DIIS - I3A
Universidad de Zaragoza
C/ María de Luna num. 1
E-50018 Zaragoza
Spain

Internal Report: 2005-V01

Visual correction for mobile robot homing [1]

C. Sagüés, J.J. Guerrero

If you want to cite this report, please use the following reference instead:
Visual correction for mobile robot homing, C. Sagüés, J.J. Guerrero,
Robotics and Autonomous Systems, Vol. 50, no. 1, pp. 41-49, 2005.

[1] This work was supported by projects DPI2000-1272 and DPI2003-07986.

Visual correction for mobile robot homing ⋆⋆

C. Sagüés *, J.J. Guerrero

Dpto. de Informática e Ingeniería de Sistemas, Universidad de Zaragoza, Edificio Ada Byron, C/ María de Luna num. 1, E-50018 Zaragoza, Spain.

Abstract

We present a method to send a mobile robot to locations specified by images previously taken from these positions, which sometimes has been referred as homing. Classically this has been carried out using the fundamental matrix, but the fundamental matrix is ill conditioned with planar scenes, which are quite usual in man made environments. Many times in robot homing, small baseline images with high disparity due to rotation are compared, where the fundamental matrix also gives bad results. We use a monocular vision system and we compute motion through an homography obtained from automatically matched lines. In this work we compare the use of the homography and the fundamental matrix and we propose the correction of motion directly from the parameters of the 2D homography, which only needs one calibration parameter. It is shown that it is robust, sufficiently accurate and simple.

Key words: Mobile robots, Vision, Lines and homographies, Teaching by doing, Robot homing.

1 Introduction

Robot navigation normally has involved the use of commands to move the robot to the desired position. These commands include the required position and orientation, which suppose good measurement of the motion made by the robot. However, odometry errors or slipping and mechanical drifts may make the desired position not to be reached. Therefore, the use of an additional

⋆⋆This work was supported by projects DPI2000-1272 and DPI2003-07986.

* Corresponding author, Phone 34-976-761940, Fax 34-976761914.
Email address: csagues@unizar.es (C. Sagüés).

perception system is mandatory. Vision is perhaps the most broadly researched perception system.

Using vision, some autonomous vehicles are able to execute tasks based on landmarks which give global localization (Map-Based Navigation). Others carry out specific tasks, building maps of the environment simultaneously (Map-Building-Based Navigation). Our system can be classified in a third group as Map-less navigation [2], because the robot can autonomously navigate without prepared landmarks or complex map-building systems. In our work the target positions are specified with images taken and memorized in the teaching phase. In the playback phase the motion correction to reach to the target position is computed from a projective transformation, which is obtained by processing the current image and the stored reference image. The geometric features extracted and matched are lines which have some advantages with respect to points [3], specially in man made environments.

This kind of navigation using a representation of the route with a sequence of images has been previously considered by a correlation based matching of the current and the reference images [7]. Extracting and matching geometric information from images is not currently time costly and geometric based approaches are less sensitive to noise or illumination changes than others. Of course, the data of the target images to memorize is lower in our proposal, since only extracted lines are stored. Other authors [9] also use vertical lines to correct robot motion, but using a calibrated trinocular vision system.

The recovering of motion from geometric features has been presented using the epipolar geometry [1]. Rotation and direction of translation are initially computed from the essential matrix. In addition, the steps to collision are also computed using a third image. However, there are situations where the fundamental matrix is not meaningful (small translations or planar scenes) and other models are needed to obtain motion [5,8]. We recover motion from homography solving matches of lines automatically. To match them, we use image information and the constraints imposed by the projective transformation. Besides that, robust statistical techniques are considered, which make the complete process useful in real applications. Some experiments show results in typical situations of visual robot homing (§6).

2 Motion from two images

In this work, motion information is obtained from the previously stored image and the current image, taken both with the same camera. We first discuss the ways to compute motion from two images of a man made environment, where straight lines and planar surfaces are plentiful.

