Mobile Robot Navigation using Omni-directional Vision

Niall Winters

Computer Vision and Robotics Group,
Department of Computer Science,
University of Dublin, Trinity College,
Dublin 2 - Ireland.

Niall.Winters@cs.tcd.ie

José Santos-Victor

Instituto de Sistemas e Robótica,
Instituto Superior Técnico,
Av. Rovisco Pais, 1,
1049-001 Lisboa - Portugal.

jasv@isr.ist.utl.pt

Abstract. This paper addresses the problem of autonomous mobile robot navigation in a simple indoor environment. We present a system which combines the paradigm of visual servoing with appearance based methods. The only sensory information available to the robot comes from an omni-directional vision camera. The geometry of our system, which uses a spherical mirror is presented. Our algorithm relies on a sequence of images obtained *a priori*. Since this image set is relatively large, it is compressed using Principal Component Analysis (PCA), thus producing a low dimensional eigenspace. This eigenspace maps the environment in a *topological* sense. Hence, from this the robot can determine its *global* topological position along a given path. In order to control *local* pose, visual servoing is applied to ground plane dewarped images. The method of obtaining these images and the closed-loop control mechanism, which is reliant upon them, is detailed. Preliminary results from both simulation and real world experiments are presented, as is a method of extending the system.

1 Introduction

The 'Holy Grail' for mobile robotic research is to design a robot that can function autonomously in a real world environment. This real world environment includes both indoor and outdoor scenes. Current research is delineated along these lines: methods that work well for the former situation may not do so for the latter. Hence, we will confine ourselves to the goal of mobile robot operation in an indoor environment. The overall aim of our research is to confer on a mobile robot goal-driven motion within this environment. We desire that the robot navigate autonomously to a specified destination. We place the additional constraint on the system that it can only use a single camera for sensing. In order to reach its final destination the robot may also, but not necessarily have to visit a number of sub-destinations. Thus, we require a relevant representation of the environment and a robust method of controlling the robot's pose. This paper presents, what we believe to be an appropriate solution to this problem: the combination of appearance-based methods and visual servoing.

In the past, non-visual sensors have been used to build up geometric representations of the environment. These are highly complex, with the input coming from a number of sensors (sonar, laser, etc.) which is then combined, using a suitable filter, into an environmental map. The main criticism of this approach is that algorithms using the map assume an accurate metric representation in a global co-ordinate system. In practice, this is problematic to achieve, as the map usually contains irrelevant information, thus adding to the computational load. In addition, these systems are slow and not very suitable for real-time navigation.

To overcome the need for multiple sensors, early research into the navigation of a mobile robot using vision centered on the 3D reconstruction of a scene. While 3D reconstruction is accurate it is also time consuming and so not suitable for real-time navigation. Active vision [1] takes the approach of altering the robot's viewpoint rather than passively observing it. If the structure of a specific scene is taken into account *customized* algorithms can be used at the local level. Unfortunately, by definition, they have no global map and so encounter difficulties with global tasks.

Appearance based methods [8] [13] offer an alternative solution to the problem and have been an active area of research in the recent past. Progression towards the goal can simply be calculated by comparing the current view with images acquired *a priori*. Localisation [7] has been achieved by matching images using the frequency domain of the Fourier Transform. Horswill [6] used the actual views as landmarks. Zheng [15] took the approach of using dynamic programming to determine which path to take from a number of choices, even though this had a very high matching cost. Neilson [11] concentrated on the translation of database images into motion commands. The work most closely related to ours is that of Vassallo [14]. She too combined appearance based methods and visual servoing, although her images, matching scheme and method of servoing were different from those detailed in this paper. With our approach, omni-directional images are used as an implicit topological representation of the environment, i.e. our global map. The map is represented as a manifold in a low dimensional eigenspace obtained from Principal Component Analysis (PCA).

It is often the case that the topological map can easily be associated with some qualitative structural elements of the environment. These can be used to control locally the robot trajectory. In this paper, the structural elements we choose to servo upon are the corridor guidelines. These are extracted from ground plane dewarped images. Thus the robot maintains its central position, correcting both orientation and position errors. Other examples of this local servoing scheme could include centering the robot relative to a door frame, orienting the robot relative to a specific object to grasp, etc. We show that the methods of extracting the image signals, which drive the closed loop, do provide meaningful information for control purposes.

This paper is outlined as follows: Section 2 presents a brief overview of our system. Section 3 details the geometry of an omni-directional camera using a *spherical* mirror while section 4 presents eigenspace construction and position

estimation. It also details how eigenspaces can be utilised as topological maps. In section 5 our control scheme and method of tracking corridor guidelines is detailed, then in section 6 preliminary experimental results obtained are presented. Finally, in Section 7 we draw our conclusions and outline the future direction of our research.

