Visual Homing: Surfing on the Epipoles

Ronen Basri*
Dept. of Applied Math
The Weizmann Inst. of Science
Rehovot 76100 Israel

Ehud Rivlin   Ilan Shimshoni†
Dept. of Computer Science
The Technion - Israel Inst. of Technology
Haifa 32000 Israel

Abstract

We introduce a novel method for visual homing. Using this method a robot can be sent to desired positions and orientations in 3-D space specified by single images taken from these positions. Our method determines the path of the robot on-line. The starting position of the robot is not constrained, and a 3-D model of the environment is not required. The method is based on recovering the epipolar geometry relating the current image taken by the robot and the target image. Using the epipolar geometry, most of the parameters which specify the differences in position and orientation of the camera between the two images are recovered. However, since not all of the parameters can be recovered from two images, we have developed specific methods to bypass these missing parameters and resolve the ambiguities that exist. We present two homing algorithms for two standard projection models, weak and full perspective. We have performed simulations and real experiments which demonstrate the robustness of the method and that the algorithms always converge to the target pose.

1 Introduction

Robot navigation and manipulation often involves the execution of commands which intend to move a robot (or a robot arm) to desired positions and orientations in space. A common way to specify such a command is by explicitly providing the robot with the three-dimensional coordinates of the desired position and the three parameters defining the desired orientation. This method suffers from several shortcomings. First, it requires accurate advance measurement of the desired pose. This is particularly problematic in flexible environments, such as when a robot is required to position itself near an object which may appear at different positions at different times. Secondly, due to occasional errors in measuring the actual motion of the robot, the robot may be unable to put itself sufficiently accurately in the desired position.

In this paper we propose a different approach to the problem of guiding a robot to desired positions and orientations. In our method the target pose is specified by an image taken from that pose (the *target image*). The task given to the robot is to move to a position where an image taken by a camera mounted on the robot will be identical to the target image. During the execution of this task the robot is allowed to take pictures of the environment, compare them with the target image and use the result of this comparison to determine its subsequent steps. We refer to the use of images to guide a robot to desired positions and orientations by *visual homing.*

We introduce a new method for visual homing. Our method differs from previous methods [2, 3, 5, 11, 12, 17, 18] in many respects. The method requires the pre-storage of the target image only. It then proceeds by comparing the target image to images taken by the robot, one at a time. No 3-D model of the environment is required, and the method requires no memory of previous images taken by the robot. Thus, the method uses minimal information and can deal also with a moving target. We present two homing algorithms for two standard projection models, weak and full perspective. The algorithms are based on recovering the epipolar geometry relating the current image taken by the robot and the target image. Correspondences between points in the current and target images are used for this purpose. (The problem of finding correspondences between feature points, however, is not addressed in this paper.) Using the epipolar geometry, most of the parameters which specify the differences in position and orientation of the camera between the two images are recovered. However, since not all the parameters can be recovered from two images, we develop specific methods to bypass these missing parameters and resolve ambiguities when such exist. The path produced by our algorithm is smooth

*Ronen Basri is an incumbent of Arye Dissentshik Career Development Chair at the Weizmann Institute.

†Ilan Shimshoni is supported in part by the Koret Foundation.

and optimal to the extent that is possible when only two images are compared. Furthermore, both simulations and real experiments demonstrate the robustness of the method and that the path produced by the algorithm always converges at the target pose.

2 Homing Under Weak-Perspective

2.1 Derivation

Our objective is to move the robot to an unknown target position and orientation S, which is given in the form of an image I of the scene taken from that position. At any given step of the algorithm the robot is allowed to take an image I' of the scene and use it to determine its next move. Denote the current unknown position of the robot by S', our goal then is to lead the robot to S.

WLOG we can assume that the pose S is the identity pose. Let $\mathbf{P}_i = (X_i, Y_i, Z_i)^T$, $1 \leq i \leq n$, be a set of n object points. Under weak-perspective projection, the image at the target pose is given by

$$x_i = X_i,\ y_i = Y_i.$$

A point $\mathbf{p}'_i = (x'_i, y'_i)^T$ in the current image I' is given by

$$\mathbf{p}'_i = [sR\mathbf{P}_i]_{(1,2)} + \mathbf{t},$$

where R is a 3×3 rotation matrix, s is some positive scale factor, $\mathbf{t} \in \mathbb{R}^2$ is the translation in the image, and $[.]_{(1,2)}$ denotes the projection to the first and second coordinates of a vector.