Two perspective images can be geometrically linked by linear algebraic relations: the fundamental matrix and the homography. An homography relates points or lines in one image belonging to a plane of the scene with points or lines in the other image. On the other hand, fundamental matrix provides a general and compact representation of the geometric relations between two uncalibrated images of a general 3D scene [6]. It thus provides a mapping from an image point to its epipolar line, but neither a point to point nor a line to line mapping.

Let us suppose two images whose projection matrices in a common reference system are $\mathbf{P}_1 = \mathbf{K}[\mathbf{I}|\mathbf{0}]$ and $\mathbf{P}_2 = \mathbf{K}[\mathbf{R}|\mathbf{t}]$ being $\mathbf{R}$ and $\mathbf{t}$ the camera rotation and translation, respectively, and $\mathbf{K}$ the internal camera calibration matrix.

The homography $\mathbf{H}$ can be related to camera motion [5] as,

$$\mathbf{H} = \mathbf{K}\,(\mathbf{R} - \frac{\mathbf{t}\,\mathbf{n}^T}{d})\,\mathbf{K}^{-\mathbf{1}} \tag{1}$$

being $\mathbf{n}$ the normal to the scene plane and d the plane depth.

The camera motion and the planar structure can be computed from $\mathbf{H}$ when the camera is calibrated [13]. In this case two solutions for motion, with a scale factor for $\mathbf{t}$ and the depth of the plane, are computed.

Camera motion can also be related with the fundamental matrix, which is a 3×3 matrix of rank 2 which encapsulates the epipolar geometry. The fundamental matrix can thus be expressed as $\mathbf{F} = \mathbf{K}^{-T}\ ([\mathbf{t}]_\times\, \mathbf{R})\ \mathbf{K}^{-1}$, being $[\mathbf{t}]_\times$ the antisymmetric matrix obtained from vector $\mathbf{t}$. Given the calibration matrix, the motion can be deduced from $\mathbf{F}$ as follows [5]:

- Compute the essential matrix $\mathbf{E} = \mathbf{K}^T\,\mathbf{F}\,\mathbf{K}$
- Compute the singular value decomposition of matrix $\mathbf{E}$, in such a way that $\mathbf{E} = \mathbf{U}\,diag(1,1,0)\,\mathbf{V}^T$
- The camera translation, up to a scale factor is $\mathbf{t} = \mathbf{U}\,(0,0,1)^T$
- The two solutions for the rotation matrix are $\mathbf{R} = \mathbf{U}\,\mathbf{W}\,\mathbf{V}^T$ and $\mathbf{R} = \mathbf{U}\,\mathbf{W}^T\,\mathbf{V}^T$, being $\mathbf{W} = \left[(0,1,0)^T, (-1,0,0)^T, (0,0,1)^T\right]$

The homography can be obtained from two images using point or line matches [4]. The fundamental matrix can be computed from corresponding points [14]. It can also be computed from homographies obtained through two or more planes, and in this case, corresponding lines in two images can be used [10]. In case of pure rotation or if there exists only one plane in the scene, the epipolar geometry is not defined and only one homography can be computed.

The fundamental matrix has been classically used to compute motion [12], [1]. In our robotic application the rotation must be computed accurately, because even a small rotation error produces high location drifts when the robot advances. A pair of images with small baseline are, in these cases, many times compared because the robot moves with odometry to locations close to target positions, where reference images were taken in the teaching phase. Besides that, the planar structure is dominant in man made environments. In these cases the homography turns out to be better than the fundamental matrix.

3 Computation of 2D homography

Our approach takes straight lines in the image as key features, because they are plentiful in man made environments. The straight lines have a simple mathematical representation, they can be extracted more accurately than points and they can be used in cases where there are partial occlusions. After extracting the lines, automatic computation of correspondences and homographies is carried out as previously presented [4]. Thus, the extracted lines are initially putatively matched to the weighted nearest neighbor using brightness-based and geometric-based image parameters. From them, robust homographies are computed, allowing to detect and reject wrong matches, and growing also additional matches in the final stage.

