. α_{AB} is the angle or direction of that epipole, i.e. its angular polar coordinate in view A. The reference axis for the measurement of the angles is the robot's axis of orientation (the vertical axis of the images). The angles are measured counterclockwise and are defined between $-\pi$ and π.

We propose a homing method that makes use of the angular information between omnidirectional views extracted by means of the 1D trifocal tensor. The three-view geometric constraints enforced by this tensor on the point correspondences make the calculations more robust to outliers when compared to feature-based methods. Our 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 home (or goal) position. A control law based on the estimated angles is used to guide the robot to the target.

The contents of the paper are organized as follows: In section II we discuss the calculation of the angles between views from the estimation of the 1D trifocal tensor. We also propose a method for the resolution of the ambiguities in the estimation of the angles. Section III presents the procedure for the computation of all the angular information needed for the homing task. In section IV the designed control strategy is described. Section V provides an account of the results of the simulations and experiments conducted with real images. Finally, in section VI the conclusion is given. Some nomenclature and conventions used throughout the paper are illustrated in Fig. 1.

II. Angles from the 1D trifocal tensor

The trifocal tensor is the mathematical entity that encapsulates all the geometric relations between three views that are independent of scene structure. In particular, the 1D trifocal tensor relates three 1D views on a plane, and presents

some interesting properties; namely, it is the only matching constraint between 1D views, it can be estimated linearly from a minimum of seven three-view point matches (or five, if the calibration of the cameras is known [12]), and 2D projective reconstruction can be obtained from it.

A. 1D trifocal tensor computation and epipole extraction

The projections of a given point in three 1D views (which we will refer to as A, B and C) on a plane are related by the following trilinear constraint [9]:

$$\sum_{i=1}^{2}\sum_{j=1}^{2}\sum_{k=1}^{2} T_{ijk} u_i^A u_j^B u_k^C = 0, \tag{1}$$

where T_{ijk} are the elements of the $2 \times 2 \times 2$ *1D trifocal tensor*, $\mathbf{T}$, between the views, and $\mathbf{u^A}$, $\mathbf{u^B}$ and $\mathbf{u^C}$ are the homogeneous coordinates of the projections of the point in each view. $\mathbf{T}$ is defined up to a scale factor and therefore can be calculated, in the uncalibrated case, from a minimum set of seven point correspondences across the views.

The process we follow to estimate $\mathbf{T}$ starts by detecting relevant features in three omnidirectional images, e.g. by means of the SIFT keypoint extractor [13], and finding matches between them. The angles (α) of the matched image points, measured counterclockwise from the vertical axis, are converted to a 1D projective formulation, with the corresponding homogeneous 1D coordinates being $(\sin\alpha, \cos\alpha)^T$. In this mapping, the distinction between angles differing by π is lost.

Each of the point matches in 1D projective coordinates gives rise to an equation of the form of (1). If more than seven correspondences are available, we find a least squares solution to the resulting system of linear equations through SVD. In this process, a robust estimation method (RANSAC) is employed in order to reject wrong matches.

After $\mathbf{T}$ has been estimated, the epipoles are extracted from it using a procedure presented in [14] that we briefly describe next. A 1D homography is a mapping between projected points in two lines (two of the 1D views, in our case) induced by another line. From the coefficients of the trifocal tensor, we can directly extract what are known as the *intrinsic homographies*; for example, the two intrinsic homographies from A to B, $\mathbf{K_{AB}}$ and $\mathbf{L_{AB}}$, are obtained by substituting the lines defined by $\mathbf{u^C} = (1,0)^T$ and $\mathbf{u^C} = (0,1)^T$ in (1), yielding

$$\mathbf{K_{AB}} = \begin{bmatrix} -T_{211} & -T_{221} \\ T_{111} & T_{121} \end{bmatrix}, \mathbf{L_{AB}} = \begin{bmatrix} -T_{212} & -T_{222} \\ T_{112} & T_{122} \end{bmatrix}. \tag{2}$$

Now, $\mathbf{H_A}$=$\mathbf{K_{AB}}\mathbf{L_{AB}^{-1}}$ is a homography from A to itself; by definition, the epipoles are the only points that are mapped to themselves by such a homography, i.e.: $\mathbf{e_{AB}} = \mathbf{H_A e_{AB}}$ and $\mathbf{e_{AC}} = \mathbf{H_A e_{AC}}$. Therefore we can calculate them as the eigenvectors of matrix $\mathbf{H_A}$; it is important to note, though, that with this method we do not know which of the other two views (B or C) each of the two recovered epipoles corresponds to. By mapping this pair of epipoles to the other views through the intrinsic homographies, we finally obtain the six epipoles of the set of three 1D views.