2 System Overview

As shown in Figure 1(a) our system is a TRC Labmate from HelpMate Robotics Inc. Mounted upon this is an omni-directional vision system (Figure 1(b)) built

(a) (b)

Fig. 1. (a) Labmate mobile robot with attached omni-directional camera. (b): Close-up of the camera.

in-house. It contains a Cohu CCD camera pointed upwards, looking at a spherical mirror. Grayscale images were captured using a Captivator frame grabber at full resolution, 768x576 pixels. These were sub-sampled to 128x128 images for PCA and 600x600 for visual servoing. The images obtained are of three types: warped panoramic, ground plane dewarped and dewarped panoramic. These three types are shown in Figure 2.

As the robot moves down the corridor it captures two images, one warped panoramic and one ground plane dewarped. The warped panoramic image is passed to the appearance based system, so as the robot can determine how far it has progressed along the path to its goal. The ground plane image is passed to the visual servoing system. The sequence is as follows:

- Determine the topological position by projecting the current warped omni-directional image into the eigenspace.
- Edge detect within bounding boxes in the ground plane image to extract the corridor guidelines.

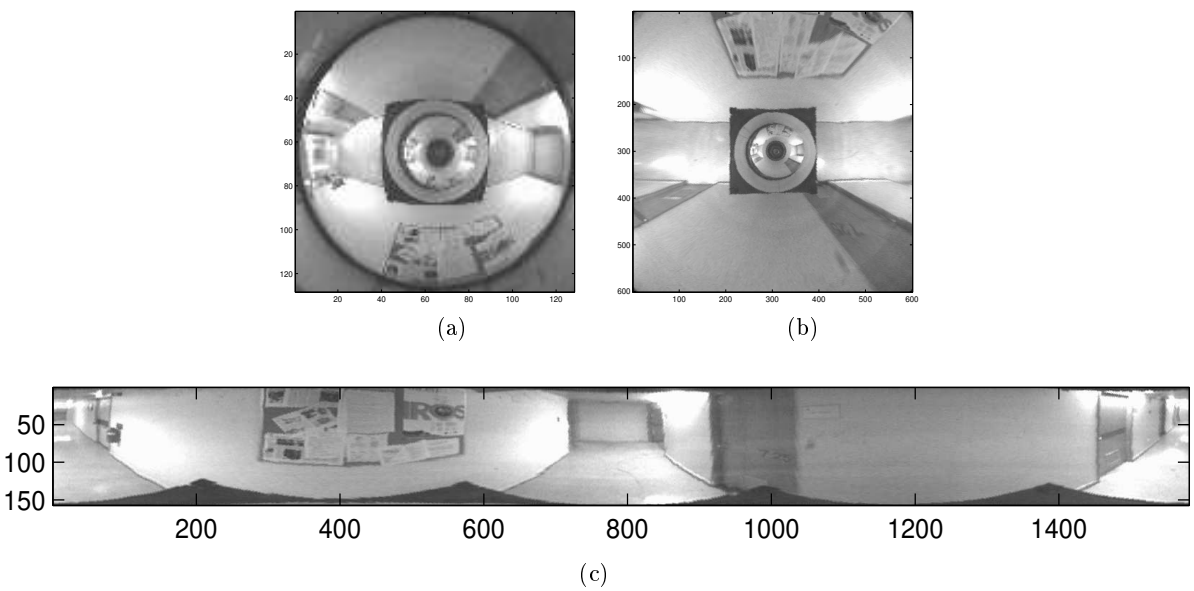


Fig. 2. The three types of images that can be obtained from the omni-directional vision system: (a) warped panoramic, (b) ground plane dewarped and (c)dewarped panoramic.

- Calculate the robot's orientation and local position, relative to the centre of the corridor.
- Generate a reference angular velocity command in order to minimize the robot's orientation and position errors.
- Relying on the appearance based system, make decisions at specific locations, e.g. turn left at the end of the corridor.

3 Omni-directional Vision

Omni-directional vision [2] [12] is the name given to the generation of panoramic images from a single camera. Images are obtained by placing a convex mirror a short distance above a camera as shown in Figure 1(b). A system provides a 360° view of the robot's environment around the vertical in addition to $\sim 30°$ above the horizontal. They have becoming popular over the last number of years and are now used in many applications including autonomous navigation and video conferencing. Their many advantages include:

- Landmarks are always in the field of view, except for occasional occlusion. This is advantageous when utilizing topological maps in that the more information the image contains the more stable it is.
- Panoramic views are acquired in a *single frame*.