In most cases, and specially when the robot moves in man made environments, the motion is on a plane and vertical lines give enough information to carry out homing. With vertical lines, only the x coordinate is relevant and the problem is simplified. As the homography has now four parameters up to a scale factor, and each vertical line gives one equation, three matches are enough.

The proposed matching is similar to the presented for lines in all directions [4], which uses robust estimation techniques. Here, we can use the point based formulation because a vertical line can be projectively represented as a point with $(x, 1)$ coordinates. Therefore, for each corresponding vertical line having x_{m1} and x_{m2} coordinates in both images we have

$$\begin{pmatrix} \lambda\, x_{m2} \\ \lambda \end{pmatrix} = \mathbf{H} \begin{pmatrix} x_{m1} \\ 1 \end{pmatrix} = \begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{pmatrix} \begin{pmatrix} x_{m1} \\ 1 \end{pmatrix},$$

which provides (eliminating the scale factor λ) one equation to solve $\mathbf{H}$

$$\begin{pmatrix} x_{m1} & 1 & -x_{m1}x_{m2} & -x_{m2} \end{pmatrix} \begin{pmatrix} h_{11} & h_{12} & h_{21} & h_{22} \end{pmatrix}^T = 0.$$

With the coordinates of at least, three vertical lines, we can construct a 3x4 $\mathbf{M}$ matrix. The homography solution corresponds to the eigenvector associated to the least eigenvalue of the $\mathbf{M}^T\mathbf{M}$ matrix and it can be solved by singular value decomposition. To compute the homography, we have implemented the least median of squares, which is a robust method to consider the existence of outliers [11]. This method makes a random search of the solution from subsets of minimum number of putative matches. The subsets to explore are established from the estimated ratio of outliers and the probability of missing the computation. The robust solution is selected using the median of the squares of the residue as quality criterion.

In the application of robot homing we obtain about 25% of wrong putative matches, but using the just commented robust technique we obtain less than 5% of wrong final matches and even in many cases (depending on the motion and the scenes) we obtain 100% of correct matches. As commented earlier, complete details of this robust matching have been previously reported [4].

4 2D homography and planar motion

Let us suppose vertical images of vertical planar scenes. The relative motion between the reference system attached to the first camera position and the reference attached to the planar scene can be written as a function of the distance d from the origin to the plane and the angle φ of the plane with respect to the camera reference system. Similarly, the transformation between both camera locations can be written as a function of the rotation θ and the translation (t_x, t_z). Developing equation (1) with a centered reference system $(C_x = 0)$, and naming f the focal distance, the homography can be expressed as a function of these scene and motion parameters as,

$$\mathbf{H} = \begin{bmatrix} \frac{1}{C\varphi}\left(t_z\, S\varphi\, S\theta - t_x\, S\varphi\, C\theta + d\, C\varphi\, C\theta\right) & f\left(-d\, S\theta + t_z\, S\theta - t_x\, C\theta\right) \\ \frac{1}{f\, C\varphi}\left(-t_z\, S\varphi\, C\theta - t_x\, S\varphi\, S\theta + d\, C\varphi\, S\theta\right) & d\, C\theta - t_z\, C\theta - t_x\, S\theta \end{bmatrix} \tag{2}$$

Let us write this homography as: $\mathbf{H} = \begin{bmatrix} \alpha & \mu \\ \rho & 1 \end{bmatrix}$

4.1 Approaching motion from homography

We can make some approximations to this equation, trying to avoid eye-wheel calibration. As it will be shown, the approximation proposed is accurate, simple and robust in real situations, where image disparity is mainly due to robot rotation and there exist random perturbations.