B. Ambiguity resolution

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.

Firstly, as mentioned in section II-A, an epipole in a given view recovered from the 1D trifocal tensor may be assigned to any of the two other views. This results in two possible solutions in the assignment of the set of six epipoles between the three views. As shown in [9], [12], both solutions give completely self-consistent 2D projective reconstructions, regardless of the number of point matches between the views. This fundamental ambiguity in the 2D reconstruction from three 1D views can only be resolved through the use of a fourth view, as noted in [10]. The method we employ to resolve the ambiguity operates in the following way: having a group of four views (which we can call A, B, C and D), we calculate two different trifocal tensors between them; for instance, the tensor relating A, B and C, and the tensor between B, C, and D. Since the epipoles between B and C must be identical in the two estimations, by detecting the common (or, in a real situation, the closest) epipoles in these two views we can disambiguate the assignment of the complete set of epipoles.

The origin of the two other ambiguities lies in the fact that in the mapping of 2D image points to 1D projective coordinates, the distinction between bearings differing by π is lost. The angle of a recovered 1D epipole $(e_1, e_2)^T$ is obtained as $\arctan(e_1/e_2)$ in 2D. As a consequence, from the 1D epipole we can extract two different angles in a 2D view, separated by π radians. There are, therefore, four possible solutions for the values of the epipoles between two given views A and B, which may be interpreted as emerging from two combined reconstruction ambiguities; namely, an ambiguity in the direction of the translation vector from view A to view B, which accounts for the difference between solutions (a) and (c) in Fig. 2, and an ambiguity of π radians in the orientation of view B, illustrated, for example, by solutions (a) and (b) in the same figure.

This double ambiguity for a set of two views might be resolved through point reconstruction, but instead we use a simple method employing only the angles of matched image points. We first choose one of the two possible values for each angle. We name these selected angles α^s_{AB} and α^s_{BA}. Although these two choices are arbitrary, let us suppose, for simplicity, that we have picked the angles so that both of them are between 0 and π radians. Our procedure is based on the alignment of the two views, which is achieved through the rotation of the image points by the selected angles. This is done simply by subtracting the selected angle from the angular coordinate of every matched image point. Once the images have been aligned, if the two cameras are pointing in the same direction, the two projections of any given point in them will lie on the same side with respect to the camera's axis of orientation, whereas if the cameras are pointing in opposite directions, the projections will lie on opposite sides

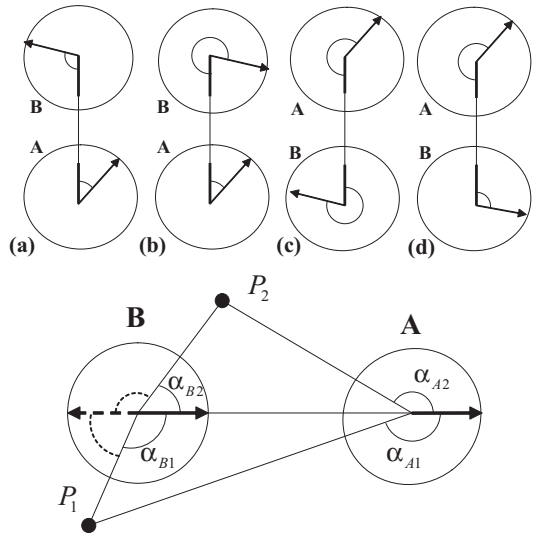

Fig. 2. Four possible 2D reconstructions from the epipoles between two views extracted from the 1D trifocal tensor (top). The relations between the angles of the projections of matched points, such as P_1 and P_2, in two aligned views are used to resolve the 2D reconstruction ambiguities (below).