– They have no moving parts, thus consume less power, vis-à-vis a pan-tilt mechanism and therefore have increased reliability.
– Circles perpendicular to the axis of rotation are invariant under rotation.

Their main downfall is that images acquired are of lower resolution than standard. Therefore, they are not suitable in situations where high resolution is a prerequisite, personal identification for example.

Many different shapes of mirror can be used for building a system: spherical, parabolic or hyperbolic. Hyperbolic mirrors have the advantage of a single centre of projection. Thus, if one wanted to obtain multiple perspective views of the environment this mirror would be the idea choice. Since this is not our stated goal, we chose a system with a spherical mirror. Figure 3 shows the geometry of our system. We wish to determine the *image co-ordinates* of a real world 3D point.

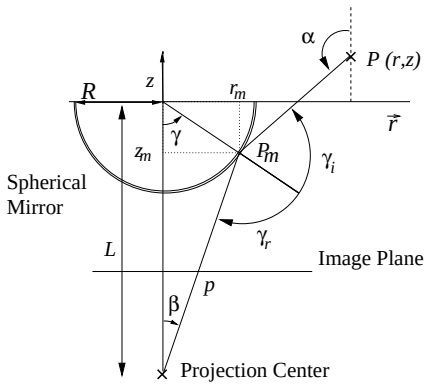


Fig. 3. Spherical mirror geometry.

To achieve this, we must first determine the position of a 3D point on the mirror surface and then find its projection onto the image plane. A 3D point in the world can be expressed in cylindrical coordinates as

$$P = (\varphi, r, z) = (artan(y/x), \sqrt{(x^2 + y^2)}, z)$$

and we define a point on the mirror as

$$P_m = (r_m, z_m)$$

Notice that knowing P_m allows us to find β. Also, α can be written as

$$\alpha = \arctan(\frac{z - z_m}{r - r_m}) + \frac{\pi}{2}$$

P_m can be thus be found by solving the following non-linear equations:

$$z_m = r.\tan(\frac{\pi}{2} - \beta) - L$$

$$z_m^2 + r_m^2 = R^2$$

$$2\arctan(\frac{r_m}{-z_m}) = \alpha - \beta \; since \; \gamma_i = \gamma_r$$

Now, to find a point in the image, $p = (u, v)$ we use a standard perspective projection model and our knowledge of the intrinsic parameters of the camera. Thus, we obtain:

$$(u, v) = f \; \tan\beta \; (\cos\varphi, sin\varphi)$$

4 Eigenspace Construction & Position Estimation

4.1 Overview of our Method

Our approach to constructing an eigenspace representation of the environment is based on the method by [10]. A large image set is acquired and compressed using Principal Component Analysis. The eigenvectors of the covariance matrix of the set of images forms an orthonormal basis. These are the principal components. There are as many basis vectors as there are images. For example, if we have 100 images, then we are dealing with a 100D space. All the eigenvectors contain components from all the images and so are sometimes referred to as *eigenimages*. If exact classification is required, all eigenvectors need to be utilized. In fact, for almost all situations this is not the case. Hence, we need only use a small number (typically ten to twelve) of these eigenvectors as an orthonormal basis for a *lower* dimensional space. Which eigenvectors to keep is determined by their associated eigenvalues. For our experiments the first ten to twelve eigenvectors are enough. Position is estimated by projecting the most recently captured (warped omni-directional) image into the low dimensional eigenspace. The closest point, on the manifold, to this projected point is the best estimation of position.

4.2 Eigenspace Construction

This subsection outlines the method used to construct the eigenspace. N images are read in a line-scan manner into an array $\mathbf{X}$, normalized and converted from 128 x 128 matrices to 16384 x 1 column vectors. The average of all the images, $\mathbf{a}$ is determined and subtracted from all N images thus giving a matrix $\mathbf{B}$. The eigenvectors, $\mathbf{e}$ of the covariance matrix, $\mathbf{R} = \mathbf{B}\mathbf{B}^\mathbf{T}$ form the orthonormal basis

we require. Unfortunately, computing them is highly computationally intensive. Fortunately, a solution [9] to the problem lies in computing the singular value decomposition (SVD) of $\tilde{\mathbf{R}} = \mathbf{B^T B}$. Thus, we obtain singular vectors $\tilde{\mathbf{e}}$ and values $\tilde{\lambda}$ which are then converted to the required eigenvectors and eigenvalues by the equations:

$$\lambda_i = \tilde{\lambda}_i$$

$$\mathbf{e}_i = \tilde{\lambda}_i^{-1/2} \mathbf{B} \tilde{\mathbf{e}}_i.$$

In the next stage a lower dimensional subspace is built using only the first M, ($M \ll N$) eigenvectors. This lower dimensional eigenspace has the advantage of accounting for most of the variance in the images. Each image, minus the average, is projected into this space by the following dot product to obtain a set $\mathbf{T}$ of projected points:

$$\mathbf{T}_i = \mathbf{e}_i^T . (\mathbf{X}_i - \mathbf{a})$$

Each of the successively projected points are connected by a linear interpolation forming a manifold in the eigenspace as shown in Figure 4. Points in the eigenspace which are closely related correspond to similar images in terms of the $l^2 norm$. It should be noted that a 3D eigenspace is shown for illustrative purposes only and represents one corridor.