[6, 7, 9] showed that using at least four corresponding points in two images the epipolar constraints relating them can be recovered. From these constraints the scale can be derived. It can be verified that the translation component orthogonal to the epipolar lines can be recovered. The translation component parallel to the epipolar lines cannot be determined from this equation but is estimated using one pair of corresponding points. The estimate improves as the error in the viewing direction diminishes. For the rotation components it can be easily shown that every rotation in space can be decomposed into a product of two rotations, a rotation around some axis that lies in the image plane followed by a rotation of the image around the optical axis. The image rotation can be compensated for by rotating the epipolar lines in the current image until they become parallel to the epipolar lines in the target image. Differences in the viewing direction, however, cannot be resolved from two images. This is the reason why structure from motion algorithms that assume an orthographic projection require at least three images to recover all the motion parameters [6, 15].

Although two images are insufficient to resolve the differences in viewing direction completely, the axis of rotation required to bring the robot to the target pose can still be recovered from the images leaving the angle of rotation the only unrecoverable parameter. Knowing the axis of rotation will allow us to gradually rotate the robot until its viewing direction will coincide with the target viewing direction. In addition, the direction of rotation is subject to a twofold ambiguity; namely, we cannot determine whether rotating to the right or to the left will lead us faster to the target orientation. In [1] we show that the axis of rotation is orthogonal to the epipolar lines. Thus, the possible viewing directions lie on a great circle on the viewing sphere which passes through the viewing directions of the target and current images. Therefore by rotating the camera parallel to the direction of the epipolar lines we can compensate for the differences in the viewing direction.

2.2 Resolving the ambiguity

We have been able to determine the great circle on the viewing sphere along which the robot should rotate. However, we have not determined which direction on the circle is the shorter of the two directions connecting the current and target viewing directions.

To resolve this ambiguity we introduce a similarity measure that can be applied to the current and target images. While the robot is changing its viewing direction along the great circle we will evaluate the similarity between the images and see whether they become more or less similar. Using this information we will be able to determine if the robot is changing its viewing direction along the shortest path to the target viewing direction, or if it is rotating in the other direction, in which case we can correct its rotation. This similarity measure should vary with a change in the viewing direction, but be invariant to scale changes, translation, and image rotation.

The measure of similarity we have chosen is based on the apparent angles formed by triplets of points in the current and target images. Figure 1 shows several examples of how apparent angles change as the viewing direction moves on a great circle. Given an angle Φ in the scene and a great circle on the viewing sphere we denote the apparent angle as a function of the angle on the great circle θ by $\phi(\theta)$. $\phi(\theta)$ has the following characteristics: it is a periodic function whose period is 2π. Furthermore, $\phi(\theta) = -\phi(\theta + \pi)$. Also, $\phi(\theta)$ has a single maximum at some angle, θ_{max}, and a single minimum, obtained at $\theta_{min} = \theta_{max} + \pi$. Finally, each angle between the maximum and minimum appears exactly twice in the function.

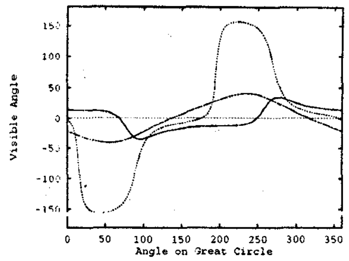

Figure 1: Three examples showing the effect of changing the viewing direction along a great circle on projected angles.

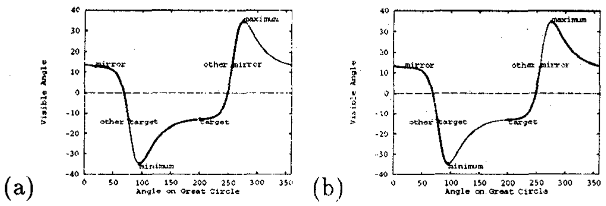

Figure 2: "Good" (thick lines) and "bad" (thin lines) sections with respect to the desired angle at the target (left) and mirror (right) images obtained while moving along a great circle on the viewing sphere.