If we make $t_x \simeq 0$ and $t_z \simeq 0$ in equation (2), it can be observed that the influence of the scene disappears, and then $\mu = -f \; tan\theta$ and $\rho = \frac{tan\theta}{f}$. Rotation can thus be computed in two ways, from μ and from ρ, being both proportional to $tan\theta$ through a previously calibrated parameter. The rotation (θ) can also be directly computed in the uncalibrated case from **H**. As the eigenvalues are invariant to the change of reference, they are the same as those of the rotation, in such a way that they can be computed from the eigenvalues of the homography. So, the eigenvalues of **H** will be $e^{j\,\theta}$ and $e^{-j\,\theta}$, θ being the camera rotation. In this case, only the value and not the direction of rotation can be computed, although no internal camera parameters are now needed.

On the other hand, we may be interested in the translation, and then we can make $\theta = 0$, but now the simplifications on the equation (2) are no so drastic. However, when the robot moves close to a previously learnt position the scene depth (d) will be larger than the corresponding translation between the current and the reference position. Considering also that the computed homography does not correspond to a scene plane perpendicular to the image plane, we can assume $d \gg t_x \, tan\varphi$, and therefore we will have,

$$\alpha \simeq \frac{d}{d - t_z} \tag{3}$$

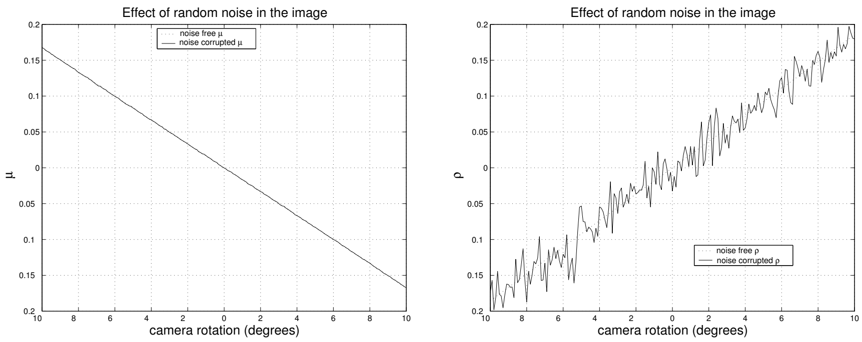
The scale factor between depth and translation remains here and, therefore, it is more correct to express the translation in relative units. So, the relative translation along the z axis can be computed as,

$$t_{z_r} = \frac{t_z}{d} \simeq \frac{\alpha - 1}{\alpha}, \tag{4}$$

which will be useful in our application, because it allows to correct the advance of the robot when it is looking ahead.

The relative translation along the x axis mainly affects to μ, but we do not explicitly compute it and we do not correct the t_x translation, because it is indirectly corrected as if it was a rotation. The scene direction can also be computed from equation (2), but in robot homing we are not normally interested on its computation.

The simplifications just explained are a reasonable solution in many practical situations. In the next sections we analyze the accuracy and the robustness comparing with other solutions, such as those obtained from the fundamental matrix.



(a) (b)

Fig. 1. Effect of gaussian random noise (zero mean and 1 pixel standard deviation) on the parameters used to compute the rotation. For parameter μ, the noise corrupted behavior is nearly over the noise free behavior (a).

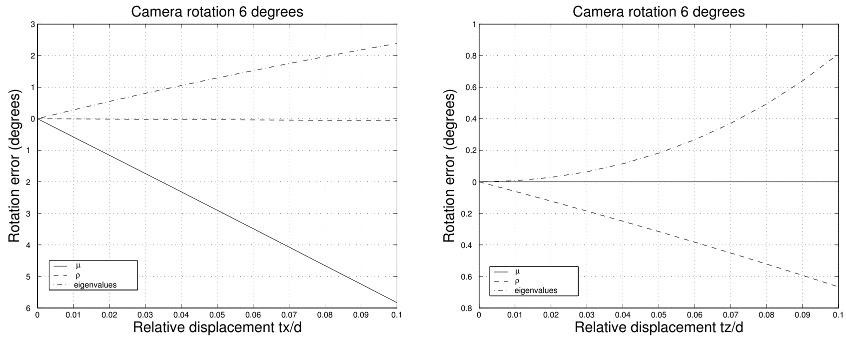
5 Performance of approximate motion from homography

We have built a simulator of scenes and motions and we have compared the homography parameters obtained in different situations in relation to the motion. The comparison has been made in three situations: Rotation with no translation, translation with no rotation, and rotation with translation. As told (§4), the rotation can be approached from μ, ρ or from the eigenvalues and the relative translation along the z axis (which indicates the advance of the robot when it is looking ahead) can be estimated from α.