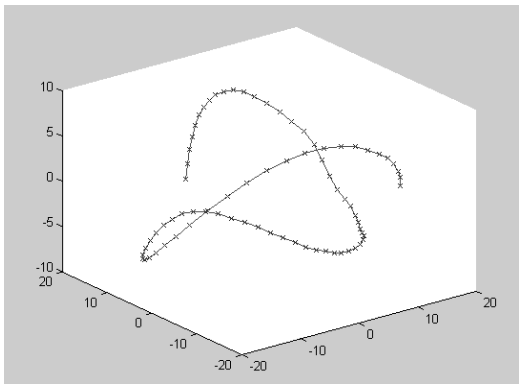


Fig. 4. The projected points connected together forming a manifold in the M=3 eigenspace.

4.3 Position Estimation

An unknown image representing the robot's current position is projected into the eigenspace forming a point, $\mathbf{c}$. Since it is similar to previously acquired images it will lie close to the manifold but not on it. Which image it is closest to, i.e.

the robot's current topological position is calculated as follows:

$$\mathbf{c} = \frac{\sum_{i=1}^{m} (\mathbf{P}_i^2 \lambda_i)}{\sum_{i=1}^{m} \lambda_i}$$

where $\mathbf{P}$ holds the distances, along each dimension in the eigenspace, between the projected point (representing the newly acquired image) and each point (representing the *a priori* data) on the manifold. λ_i is the associated eigenvalue.

4.4 Eigenspaces as Topological Maps

The eigenspace we construct can be viewed as a form of topological map. This is the method we used to characterise the global environment. In terms of appearance based methods a point in the eigenspace can represent, qualitatively, the position of the robot. As stated previously, each point corresponds to an image, a database of which we obtained *a priori*. These points form our map. For example, a single corridor can be represented by a sequence of images (points). This sequence of points forms a manifold in the eigenspace. This manifold can then be defined as a *link* in a graph structure. A number, N $\geq$ 2, of such links are connected via *nodes*. Nodes therefore represent specific places in the real world where actions (such as turn right, for example) can be taken. Thus nodes include corridor intersections, entrances to offices, exits etc. Whenever the robot reaches a nodal point the required action can easily be launched.

Topological maps offer a parsimonious representation of the environment in that they are not complex and scale easily. One subgraph can always be connected to another, thus allowing for the autonomy of the robot to be extended in a relatively unlimited manner.

5 Line Tracking & Robot Control

5.1 Ground Plane Dewarped Images

In order to control locally the position and orientation of the robot, we wish to visually servo upon the corridor guidelines. Since they are warped in the original omni-directional image this task is complex. Hence, we take the alternative approach of dewarping the images to a ground plane view *before* servoing. Thus the servoing task is made much simpler as now the world-to-image Jacobian is an exact linear transform. In other words, the 2D projections of 3D straight lines on the floor are themselves straight.

To achieve the required mapping it is necessary to minimize a cost function, $\mathcal{F}$ which compares the projection of the current image with known projection data. This known data is obtained from the equations detailed in [4]. Once the mapping has been calculated, all images can then be remapped using a look up table, thus increasing the speed of dewarping. Ground plane dewarped images can be seen in Figure 7.

5.2 Robot Control

To achieve stable control, visual feedback from the omni-directional camera is used. This feedback is the robot's current heading and displacement obtained from the extracted corridor guidelines. The control law used is as follows:

$$\theta_{i+1} = \theta_i - \mathcal{K}(\beta_i + \alpha_i)$$

To see how this works refer to Figure 5. We desire that θ be zero, in other words that the robot is facing forwards and moving along a straight path. β is the robot's heading. α depends on ϵ_d, the robot's displacement from the centre of the corridor, according to the following equation:

$$\alpha = \arctan\left(\frac{\epsilon_d}{D}\right)$$

This can be seen as a simple kinematic planner to generate reference signals

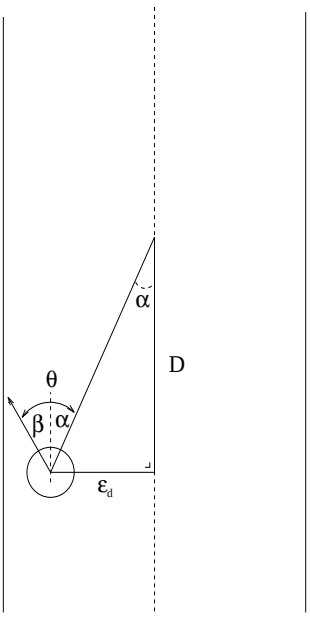


Fig. 5. Angles used in the control law. We desire that $\theta = 0$

which are followed by the robot. D can be increased or decreased to control the influence of the displacement on the overall control scheme. For our purposes it was set to 6.5% of the image width, while $\mathcal{K}$ was set to 0.45, thus ensuring a smooth trajectory back to the centre of the corridor.

5.3 Line Tracking using Prediction

With the control scheme in place, it was necessary to track the corridor guidelines. This three step process is as follows:

- Initialise the bounding boxes around each of the corridor guidelines
- Extract the corridor lines
- Predict where the corridor guidelines shall be located at the next time instant and move the bounding boxes accordingly

One can define a planar projection transformation (homography), $\mathcal{H}_{ir}$ which enables us to determine the image projection of a point, given its world co-ordinates. This homography performs three co-ordinate changes which are defined as:

- ${}^{r}T_{w}$ transforms a 3D point from the world co-ordinate frame to the robot frame
- ${}^{c}T_{r}$ transforms the point from the robot frame to the camera frame
- ${}^{i}T_{c}$ transforms the point from the camera frame to the image frame

Each transformation can be represented by a 3x3 matrix. So, we have the following:

$$ {}^{i}p = {}^{i}T_{c}\,{}^{c}T_{r}\,{}^{r}T_{w}\,{}^{w}P $$

$$ {}^{i}p = \mathcal{H}_{ir}\,{}^{w}P $$

We now have a point in the image, ${}^{i}p$ and so predicting where the corridor guidelines shall be located only involves two steps. Firstly, we calculate the planar homography, $\mathcal{H}_{im}$ between a point in the image, ${}^{i}p_{t}$ at time instant, t and the same point in the next image at time, *t+1*:

$$ {}^{i}p_{t+1} = \mathcal{H}_{im}\,{}^{i}p_{t} $$

Unfortunately, $\mathcal{H}_{im}$ is only good for obtaining point co-ordinates from one image to the next. Since we wish to track lines we need an homography, $\mathcal{H}_{l}$ that will allow us to do this. Fortunately, it can be shown that this homography is just the *inverse transpose* of the homography for points that we have already obtained. Hence, for our second step we have that:

$$ {}^{i}l_{t+1} = \mathcal{H}_{l}\,{}^{i}l_{t} $$

$$ where\ \mathcal{H}_{l} = \mathcal{H}_{ir}^{-T} $$

and ${}^{i}l$ are the projective co-ordinates of a line on a plane.

6 Experimental Results

6.1 Experimental Environment

Real world experiments were undertaken along a corridor of our research institute. It represents a simple indoor office environment. Simulated results were obtained from non-optimized code running on a 300MHz Pentium II PC.

Reference positions at which images were acquired were ordered according to the direction of motion, thus maintain the causality constraint. Images, both warped omni-directional and ground plane dewarped were acquired every 50 cm. A sequence of both types of images is shown in Figures 6 and 7.

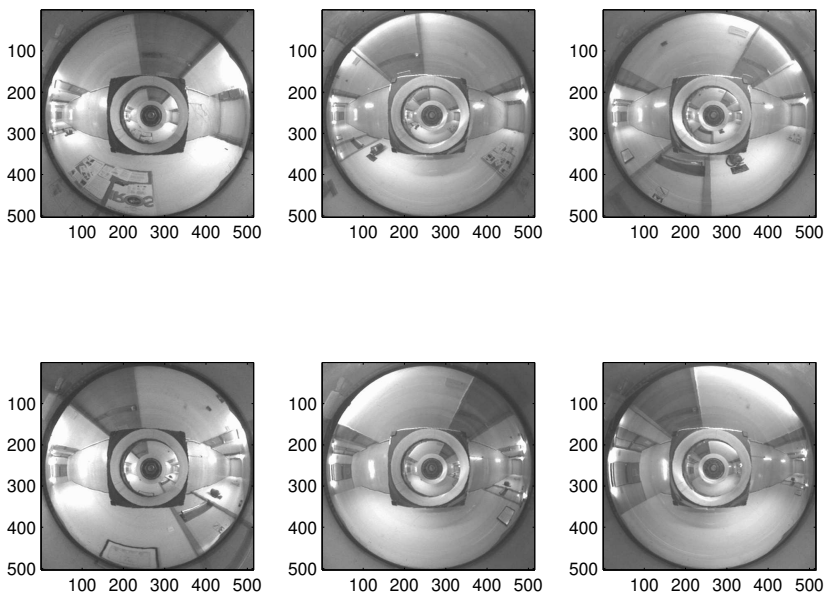


Fig. 6. A sequence of omni-directional images.