Our measure of similarity is based on testing whether the apparent angles seen in the images taken by the robot are approaching the corresponding angles in the target image. In identifying the correct direction several issues have to be addressed. First, there exists a second viewing direction on the great circle which gives rise to the same apparent angle as in the target image. We call this direction a *false target*. Fortunately, it is not difficult to distinguish between the target and the false target because every angle in the scene gives rise to a different false target. Secondly, there exist "good" and "bad" sections on the great circle, where a section is considered "good" if when rotating in the correct direction along this section the apparent angle approaches its value in the target image.

Figure 2(a) shows an example of $\phi(\theta)$. The thick segments denote the "good" sections of the great circle, and the thin segments denote the "bad" sections of the great circle. It can be seen that a "good" section exists around the target viewing direction, and as we get further away from the target "bad" sections appear. Consequently, suppose we consider the apparent angles in the current image, count how many of them approach the target, and use majority to decide on the correct direction then we are likely to make a correct choice near the target viewing direction. Our chances to be correct, however, deteriorate as we get away from the target.

We therefore define a similar measure for the mirror image. Again, the great circle can be divided to "good" and "bad" sections, where now "good" sections are sections in which walking in the wrong direction will make the apparent angle approach the mirror image (Fig. 2(b)). This measure is likely to point to the correct direction in the neighborhood of the mirror image.

Since each of the two measures, the similarity to the target and mirror images, are reliable in different parts of the great circle we would like to use each of them at the appropriate range of viewing directions. We do so by checking which of the two measures achieves a larger consensus. The rationale behind this procedure is that for every angle in the scene each of the measures covers more than half of the great circle.

To check the quality of our decision procedure we tested the procedure on a 1000 great circles chosen at random. For each circle 1000 point triplets were chosen at random. We plotted in Fig. 3 the average percentages $\mu_t(\theta)$ and $\mu_m(\theta)$ of target and mirror angles respectively which point to the correct direction. In order to show the standard deviation of those functions $\sigma_t(\theta)$ and $\sigma_m(\theta)$, we plotted $\mu(\theta) \pm \sigma(\theta)$. In these plots the target and mirror angles are equal to $0°$ and $180°$ respectively. Note that for every θ along the great circle $\max(\mu_t(\theta), \mu_m(\theta)) > 0.5$. Therefore, the decision as to which way to go is determined by finding which direction is supported by more angles by one of the two similarity measures.

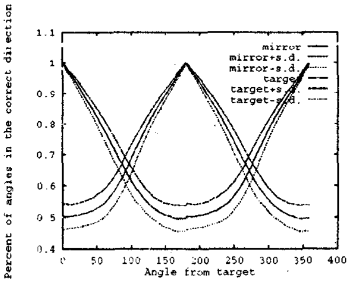

Figure 3: The plot shows the percent of angles (and standard deviations) which point in the correct direction for a given viewing direction on the great circle using the target and the mirror angles

We have shown how we can estimate the motion parameters which separate the current pose of the robot from the target pose. The rotation of the image has been recovered completely. For the translation components in the image plane we have an estimate. However the rest of the parameters, the translation in depth (indicated by a scale change) and the change in the viewing direction were estimated only as a direction, while their magnitude, the distance in depth between the two images and the angular separation between the viewing directions were not determined. In the rest of this section we show how we can estimate the missing distances to the target pose. Estimating these missing distances will enable the robot to perform a smooth motion to the target by combining at every step a similar fraction of each of the motion components.

We begin by deriving the component of translation along the optical axis from the scale changes. Suppose the scale between the current and the target image is given by s, and suppose that at the following step the scale becomes s'. It can be easily shown that the number of steps of this size is $n = s'/(s - s')$.

We estimate the angular separation between the current and target viewing directions using a Maximum Likelihood estimator of this angle which uses the percentage of angles which point to the correct direction (Figure 3). Details can be found in [1].

3 Full Perspective Homing

In this section we consider the problem of homing under perspective projection. Below we describe our method for homing when the focal length of the camera is known. For this case we show how the motion parameters can be recovered, and develop methods to resolve the ambiguity in the direction and recover the distance to the target position. In [1] we extend this formulation to the case that the focal length is unknown.