Rotation.

Image processing and other not considered errors will be simulated as gaussian random noise added to image coordinates of lines. We can appreciate the influence, on the homography parameters, of image random noise of zero mean and 1 pixel standard deviation (Fig. 1). Parameter μ is much less affected by random noise, therefore it will be very good to estimate the rotation and only the calibration of the slope will be needed (Fig. 1.a). We can see how ρ is highly affected by random noise and as a consequence this parameter will not be used to compute rotation (Fig. 1.b). The behavior of the rotation estimate from eigenvalues is similarly affected by noise but it has the advantage of being independent of internal camera parameters.

We have also made experiments to show how combined motions could affect the rotation error depending on the parameter used to obtain the rotation (Fig. 2). As in many other cases, in our application the robot has non-holonomic constraints so, when the camera looks at the direction of advance, the translation along the z axis is the most important to be corrected. We can see that μ is not affected by the translation along the z axis. It can also be appreciated how the estimation of rotation through eigenvalues is affected by the t_z



(a) (b)

Fig. 2. Effect of relative translation on the estimation error of rotation. (a) Relative translation along x axis, (b) relative translation along z axis.

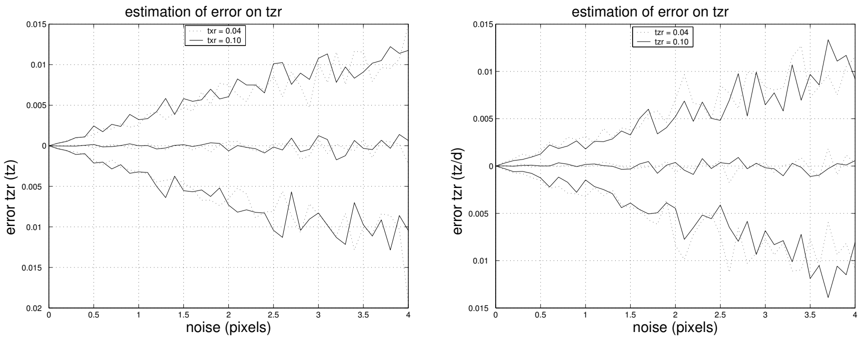
translation. On the other hand, the translation along the x axis affects μ and it does not affect the rotation error when it is estimated through ρ (Fig. 2.a) although it was previously shown that ρ is highly affected by random noise (Fig. 1.b).

The most relevant conclusions that confirm μ as a good parameter to correct heading are the following:

- Random noise affects ρ to the largest extent. It affects eigenvalues to a lesser extent and it nearly does not affect μ.
- The μ parameter is not affected by translation along the z axis.
- Although independent rotations may be supposed, in real situations the robot will arrive at reference locations with some translation deviation. As it can be seen (Fig. 2.a), translation along the x axis affects the most to μ. This confirms that, although deviation along the x axis is not normally too high, it is coupled with rotation, being this coupling captured by μ. In our application this turns out advantageous, because computing rotation from μ produces an over correction that compensates the lateral deviation when the robot advances. This compensation is higher with small scene depth, and it is stable when the camera looks towards the direction of advance.

Translation.

As said, the rotation is the most important information to make a robot correct its trajectory, but relative translation along the z axis can also be easily approached from the homography parameters. To solve the scale factor, in practice the correction of advance must be made in two steps, using the displacement provided by odometry. We show the error in the estimation of relative displacement along the z axis when we make a displacement along the x axis (Fig. 3.a) and when we make a displacement along the z axis (Fig. 3.b). It can be seen that the error does not depend on the value of the displacement (0.04 or 0.10) and they only depend on the random image noise.