6.2 Eigenspace Results

Figure 8(a) shows the first 9 eigenimages computed from the image set of 30 omni-directional images, representing one corridor. They are shown in descending order in accordance with their eigenvalues. As an example of the effectiveness of appearance based approach, Figure 9(a) shows the unknown image that was projected into the eigenspace and its closest match (Figure 9(b)) obtained from the set of previously acquired images. This retrieved image represents the qualitative topological position of the robot.

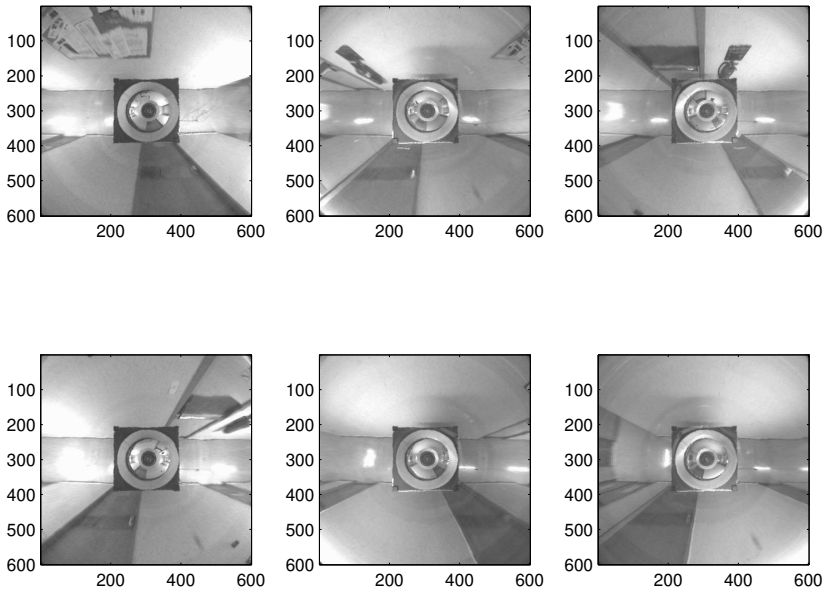


Fig. 7. A sequence of ground plane dewarped images.

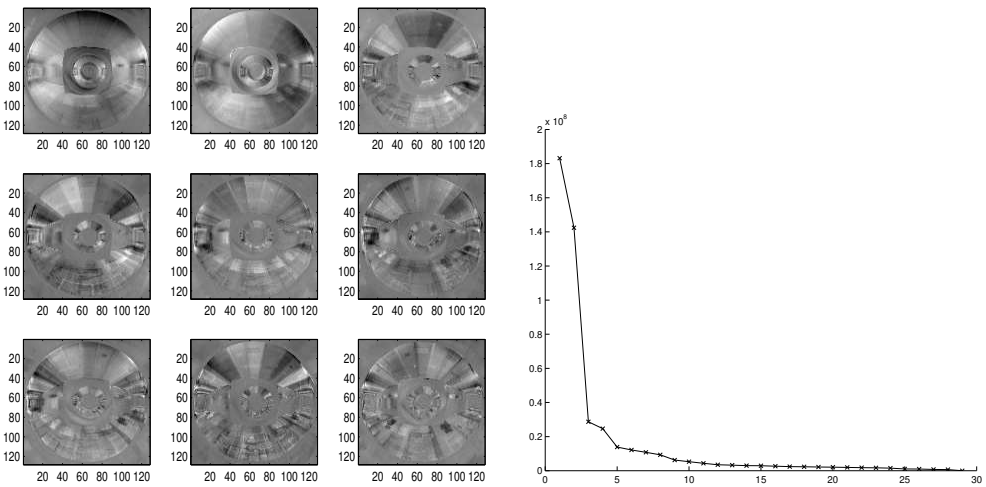
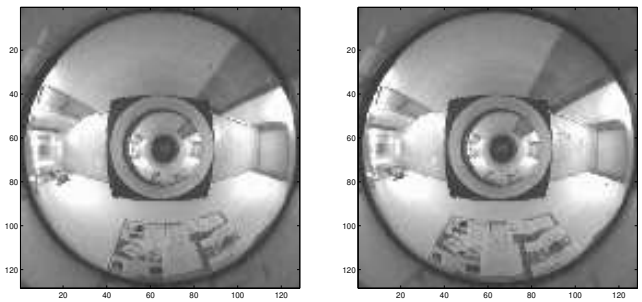


Fig. 8. (a) The first 9 eigenimages obtained with the omni-directional vision system (b) eigenvalue drop off.