3.1 Homing with a known focal length

Again, we wish to move a robot to an unknown target position and orientation S, which is given in the form of an image I of the scene taken from that position. At any given point in time the robot is allowed to take an image I' of the scene and use it to determine its next move. Denote the current unknown position of the robot by S', our goal then is to lead the robot to S. Below we assume that the same camera is used for both the target image and images taken by the robot during its motion, and that the internal parameters of the camera are all known. The external parameters, that is, the relative position and orientation of the camera in these pictures is unknown in advance.

To determine the motion of the robot we would like to recover the relative position and orientation of the robot S' relative to the target pose S. Given a target image I taken from S and given a second image I' taken from S', by finding sufficiently many correspondences in the two images we estimate the motion parameters using the algorithm described in [4, 16], which is based on the linear algorithm proposed in [10, 14]. This algorithm requires at least eight correspondences in the two images. Other, non-linear approaches can be used if fewer correspondences are available [8].

The algorithm proceeds by first recovering the essential matrix E relating corresponding points in the two images. Once the essential matrix is recovered, it can be decomposed into a product of two matrices $E = RT$, the rotation matrix R and a matrix T which contains the translation components. The rotation matrix, which determines the orientation differences between the two images, can be fully recovered. The translation components, in contrast, can be recovered only up to an unknown scale factor. These recovered translation components determine the position of the epipole in the current image, which indicates the direction to the target position. In the next section we show how to determine whether the target position is in front or behind the current position of the robot. However we cannot determine the distance to the target position.

After we recover the motion parameters we direct the robot to move a small step in the direction of the target. In addition, given the rotation matrix R we calculate the axis and angle of rotation that separates the current orientation of the robot from the target orientation and rotate the robot arm about this axis by a small angle. After performing this step the robot takes a second image. Using this image we recover the distance to the target position and use this distance to perform a smooth motion.

3.2 Resolving the ambiguity in the direction to the target

We have seen so far how given the current and target image the translation required to take the robot to the target position is indicated by the position of the epipole in the current image. However, using the epipole the direction to the target can be recovered only up to a twofold ambiguity, namely, we know the line which includes the two camera positions, but we do not know whether we should proceed forward or backward along this line to reach the target position. Below we show how by further manipulating the two images we can resolve this ambiguity.

Using the current and target images we have completely recovered the rotation matrix relating the two images. Since a rotation of the camera is not affected by depth we may apply this rotation to the current image to obtain an image that is related to the target image by a pure translation. After applying this rotation the two image planes are parallel to each other and the epipoles in the two images fall exactly in the same position. Denote this position by $(v_x, v_y, f)^T$. We may now further rotate the two image planes so as to bring both epipoles to the position $(0, 0, f)^T$. Denote this rotation by R_0. Notice that there are many different rotations that can bring the epipoles to $(0, 0, f)^T$, all of which are related by a rotation about $(0, 0, f)^T$. For our purpose it will not matter which of these rotations is selected.

After applying R_0 to the two images we now have the two image planes parallel to each other and orthogonal to the translation vector. The translation between the two images, therefore, is entirely along the optical axis. Denote the rotated target image by I and the rotated current image by I'. Relative to the rotated target image denote an object point by $P = (X, Y, Z)$. Its coordinates in I are given by

$$x = \frac{fX}{Z}, \qquad y = \frac{fY}{Z}$$

and its corresponding point $(x', y', f)^T \in I'$,

$$x' = \frac{fX}{Z+t}, \qquad y' = \frac{fY}{Z+t}.$$

t represents the magnitude of translation along the optical axis, and its sign is positive if the current position is in front of the target position, and negative if the current position is behind the target position. We can therefore resolve the ambiguity in the direction by recovering the sign of t. To do so we divide the coordinates of the points in the target image with their corresponding points in the current image, namely

$$\frac{x}{x'} = \frac{y}{y'} = \frac{Z+t}{Z} = 1 + \frac{t}{Z}.$$

This implies that

$$t = Z(\frac{x}{x'} - 1).$$

Unfortunately, the magnitude of Z is unknown. Thus, we cannot fully recover t from two images. However, its sign can be determined since

$$sign(t) = sign(Z) sign(\frac{x}{x'} - 1).$$

Notice that since we have applied a rotation to the target image Z is no longer guaranteed to be positive. However, we can determine its sign since we know the rotation R_0, and so we can determine for every image point whether it moved to behind the camera as a result of this rotation. Finally, the sign of $x/x' - 1$ can be inferred directly from the data, thus the sign of t can be recovered. Since it is sufficient to look at a single pair of corresponding points to resolve the ambiguity in the translation we may compute the sign of t for every pair of corresponding points and take a majority to obtain a more robust estimate of the actual direction.