(a) (b)

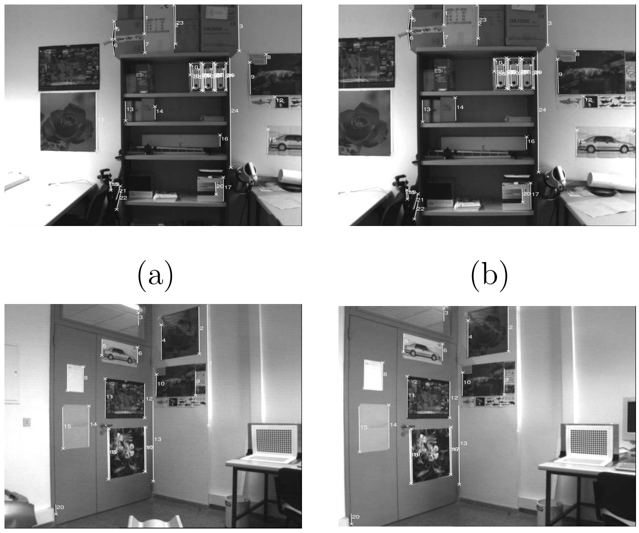
Fig. 3. Effect of random noise on the estimation of the z relative translation: mean error (central line) and confidence expressed to ± 3 times standard deviation for two values of displacement along the x axis (a), and along the z axis (b).

6 Experimental results

We have performed several experiments to evaluate the accuracy with real images (Fig. 4) computing the motion from the μ parameter. We compare it with the rotation computed through the fundamental matrix. The homography has been automatically computed using lines as proposed in [4] and the fundamental matrix could be obtained from at least two homographies [10], but we have used the "image-matching" software [15] to have a more standard benchmark to compare results. The ground truth has been measured with a precision head (UTR80, Manufactured by Newport, Resolution $1/60^o$). We center our attention in the computation of rotation.

We consider three cases. In case 1, the motion is a pure rotation of 6 degrees around the vertical axis viewing the scene of (Fig. 4.a). The fundamental matrix was computed using approximately 300 points whereas the 2D homography has been computed from about 30 automatically matched vertical lines. As described in section 3 three lines are enough and in real situations no less than ten lines are usually available. Each case has been repeated ten times and we give the mean and the standard deviation of the computed rotation. The variability is associated to the matching step. As it can be seen in Table 1 the rotation computed from the μ parameter of homography is quite good, being the error in this case less than 0.1^o. However, the fundamental matrix can not give rotation correctly because it is badly conditioned (Table 1).

In case 2 (scene of Fig. 4.a,b) the motion is composed of 6 degrees of rotation and 24 cm. of translation, the problem with motion obtained from the fundamental matrix is similar, because the scene is nearly planar. As it can be seen in Table 1 the rotation computed from the μ parameter of homography is much better than that computed from the fundamental matrix.



(a) (b)

(c) (d)

Fig. 4. Automatically matched vertical lines to compute motion from two images of indoor scenes. (a, b) Nearly planar scene with 40 line matches being 100% correct. (c, d) Multi-plane scene with 20 line matches being 100% correct.

	Rotation from μ	Rotation from $\mathbf{F}$
Case 1	5.964 (0.012)	1.467 (4.651)
Case 2	6.569 (0.013)	-1.887 (5.839)
Case 3	5.554 (0.043)	5.878 (0.201)

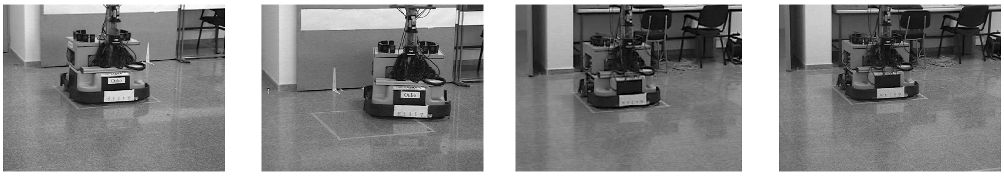
Table 1
Mean rotation computed (and standard deviation) in degrees with the three commented cases. The ground truth is in all cases 6^o.