TABLE I

DISAMBIGUATION OF THE ANGLES OF THE EPIPOLES IN TWO VIEWS

B in front of A	B reversed (∗)	α_{AB}	α_{BA}	Case in Fig. 2
1	1	α^s_{AB}	α^s_{BA}	(a)
1	0	α^s_{AB}	$\alpha^s_{BA}+\pi$	(b)
0	1	$\alpha^s_{AB}+\pi$	$\alpha^s_{BA}+\pi$	(c)
0	0	$\alpha^s_{AB}+\pi$	α^s_{BA}	(d)

(∗)After alignment of the images using the selected angles, α^s_{AB} and α^s_{BA}.

of the axis (Fig. 2). We use a voting procedure integrating the individual results of this test for all the matched points to determine whether aligning the two cameras using the selected angles leaves them pointing in the same direction or reversed with respect to one another.

Next, we want to obtain the sign of the scale of the translation from view A to view B, i.e. establish if B is in front of or behind A. Having the two images aligned and pointing in the same direction (rotating view B by an additional π radians if required) we use the fact that the projections of points in the camera that is in front will give larger angles, in absolute value, than the projected points in the camera that is situated behind. This is again illustrated in Fig. 2. For every matched point, we subtract the absolute values of the angles of its projections in views A and B. We square these results (keeping their sign), in order to give the more discriminant points a greater weight, and then add them up. If the sum is positive, A is estimated as being behind B; otherwise, A is in front of B. This result directly gives us the angle of the epipole in view A, and its combination with the outcome of the orientation test determines the value of the angle in view B, as shown in Table I. Additionally, for every group of three views the joint consistency of the three results of this two-view disambiguation procedure is checked.

III. REFERENCE-SET ANGLES COMPUTATION

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.

We will name that matrix $\mathbf{M}$, with $M(i,j) = \alpha_{ij}$ being the angle of the epipole of view j in view i. All the diagonal elements of $\mathbf{M}$ are defined as zero and the size of the matrix is $(n \times n)$, where n is the number of reference views.

Relevant features are extracted and matched between every pair of images on the reference set, and the resulting point correspondences are stored. We then start an estimation procedure that operates as follows:

- A set of four images (which we can call A, B, C and D) taken in two groups of three (e.g. A-B-C and B-C-D) are processed in each step. For each trio of images we obtain three-view point correspondences by taking the common two-view matches between them. From a minimum number of seven point matches between the three views in each group, we can calculate two trifocal tensors, and we can finally obtain the angles of the epipoles in each of the views of the four-image set (section II).
- We run through the complete set of reference images calculating trifocal tensors and estimating the angles between the views. Whenever there is more than one estimation of a certain angle available, we choose the result that was obtained from the largest set of point matches. In addition, since in real experiments we have found that small sets of correspondences between views tend to produce unreliable results, a minimum threshold can be set for the number of correspondences, below which the calculation of the trifocal tensor for a trio of views is not attempted.
- After the preceding stage is finished, we usually end up with an incomplete $\mathbf{M}$ matrix, due to the impossibility to find sets of three-view point matches linking all the positions on the reference set. There may not be correspondences between images that are far apart; still, groups of adjacent or close images are likely to provide good sets of matches, and from the angles estimated between them we can calculate all the missing angles in matrix $\mathbf{M}$. Specifically, if the angles between two given views i and j are unknown (i.e. $M(i,j)$ and $M(j,i)$ could not be obtained in the preceding stage of the algorithm), we look for a pair of views that are linked with both i and j, and employ the procedure described in section III-A to calculate those two angles. By using this method iteratively, all the elements in matrix $\mathbf{M}$ can eventually be worked out.

The geometric consistency of the triangles obtained in every step of the process is checked, in order to increase the robustness of the results.