3.3 Recovering the distance to the target

To estimate the distance to the target position we let the robot move one step and take a second image. We then use the changes in the position of feature points due to this motion to recover the distance.

Using the current and target images we have completely recovered the rotation matrix relating the two images. Since a rotation of the camera is not affected by depth we may apply this rotation to the current image to obtain an image that is related to the target image by a pure translation. Below we refer by I' and I'' to the current and previous images taken by the robot after rotation is compensated for so that the image planes in I, I', and I'' are all parallel.

We begin by observing that any two images related purely by a translation give rise to the same epipolar lines. Given an image I and a second image I' which is obtained by a translation by $\mathbf{t} = (t_x, t_y, t_z)^T$, notice first that the two images have their epipoles in the same position. This is because the homogeneous coordinates of the epipole in I' are identical to $\mathbf{t}$, while the homogeneous coordinates of the epipole in I are identical to $-\mathbf{t}$. Consider now a point $(x, y, f)^T \in I$, and its corresponding point $(x', y', f)^T \in I'$,

$$x' = \frac{f(X + t_x)}{Z + t_z}, \qquad y' = \frac{f(Y + t_y)}{Z + t_z}.$$

Denote the epipole by $(v_x, v_y) = (ft_x/t_z, ft_y/t_z)$, it can be readily shown that both (x, y) and (x', y') lie on the same line through (v_x, v_y), since

$$\frac{x' - v_x}{y' - v_y} = \frac{x - v_x}{y - v_y}.$$

We turn now to recovering the distance to the target position. Given a point $\mathbf{p} = (x, y, f)^T \in I$, suppose the direction from the current image I' to the target position is given by $\mathbf{t} = (t_x, t_y, t_z)^T$, and that between the previous image I'' and the current image the robot performed a step $\alpha\mathbf{t}$ in that direction. Denote by n the remaining number of steps of size $\alpha\mathbf{t}$ separating the current position from the target (so that $n = 1/\alpha$). The x coordinate of a point in the target, current, and previous images are

$$x = \frac{fX}{Z}, \quad x' = \frac{f(X + t_x)}{Z + t_z}, \quad x'' = \frac{f(X + (1+\alpha)t_x)}{Z + (1+\alpha)t_z}$$

respectively. Eliminating X and Z and dividing by t_z we obtain that

$$n = \frac{(x' - x)(x'' - v_x)}{(x'' - x')(x - v_x)}.$$

The same computation can be applied to the y coordinate of the point. In fact, we can obtain a better recovery of n if we replace the coordinates by the position of the point along the epipolar line in the three images. (Thus, n is obtained as a cross ratio along this line.)

Even though a single corresponding point is sufficient to determine the distance to the target position we can combine the information obtained from all the points to obtain a more robust estimate of the distance. Notice that this computation will amplify noise in the image when either $|x'' - x'|$ or $|x - v_x|$ are small. Thus, the values obtained for points which introduce a significant change in position between the previous and current images and which their position in the target image is further away from the epipole are more reliable then points which introduce only a small change or points which appear close to the epipole in the target image.

4 Experimental results

We have tested our homing algorithm under weak perspective on a thousand initial poses chosen at random. The algorithm converged successfully in all cases. Figure 4 shows the effect of uncertainty in the vertex position measured in the image on the convergence of the algorithm. Figure 4(a) shows how the error in all the components of the pose converge to zero when there is no uncertainty. In Figure 4(b) the effect of uncertainty is shown. The uncertainty only effects the final stages of the algorithm when the error is very small. The algorithm converges more slowly until a solution is found.