(a) (b)

Fig. 9. (a) Unknown and (b) retrieved omni-directional images.

Ten to twelve basis vectors were found to be enough to correctly determine the nearest image to the newly projected one. With twelve dimensions, misclassification did not occur. Figure 8(b) shows how the first eigenvalues accounted for most of the variance in the images.

6.3 Localisation Results

Once the robot's position in the topological map has been determined, the next stage involves the calculation of it's heading and orientation, in order to localise the robot in the centre of the corridor while simultaneously keeping it moving in a straight line. This was achieved by dewarping the image to ground plane view (see subsection 5.1) and performing Sobel edge detection [5] in the horizontal direction. The RANSAC [3] fitting procedure was then implemented to determine the corridor guidelines. This procedure was chosen due to the fact that it is robust to outliers, having a 99% chance of finding two correct points even if 25% of the total number of points are outliers. An alternative approach is to use first order derivatives to determine the heading direction. In our experience, this method is not robust due to the fact that ground plane dewarped images are relatively noisy.

Once the guidelines were found the robot determined its translation from the centre of the corridor and its orientation. The use of the ground plane dewarped images significantly simplifies this task since the image of the ground plane is just a uniformly scaled version of the ground plane co-ordinates (no perspective effects). In other words, the world-to-image Jacobian is an exact linear transformation. These results were used as input to a simulator. This simulated the robot's movement down a corridor under closed loop control. Initially, the robot's heading direction was offset and it was positioned so as to be displaced from the centre of the corridor. Figure 10 shows how the robot corrected its orientation and translation errors over time. This was achieved by using prediction to track the corridor guidelines (see Section 5) and then utilising this visual feedback for control purposes. The dashed-dot lines in each of the images are

the *predicted* positions of the corridor lines for that time instant. As can clearly be seen, this works well, enhancing the probability that the bounding boxes will contain corridor guidelines and not erroneous data such as door segments. Figure 11 (a) shows the trajectory over time and Figure 11 (b) shows the change in the robot's heading and displacement over the same period.

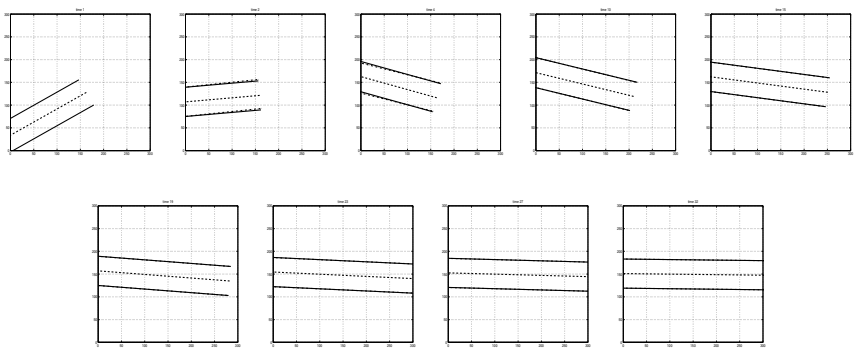


Fig. 10. Correction of robot orientation and translation over time. The dash-dot lines represent the predicted position of corridor guidelines.

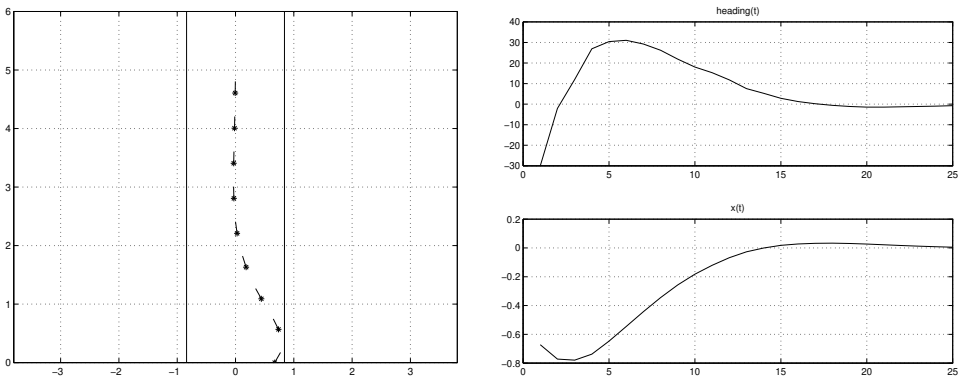


Fig. 11. (a) Simulated robot trajectory, (b) heading and translation.