A. Complete solution of four-view sets

In practice, it is usually not possible to find matches across all the images. Next, we propose a method to compute all the angular information using the matches between sets of adjacent or close images. A geometric setting of the type shown in Fig. 3, where two triangles are known between the locations of four views, comes up in our method every time

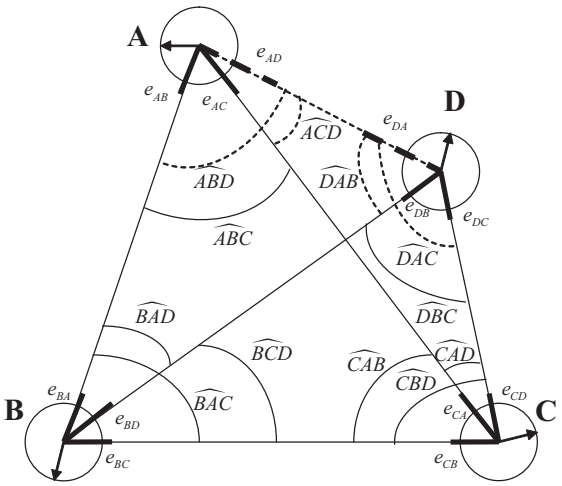

Fig. 3. Geometric setting with four views and two known triangles.

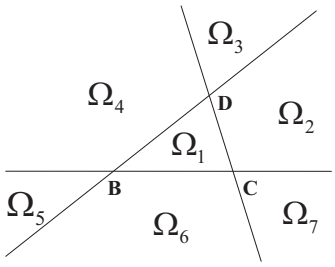

Fig. 4. Seven regions where point A can be located.

TABLE II
VALUES OF SIGNS FOR THE DIFFERENT REGIONS IN WHICH A MAY LIE

Region of vertex A	Relation between angles at vertex A	$sign_1$	$sign_2$
$\mathbf{\Omega_1}$	$\widehat{\mathbf{ACD}} = 2\pi - \widehat{\mathbf{ABD}} - \widehat{\mathbf{ABC}}$	-1	1
$\mathbf{\Omega_2}$, $\mathbf{\Omega_5}$	$\widehat{\mathbf{ACD}} = \widehat{\mathbf{ABD}} + \widehat{\mathbf{ABC}}$	1	-1
$\mathbf{\Omega_3}$, $\mathbf{\Omega_6}$	$\widehat{\mathbf{ACD}} = \widehat{\mathbf{ABC}} - \widehat{\mathbf{ABD}}$	1	1
$\mathbf{\Omega_4}$, $\mathbf{\Omega_7}$	$\widehat{\mathbf{ACD}} = \widehat{\mathbf{ABD}} - \widehat{\mathbf{ABC}}$	-1	-1

we estimate two trifocal tensors from a four-view set. This section describes the method employed to calculate the two unknown angles in this configuration.

We use the notation $\widehat{ABC}$ to refer to the angular size (> 0) of the angles in a triangle. Without loss of generality, we can formulate the problem in the following terms: all the angles from every view to the others in the set are known except the angles of the epipoles between views A and D. Therefore all the angles in the four triangles formed by the set of four views are known, except the ones including both vertices A and D (represented with dashed lines in Fig. 3). Our objective is to calculate the angles α_{AD} and α_{DA} of the epipoles e_{AD} and e_{DA}, which can be directly obtained from the knowledge of the angles of the triangles at those vertices. We start by applying the law of sines on the set of four triangles (ABC, ABD, ACD and BCD), which finally yields the following expression

$$\frac{\sin\widehat{ABD}}{\sin\widehat{ACD}} = K_A, \tag{3}$$

where K_A is a known value given by

$$K_A = \frac{\sin\widehat{CBD}\cdot\sin\widehat{BAD}}{\sin\widehat{BCD}\cdot\sin\widehat{CAD}}. \tag{4}$$

Using the intrinsic relationship between the three angles at vertex A and applying trigonometric identities, we can calculate the individual values of the angles in (3). We must, however, take into account the fact that the location of A with respect to the other three vertices changes the geometry of the set and, consequently, the relation between the angles at the aforementioned vertex. Therefore, we need to divide the plane into seven regions, as shown in Fig. 4, to account for these differences. It turns out that the expression that gives the angle $\widehat{ABD}$ has the same form in all cases (i.e. for all regions), but the signs of two of its terms, denoted as $sign_1$ and $sign_2$, are dependent on the region where A lies

$$\widehat{ABD} = \arctan\frac{sign_1\cdot K_A\sin(\widehat{ABC})}{1 + sign_2\cdot K_A\cos(\widehat{ABC})}. \tag{5}$$

We can easily determine the region in which A is located using the known angles of the epipoles in views B and C, and choose the appropriate values of $sign_1$ and $sign_2$ as shown in table II.