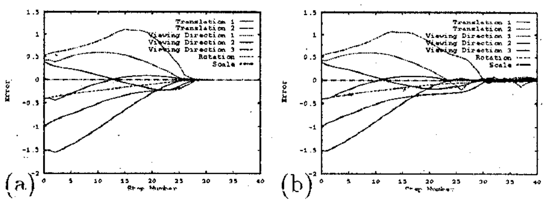

Figure 4: The convergence of the components of the pose as the algorithm progresses. The pose is composed of seven components: the three Euclidean coordinates of the viewing direction, two components of the translation, the scale factor, and the image rotation. (a) No noise; (b) Noise level of 1%.

Figure 5 shows an example of applying the perspective procedure to simulated data in a noise-free and a noisy environment. As can be seen, in the noise-free example, the robot moved in the shortest path to the target while changing its orientation gradually until it matched the target orientation. Notice that since at the first step the robot could not yet estimate its distance to the target its first rotation differed from the rest of the rotations.

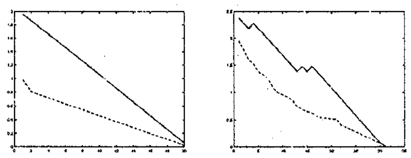

Figure 5: A simulation of homing under perspective projection. The solid line represents the distance of the robot from the target position, and the dashed line represents the angle separating the current orientation from the target orientation. Left: no noise. Right: Gaussian noise added to the pixel positions at every image.

Finally, we mounted a CCD camera on a robot arm (SCORBOT ER-9, from Eshed Robotec Inc.). The arm was set in a target position and an image was taken (target, see Fig. 6(f)). The arm was then set in another position, from which part of the target scene was visible (source, see Fig. 6(a)). The correspondences between the source and the target was provided manually. Then, the algorithm described in Section 2 was run. We maintained correspondences between successive frames by tracking the points using a correlation based tracking algorithm. We took twenty features so that we can afford losing some of the features along the way (because of noise and occlusion) without impairing our ability to recover the epipolar constraints. In computing the epipolar lines at every step of the algorithm we used at least ten corresponding points using and applied a least squares fit.

The different steps of the experiment are shown in Fig. 6(a)(h), where (a) is the source image, b-h the intermediate steps. The final image is shown in Fig. 7(a), note the similarity between it and the target image shown to its right(b). The joint values of the robot in its final position after the homing was completed were different, from target joint values, by less than 1° for the five revolute joints, and by less than 1cm for the linear shift bar.

5 Conclusions

In this paper we have introduced a novel method for visual homing. Using this method a robot can be sent to desired positions and orientations specified by images taken from these positions. The method requires the pre-storage of the target image only. It then proceeds by comparing the target image to images taken

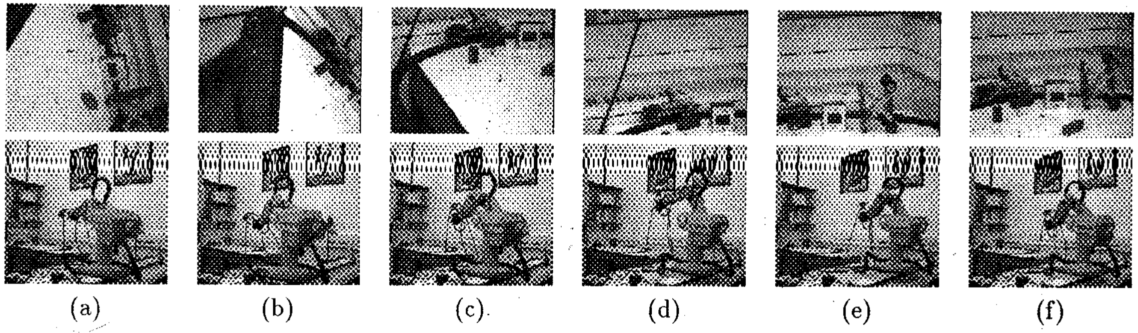

Figure 6: A run of an experiment with a six degrees of freedom robot: (a) The initial image; (b-f) Intermediate images; Top images: the images seen by the robot. Bottom: the position of the robot taken from a fixed camera.