In case 3 we use a different scene (Fig. 4.c,d) and the motion is also composed of 6 degrees of rotation and 24 cm. of translation. In this case, two planes appear in the scene and the fundamental matrix is better conditioned. As it can be seen in Table 1 the rotations computed from both methods are similar, being a bit better when computed from the fundamental matrix, but it uses many more features. The computational cost is similar with both methods being a bit smaller using μ. Currently, the computation time including the extraction of lines, the matching and the heading computation with homography is less than 200 milliseconds with a Pentium III on board.

We have made also a set of reruns rotating the camera with real images, to see the behavior of the μ parameter. We have tested that the evolution of the parameter as function of the rotation can be adjusted to a line of constant slope for small rotations ($\pm 10^o$). With the first scene (Fig. 4.a) we have experimented using $f = 12\,mm.$ and $f = 6\,mm.$ and with the second scene (Fig. 4.c) we have experimented using $f = 6\,mm.$ Table 2 shows the

	Residue (^{o})	Slope ($\mu/^o$)
scene Fig. 4.a, $f = 12\,mm.$	0.076	-0.0334
scene Fig. 4.a, $f = 6\,mm.$	0.056	-0.0174
scene Fig. 4.c, $f = 6\,mm.$	0.068	-0.0173

Table 2
Residue obtained in the linear fitting of μ to the rotation, and slope of the linear model which is the sole calibration parameter required.



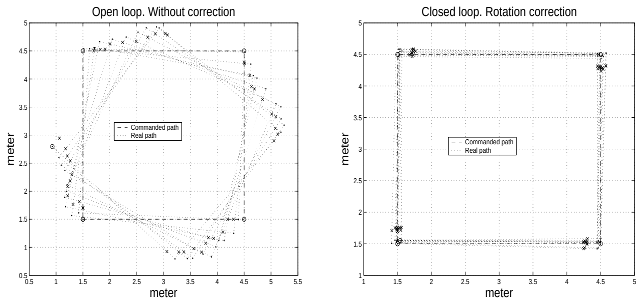
(a) (b) (c) (d)

Fig. 5. We show the start position to execute the path through a square without correction (a). In (b), we show the final position after 2 laps when there is no heading correction. In (c), we show the start position to execute the path through a square with correction and in (d) we show the final position after 10 laps when heading is corrected from μ.

residue of the linear fitting of the μ parameter to the camera rotation, which indicates the achievable accuracy. It is close to the resolution of the precision head used to obtain the ground truth. We have also checked how the working range is higher with lower focal length, being about $\pm 20^o$ with $f = 6\,mm$.

Besides the experiments to test the accuracy just shown, some experiments have also been carried out using our mobile robot (Fig. 5). For the experiment shown here, we use a camera with the image plane approximately vertical. In this case, the robot is commanded to go round a square of $3\,m$ of side. The trajectory is specified with 4 images taken at the corners of the square. In the playback phase a new image is taken at each corner which is compared with the previously stored image to correct robot motion. We show the start position and the position reached after 2 laps using only the odometry (Fig. 5.a,b). We also show the start position and the position reached after 10 laps with the heading correction in each corner from μ (Fig. 5.c,d). In this case, the error is not appreciable on view, because it is in the order of 1 centimeter after ten laps (120 meters of trajectory). Besides, the error will be the same if more laps are realized because it is not an accumulative error. To illustrate the path, a plant view of the trajectory with the positions reached at each corner of the square can be seen (Fig. 6).

Other experiment of autonomous navigation in real time is reported. Here the robot is commanded to go through a room moving 5 meters and making U-turns (rotation of 180^o) at each end. Using only odometry the robot deviates



(a) (b)

Fig. 6. Ten laps through a room without correction (a) and correcting the heading from the parameter μ (b).