The angle of the epipole of view D in view A is finally obtained as follows

$$\alpha_{AD} = \begin{cases} \alpha_{AB} + \widehat{ABD}, & \text{if } 0 \le \alpha_{BA} - \alpha_{BD} < \pi \\ \alpha_{AB} - \widehat{ABD}, & \text{if } 0 > \alpha_{BA} - \alpha_{BD} \ge -\pi \end{cases}. \tag{6}$$

The angle of the epipole in view D of view A (α_{DA}) can be calculated through a completely analogous process, simply interchanging the roles of vertices A and D. The results are validated using geometric consistency checks. By employing the procedure we have just presented, we can calculate the two unknown angles and thus obtain the complete set of angles between the four views. In addition, this method is useful for two other purposes within our homing technique:

- In the initial stage, detailed in section III, this method allows to fill in the missing elements in the matrix of epipole angles, corresponding to pairs of views that could not be linked directly due to the impossibility to find a sufficiently large set of three-view matches between them.
- During homing, it enables us to obtain all the angles needed to generate the motion commands employing a minimum number of three views; we only need to compute the trifocal tensor between the current image taken by the robot and two of the reference images, which reduces the cost of the algorithm.

IV. HOMING STRATEGY

In this section we describe the strategy designed in order for the mobile robot to perform homing. We assume the robot moves on the ground plane and has nonholonomic motion constraints. 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 section III. Therefore only the angles between the robot and the reference views must be worked out during homing.

In every step of the robot's motion, the camera takes an omnidirectional image, from which key points are extracted. When sufficient point matches are found between the current and two of the reference images, the 1D trifocal tensor is calculated as detailed in section II-A. From the tensor, aided by the knowledge of the angles on the reference set, we can extract the angles between the current and the two other views. Finally, with the method explained in section III-A all the angles of the epipoles in all the views can be computed.

A. Control law

For every reference view $R_i(x_i, z_i, \varphi_i)$ (where x_i, z_i and φ_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. 5. 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

$$v = k_v \, sign(\cos\alpha_{CG}) \cdot \frac{1}{n}\sum_{i=1}^{n} S_i, \tag{7}$$

where $k_v > 0$ is a control gain. When the target is behind the robot, $sign(\cos\alpha_{CG})$ 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 α_{CG} of the epipole e_{CG}. The angular velocity of the control law is given by

$$\omega = k_\omega(\alpha_{CG} - \alpha^d_{CG}), \tag{8}$$

$$\alpha^d_{CG} = \begin{cases} 0 & \text{if} \quad |\alpha_{CG}| \leq \frac{\pi}{2} \\ \pi & \text{if} \quad |\alpha_{CG}| > \frac{\pi}{2} \end{cases}, \tag{9}$$

where $k_\omega > 0$ is a control gain. 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.

B. Stability Analysis

In the following, the stability of the control scheme is analyzed by means of the *Lyapunov's Direct Method* [15]. We define the candidate Lyapunov function as

$$V(\mathbf{x}, t) = \frac{\rho^2}{2} + \frac{(\alpha_{CG} - \alpha^d_{CG})^2}{2} \tag{10}$$

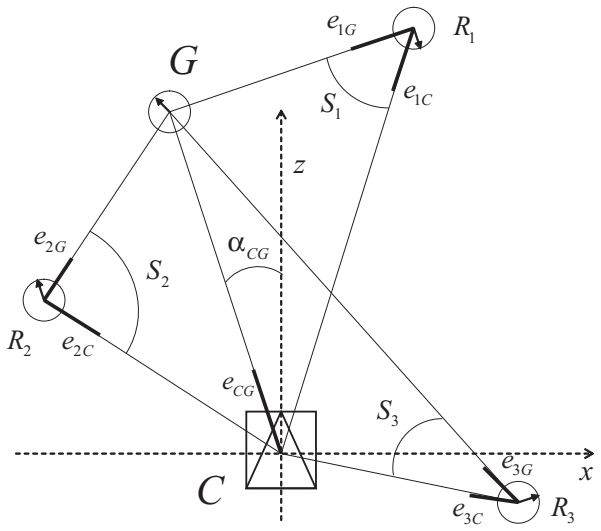

Fig. 5. Elements involved and angles employed in the homing strategy. C is the robot's current localization, at the coordinate origin (0, 0, 0). G is the goal location. R_i are reference views. Three of the n views on the reference set are depicted as example.

where ρ is the distance between the current and goal positions, and $\mathbf{x}$ is the state of the system, determined by ρ and α_{CG}.