7 Conclusions & Future Work

In this paper we have demonstrated how to obtain the qualitative position of a mobile robot in a simple indoor environment using a single omni-directional

vision sensor. Neither odometric information nor artificial landmarks were required. This was achieved by merging two distinct paradigms, namely appearance based methods and visual servoing. The geometry of our camera set-up was detailed. Since we had a relatively large image set, PCA allowed us to build up a low dimensional eigenspace representation and project images obtained at unknown positions into it. The manifold in the eigenspace acted as a topological map of the environment. A method for obtaining ground plane dewarped images was outlined. Using a simulation of the robot in its environment the effectiveness of our control algorithm was shown. A method of predicting where next to search for relevant information, namely corridor guidelines was presented.

The next step in our research is to run the robot under closed loop control in the real world. Simulation has shown that our control and tracking mechanism is tuned to this task. Prediction proved to be beneficial in disregarding irrelevant information.

Future work will be concerned with providing a mechanism whereby the robot can control its motion through doors and around corners. As an initial step to achieving this goal, multiple eigenspaces need to be connected together, thus extending our topological representation of the environment. It seems natural to combine this work with [4] to allow accurate, as well as qualitative positioning within multiple indoor environments. In addition, an automatic landmark extractor would be extremely useful in order to only retain images with beneficial information. Currently, reference images are acquired *a priori* at specified intervals.

Acknowledgements

We wish to thank José Gaspar for his helpful comments on this work. This work has been partially funded by the project PRAXIS 2/2.1/TPAR/2074/95, SIVA and the EU TMR network SMART II.

References

1. Aloimonos, J., Weiss, I., Bandyopadhyay, A.: Active vision. *International Journal of Computer Vision*, 1(4):333–356, January 1988.
2. Baker, S., Nayar, S.K.: A theory of catadioptric image formation. In *International Conference on Computer Vision (ICCV'97)*, pages 35–42, Bombay, India, January 1998.
3. Fischler, M.A., Bolles, R.C.: Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24(6):381–395, June 1981.
4. Gaspar, J., Santos-Victor, J.: Visual path following with a catadioptric panoramic camera. In *Proceedings of the 7th International Symposium on Intelligent Robotic Systems (SIRS'99)*, Coimbra, Portugal, July 1999.
5. Horn, B.: *Robot Vision*. MIT Press, 1986.
6. Horswill, I.: Polly: A vision-based artifical agent. In *Proceedings of the AAAI-93*, Washington, DC, USA, 1993.

7. Ishiguro, H., Tsuji, S.: Image-based memory of environment. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'96)*, pages 634–639, 1996.
8. Jones, S.D., Andresen, C., Crowley, J.L.: Appearance based processes for visual navigation. In *Processings of the 5th International Symposium on Intelligent Robotic Systems (SIRS'97)*, pages 551–557, July 1997.
9. Murakami, H., Kumar, V.: Efficient calculation of primary images from a set of images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 4(5):551–515, 1982.
10. Murase, H., Nayar, S.K.: Visual learning and recognition of 3d objects from appearance. *International Journal of Computer Vision*, 14(1):5–24, January 1995.
11. Neilsen, J., Sandini, G.: Learning mobile robot navigation: A behaviour based approach. In *Proceedings of the IEEE International Conference on Systems, Man and Cybernetics*, San Antonio, Texas, October 1994.
12. Svoboda, T., Pajdla, T., Hlavac, V.: Epipolar geometry for panoramic cameras. In *Proceedings of the 5th European Conference on Computer Vision (ECCV'98)*, pages 218–231, Freiburg, Germany, June 1998.
13. Turk, M.A., Pentland, A.P.: Face recognition using eigenfaces. In *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR'91)*, pages 586–591, 1991.
14. Vassallo, R.F., Schneebeli, H.J., Santos-Victor, J.: Visual navigation: Combining visual servoing and appearance based methods. In R.B. Fisher and G. M. Hayes, editors, *Proceedings of the 6th International Symposium on Intelligent Robotic Systems (SIRS'98)*, pages 137–146, Edinburgh, Unitied Kingdom, July 1998.
15. Zheng, J.Y., Tsuji, S.: Panoramic representation for route recognition by a mobile robot. *International Journal of Computer Vision*, 9(1):55–76, January 1992.