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

Fig. 7. (a,b) Reference images of the start and end position for the experiment of U-turns. (c,d) Examples of current images to correct heading during the experiment.

and collides with the wall in three U-turns. Taking two reference images (Fig. 7.a,b) one at each end and correcting the heading with μ as proposed, the robot remains "forever" without appreciable deviations on the commanded trajectory. The current images during real time executions (Fig. 7.c,d) have an approximate disparity of 13% and the proposed method can match and compute rotation with image disparity in the range of 50% [4].

7 Conclusions

For teaching by doing in robotic tasks, we have argued the computation of visual motion based on homographies. These tasks, also referred to as homing, have many times been solved using the fundamental matrix, but the correction from homographies behaves better in many practical situations. Additionally, homography based on vertical lines is sufficient for robots moving indoors. Several experiments, in simulation and with controlled images, have been made to show the performance of homography parameters. A real time implementation for a mobile robot has been carried out with success. In this case, where enough reference images are stored, the motion correction obtained with the homography makes possible to repeat a previously learnt trajectory without drift problems.

References

[1] R. Basri, E. Rivlin, and I. Shimshoni. Visual homing: Surfing on the epipoles. In *IEEE Conference on Computer Vision*, pages 863–869, 1998.

[2] G.N. DeSouza and A. C. Kak. Vision for mobile robot navigation: A survey. *IEEE Trans. on Patt. Analysis and Machine Intelligence*, 24(2):237–267, 2002.

[3] J.J. Guerrero and C. Sagüés. Uncalibrated vision based on lines for robot navigation. *Mechatronics*, 11(6):759–777, 2001.

[4] J.J. Guerrero and C. Sagüés. Robust line matching and estimate of homographies simultaneously. In *IbPRIA, Pattern Recognition and Image Analysis, LNCS 2652*, pages 297–307, 2003.

[5] R. Hartley and A. Zisserman. *Multiple View Geometry in Computer Vision*. Cambridge University Press, Cambridge, 2000.

[6] Q.T. Luong and O. Faugeras. The fundamental matrix: Theory, algorithms, and stability analysis. *Int. Journal of Computer Vision*, 17(1):43–76, 1996.

[7] Y. Matsumoto, M. Inaba, and H. Inoue. Visual navigation using view-sequenced route representation. In *IEEE Int. Conf. Rob. and Autom.*, pages 83–88, 1996.

[8] J. M. Martinez Montiel and D. Ortin. Indoor robot motion based on monocular images. *Robotica*, 19:331–342, 2001.

[9] A. Ohya, Y. Miyazaki, and S. Yuta. Autonomous navigation of mobile robot based on teaching and playback using trinocular vision. In *Proceedings of the 27th IEEE Industrial Eletronics Conference (IECON'01)*, pages 398–403, 2001.

[10] O. Pellejero, C. Sagüés, and J.J. Guerrero. Automatic computation of fundamental matrix from matched lines. In *Current Topics in Artifical Intelligence, LNCS-LNAI 3040*, pages 197–206, 2004.

[11] P.J. Rousseeuw and A.M. Leroy. *Robust Regression and Outlier Detection*. John Wiley, New York, 1987.

[12] T. Tuytelaars, L. Van Gool, L. D́haene, and R. Koch. Matching of affinely invariant regions for visual servoing. In *IEEE Conference on Robotics and Automation*, pages 1601–1606, 1999.

[13] J. Weng, T.S. Huang, and N. Ahuja. *Motion and Structure from Image Sequences*. Springer-Verlag, Berlin-Heidelberg, 1993.

[14] Z. Zhang. Determining the epipolar geometry and its uncertainty: A review. *International Journal of Computer Vision*, 27(2):161–198, 1998.

[15] Z. Zhang, R. Deriche, O. Faugeras, and Q.T. Luong. A robust technique for matching two uncalibrated images through the recovery of the unknown epipolar geometry. *Artificial Intelligence*, 78:87–119, 1995.