Next, we show that the candidate function (10) is a Lyapunov function when using the proposed control law. We need to prove that V is positive definite, $\dot{V}$ is negative definite and V is radially unbounded. The function V is positive definite, given that $V > 0$ for all $\mathbf{x} \neq \mathbf{0}$ and $V(\mathbf{0}) = 0$. It is straightforward that $V(\mathbf{x})$ is radially unbounded, given that $V(\mathbf{x}) \to \infty$ as $\|\mathbf{x}\| \to \infty$. Next, we prove that the derivative $\dot{V}(\mathbf{x})$ is negative definite. The Lyapunov candidate function derivative is

$$\dot{V} = \rho\,\dot{\rho} + (\alpha_{CG} - \alpha^d_{CG})\,\dot{\alpha}_{CG}\ . \tag{11}$$

The dynamics of the system as a function of the input velocities are given, using the derivatives in polar coordinates with the origin at the goal, by $\dot{\rho} = -v\cos(\alpha_{CG})$ and $\dot{\alpha}_{CG} = -\omega + v\sin(\alpha_{CG})/\rho$. Using the control velocities (7), (8) we obtain

$$\begin{aligned} \dot{V} \;=\; & -k_v\,\rho\, sign(\cos\alpha_{CG})\,\cos(\alpha_{CG})\,\frac{1}{n}\sum_{i=1}^{n} S_i \\ & -k_w(\alpha_{CG} - \alpha^d_{CG})^2 - (\alpha_{CG} - \alpha^d_{CG}) \\ & \cdot\sin(\alpha_{CG})\,\frac{k_v}{\rho}\,sign(\cos\alpha_{CG}) \cdot \frac{1}{n}\sum_{i=1}^{n} S_i\ . \end{aligned} \tag{12}$$

By definition $\rho \geq 0$ and $S_i \geq 0$. It is straightforward that the first two terms of (12) are negative definite, but the last term can be positive. The interpretation is that the convergence speed provided by the angular velocity has to be higher than the linear velocity. Otherwise, the angular error is not corrected fast enough and the robot will move following spirals around the goal. However, the stability can be guaranteed if the control gains are selected properly. It is

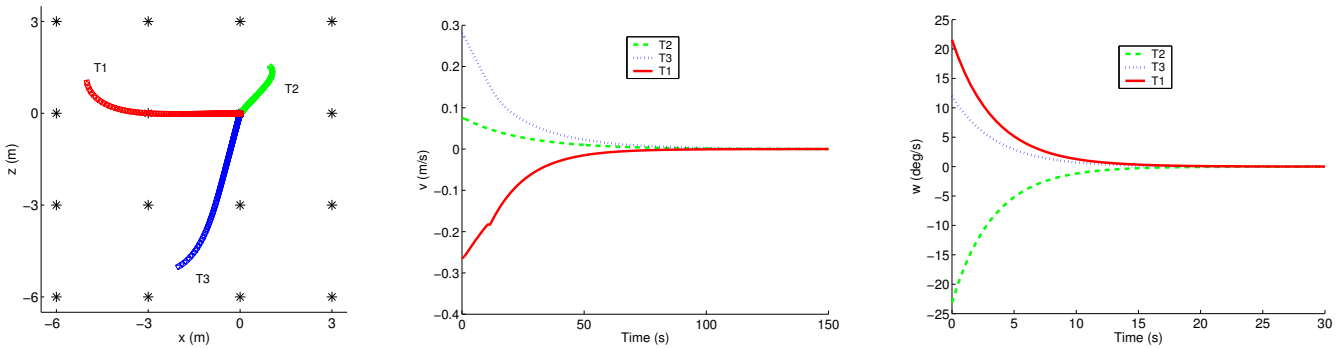

Fig. 6. Robot path (left), linear velocity (center) and angular velocity (right) of three sample simulated homing trajectories.