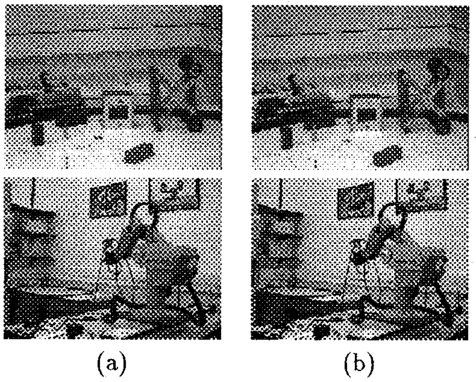

Figure 7: (a) The final image after homing was completed; (b) The target image. Top images: the images seen by the robot. Bottom: the position of the robot taken from a fixed camera.

by the robot while it moves, one at a time. Unlike existing approaches, our method determines the path of the robot on-line, and so the starting position of the robot is not constrained. Also, unlike existing methods, which are largely restricted to planar paths, our method can send the robot to arbitrary positions and orientations in 3-D space. Nevertheless, a 3-D model of the environment is not required. Finally, our method requires no memory of previous images taken by the robot. Thus, the method uses minimal information and can deal also with a moving target.

References

[1] R. Basri, E. Rivlin, and I. Shimshoni, Visual homing: surfing on the epipoles *Department of Computer Science Technical Report CIS9709, The Technion.*

[2] R. Basri and E. Rivlin, Localization and homing using combinations of model views. *AI* **78**:327–354, 1995.

[3] G. Dudek and C. Zhang, Vision-based robot localization witout explicit object models. *IEEE Int. Conf. on Robotics. and Automation*:76–82, 1996.

[4] R. Hartley. In defence of the 8-point algorithm. *ICCV-95*:1064–1070, 1995.

[5] J. Hong, X. Tan, B. Pinette, R. Weiss, and E. M. Riseman. Image-based homing. *IEEE Control Systems*:38–44, 1992.

[6] T. S. Huang and C. H. Lee, Motion and Structure from Orthographic Projections. *PAMI*, **2**(5):536–540, 1989.

[7] L. L. Kontsevich, Pairwise comparison technique: a simple solution for depth reconstruction. *Journal of Optical Society*, **10**(6):1129–1135, 1993.

[8] E. Kruppa, Zur Ermittlung eines Objektes aus zwei Perspektiven mit innerer Orientierung. *Sitz.-Ber. Akad. Wiss., Wien, Math. Naturw. Kl., Abt. IIa.*, **122**:1939-1948, 1913.

[9] C. H. Lee and T. S. Huang, Finding point correspondences and determining motion of a rigid object from two weak perspective views. *CVGIP*, **52**:309–327, 1990.

[10] H. C. Longuet-Higgins, A computer algorithm for reconstructing a scene from two projections. *Nature*, **293**:133–135, 1981.

[11] Y. Matsumoto, I. Masayuki and H. Inoue, Visual navigation using view-sequenced route representation. *IEEE Int. Conf. on Robotics and Automation*:83–88, 1996.

[12] R. N. Nelson. Visual homing using an associative memory. *DARPA Image Understanding Workshop*:245–262, 1989.

[13] L.S. Shapiro, A. Zisserman, and M. Brady. 3d motion recovery via affine epipolar geometry. *IJCV*, 16(2):147–182, October 1995.

[14] R. Y. Tsai and T. S. Huang, Uniqueness and estimation of three-dimensional motion parameters of rigid objects with curved surfaces. *PAMI*, **6**(1):13–27, 1984.

[15] S. Ullman. The interpretation of visual motion. *M.I.T. Press*, Cambridge, MA, 1979.

[16] J. Weng, T.S. Huang, and N. Ahuja. Motion and structure from two perspective views: Algorithms, error analysis, and error estimation. *PAMI*, **11**(5):451–476, 1989.

[17] J. Y. Zheng and S. Tsuji. Panoramic representation for route recognition by a mobile robot. *IJCV*, **9**(1):55–76, 1992.

[18] D. Zipser. Biologically plausible models of place recognition and goal location. In *D. E. Rumelhart et al., Parallel Distributed Processing: Explorations in the Microstructure of Cognition*, **2**, MIT Press:432–471, 1986.