guaranteed that $\dot{V} < 0$ if the next inequality holds

$$|k_w \cdot (\alpha_{CG} - \alpha^d_{CG})| > |\sin(\alpha_{CG}) \frac{k_v}{\rho} \cdot \frac{1}{n} \sum_{i=1}^{n} S_i| \,. \quad (13)$$

The terms of the right expression are bounded except $1/\rho$. However, note that $\sum_{i=1}^{n} S_i/(n\rho)$ is bounded indeed. We can express S_i as a nonlinear function of ρ: $\sin(S_i) = f(\rho, \overline{R_i G}, \alpha_{R_i G})$; when $\rho \to 0$ we can approximate $\sin(S_i) \approx S_i$ and then $S_i = \rho\, f(\overline{R_i G}, \alpha_{R_i G})$. Therefore, the right part in (13) is bounded. Then, taking into account the initial value $\alpha_{CG}(0)$, we can always define k_ω in such a way that $\dot{V} < 0$ and the system under the proposed control is globally asymptotically stable.

V. Experimental results

The performance of the proposed method has been tested both in simulation and with real images.

A. Simulations

For the simulations shown below, 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 and the evolutions of their corresponding motion commands are displayed in Fig. 6. A maximum threshold was set in order to limit the variation of the sizes of the individual sectors S_i between two consecutive steps; this avoids abrupt changes in the linear velocity that may occur when the robot moves right across one of the reference positions.

We also added Gaussian noise to the angles of the projected points to evaluate the performance of the homing method. Figure 7 displays the final position error obtained after adding variable noise in simulations with sets of 4 (the minimum number for our method) and 16 reference images. Increasing the number of reference views makes the system more robust to noise, since the control operates averaging the contributions of the individual views.

B. Experiments with real images

The setup for the real experiments consisted of an ActivMedia Pioneer nonholonomic unicycle robot base with a catadioptric vision system, made up of a Point Grey FL2-08S2C camera and a Neovision HS3 hyperbolic mirror, mounted on top. The resolution of the employed images was 800 × 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. 8.

To generate the reference set of views, 20 images were acquired from locations forming a 5 × 4 rectangular grid with a spacing of 1.2 m., thus covering a total area of 4.8 × 3.6 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-view correspondences employed to obtain the trifocal

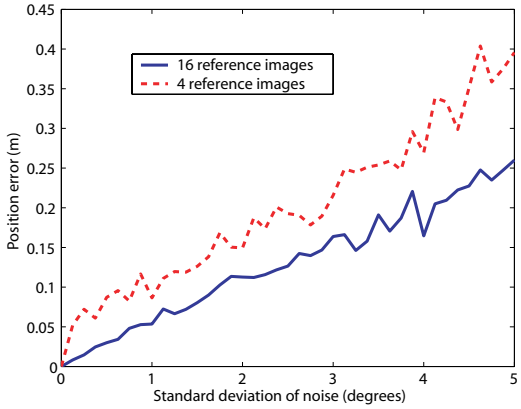

Fig. 7. Final position error vs Gaussian noise.

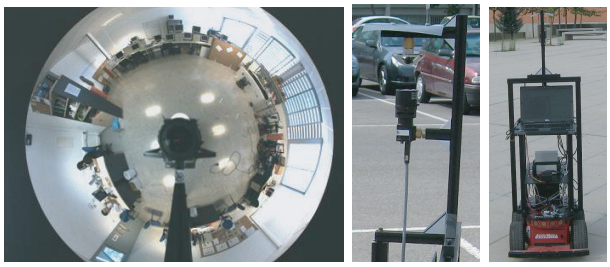

Fig. 8. Example image (left), omnidirectional camera (center) and complete setup (right) used for the experiments.

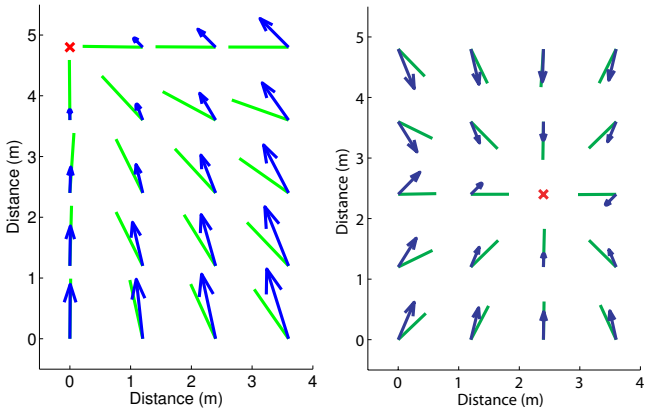

Fig. 9. Displacement vectors (arrows) and directions of the epipoles (line segments) with respect to the goal estimated at every reference position for two different goal locations (marked with a cross) in real setting.

tensor estimations lied in the range of 30 (the threshold below which the results started to become unreliable) 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. 9. 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. They all have been scaled by an equal factor. As can be seen, the magnitude of the vectors becomes larger as the distance to the target increases. The line segments show the estimated directions of the epipoles of the goal position in each of the reference locations. The results show good accuracy despite the presence of outliers in the putative matches.

A sequence of 170 images was captured by the robot while moving at constant speed along a straight-line, 5 m. long diagonal path crossing the grid from one of its outer sides to reach a goal position near the opposite side. The linear velocity commands that the homing method would generate at every step in the sequence and the estimated angle from the current to the goal position (to which the angular velocity of the control law would be proportional) are displayed in Fig. 10. The results of these preliminary experiments show that the homing method can be successful in an environment with sufficiently large sets of feature matches. Further illustration of the experimental results can be seen in the video attachment.

VI. Conclusion

We have presented a method for omnidirectional visual homing to be used on a robot moving on the ground plane. The visual information is provided by a set of omnidirectional reference images, and the 1D trifocal tensor is the tool used to estimate the angular information, being this information very precise. We have presented a method to compute all the angular relations between the views even

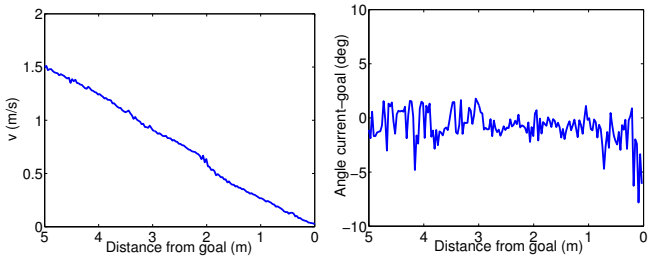

Fig. 10. Linear velocity (left) and angle to the goal (right) estimated in real image sequence.

if there is no direct matching information between them. The designed control law employs these angular relations to guide the robot to the target location. The stability of this control law has been analyzed and the experimental evaluation shows the feasibility of the proposed technique. The computational cost of the proposed method is low, and the speed of its implementation is mainly limited by the time consumption of the feature extraction process. The method can be directly applied in settings where stored image databases are available. In addition, it can be robust to changes in the environment, as long as sufficient features between images can be matched.

References

[1] 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.

[2] D. Lambrinos, R. Möller, T. Labhart, R. Pfeifer, and R. Wehner, "A mobile robot employing insect strategies for navigation," *Robotics and Autonomous Systems*, vol. 30, no. 1-2, pp. 39–64, 2000.

[3] K. Weber, S. Venkatesh, and M. V. Srinivasany, "Insect-inspired robotic homing," *Adaptive Behavior*, vol. 7, pp. 65–97, 1998.

[4] 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.

[5] A. A. Argyros, K. E. Bekris, and S. C. Orphanoudakis, "Robot homing based on corner tracking in a sequence of panoramic images," in *Computer Vision and Pattern Recognition Conf.*, 2001, pp. 3–10.

[6] 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.

[7] 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.

[8] 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.

[9] L. Quan, "Two-way ambiguity in 2D projective reconstruction from three uncalibrated 1D images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 2, pp. 212–216, 2001.

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

[11] 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*, vol. 24, no. 2, pp. 494–501, 2008.

[12] K. Åström and M. Oskarsson, "Solutions and ambiguities of the structure and motion problem for 1D retinal vision," *Journal of Mathematical Imaging and Vision*, vol. 12, no. 2, pp. 121–135, 2000.

[13] D. Lowe, "Distinctive image features from scale-invariant keypoints," *Int. Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[14] A. Shashua and M. Werman, "Trilinearity of three perspective views and its associated tensor," in *International Conference on Computer Vision*, 1995, pp. 920–925.

[15] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2001.