Parametric Eigenspace Representations of Panoramic Images *

Matjaž Jogan and Aleš Leonardis
Computer Vision Laboratory
Faculty of Computer and Information Science
University of Ljubljana
Tržaška c. 25, 1001 Ljubljana, Slovenia
{matjaz.jogan,alesl}@fri.uni-lj.si

Abstract

This paper describes a novel approach for robot localization using a view-based representation with panoramic images. We propose to use a representation based on a complex basis of eigenvectors. We demonstrate that this results in a speed up of building the eigenspace and in a fast and accurate localization.

1 Introduction

To enable localization of a mobile robot in an outdoor or indoor environment, a number of methods was proposed that construct a model of the appearance of the environment by capturing images of locations, obtained with an omnidirectional sensor [4, 1, 3, 6, 8, 7]. The model of appearance is predominantly compressed based on the eigenspace approach, which has been successfully used in many areas of computer vision [11, 10].

In the training stage, we acquire panoramic images at several locations. All the images are then compressed by the PCA, resulting in a set of eigenvectors, which, together with the points in the eigenspace corresponding to the images, form the final representation. During navigation, localization is performed by projecting the momentary image directly onto the eigenspace (in the case of robust procedure we have to solve an overdetermined set of equations [9]), followed by the search for the nearest point in the eigenspace. One of the major problems with this kind of representation is how to adequately deal not only with the location of the sensor, but also with its orientation. Some recent approaches [1, 4] used representations that were invariant to rotation, while others tried to explicitly encode the rotation [12, 6]. As we argued in [7], none of them is suitable for *robust* localization in the presence of unexpected obstacles or excessive noise, so we proposed a representation which explicitly integrates all possible rotations in the eigenspace. The main drawback of this method is the high computational complexity of building the eigenspace.

In this paper we show how we can exploit some properties of the space of rotated panoramic images to alleviate this problem. Our method was triggered by the previous work of Uenohara and Kanade [14] that investigated the eigen-basis for a set of images of a single rotated template. In their contribution, Uenohara and Kanade described the relationship between the eigenvectors of a set of uniformly inplane rotated images of an object and the basis vectors of the DCT. They show that the eigenvectors are completely defined by the fact that the inner product matrix of the image vectors is a symmetric Toeplitz matrix. As they claim, the eigenvectors of the inner product matrix are invariant of the image content and they can be generated much more efficiently by calculating the DCT transforms of the autocorrelation vector. This greatly alleviates the computational expense of the training phase.

In this paper we extend the approach to multiple instances of rotated panoramic snapshots of the environment and show how a parametric eigenspace of a set of rotated panoramic images can be built without the need to perform the SVD algorithm on a complete covariance or inner product matrix. Starting from the fact that image shifting can be optimally described with complex Fourier basis, we show that one can construct a compound representation that integrates the Fourier basis and the eigenspace approach.

The paper is organized as follows. In section 2 we give an overview on how eigenvectors can be calculated for a set of rotated versions of a panoramic image. In section 3 we then give a detailed description of our approach to the building of the eigenspace for a set of rotated panoramic images obtained at multiple locations. In section 4 we show the results of applying the proposed algorithm to the problem of

*This work was supported by the Ministry of Education, Science and Sport of Republic of Slovenia (Project J2-0414 and Research Program Computer Vision 1539-506).

localization. The special properties of complex eigenspaces that enable efficient and fast localization are also described in this section. We conclude with a summary and an outline of the work in progress.

2 Eigenspace representation of a set of rotated versions of a panoramic image

Let us first assume, that all of the images in the training set capture the same scene from a single point of view, but under different inplane rotation. The only constraint is that with inplane rotation the information content is preserved. Such is the case when rotated an image of an object on a homogeneous background [14], or with panoramic images rotated around the optical axis [7]. If we unwarp the panoramic image to a cylindrical projection, the rotation can be expressed simply as a rowwise shift (Fig. 1). Since the images are uniformly rotated (shifted), each image can be generated by rotating (shifting) the original image x_0 for $2\pi/N$.

We represent images from the training set as normalized image vectors, from which the mean image is subtracted, in an image matrix $X \in \mathbb{R}^{n\times N}$

$$X = \begin{bmatrix} \mathbf{x}_0 & \mathbf{x}_1 & \dots & \mathbf{x}_{N-1} \end{bmatrix} ,$$

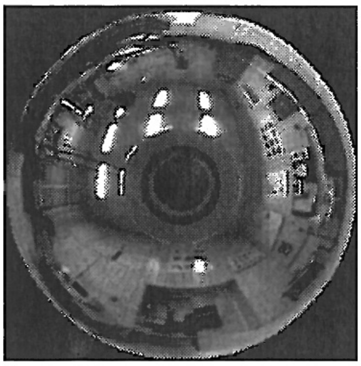
where n is the number of pixels in the image and N is the number of images.

The most straightforward way to calculate the eigensystem is first to subtract the mean of all the images and then to normalize all image vectors to unit energy. We then calculate the SVD of the covariance matrix $C \in \mathbb{R}^{n\times n}$ of these normalized vectors

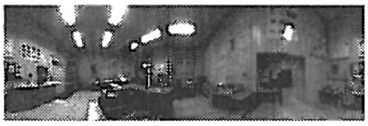
$$C = XX^T = \begin{bmatrix} \mathbf{x}_0 & \mathbf{x}_1 & \dots & \mathbf{x}_{N-1} \end{bmatrix} \begin{bmatrix} \mathbf{x}_0^T \\ \mathbf{x}_1^T \\ \dots \\ \mathbf{x}_{N-1}^T \end{bmatrix} .$$

The eigenvectors $\mathbf{v}_i,\ i = 0,\dots,N-1$ form an orthogonal basis. Sorted with respect to descending eigenvalues $\lambda_i,\ i = 0,\dots,N-1$, they represent the best linear approximation of the image data.

Since the number of pixel elements n in an image is usually high, the computation of the matrix C is a time consuming task of high storage demands. However, it is possible to formulate the equations in such a way that it becomes sufficient to calculate the eigenvectors $\mathbf{v}'_i,\ i = 0,\dots,N-1$, of the inner product matrix $Q \in \mathbb{R}^{N\times N}$



(a)



(b)

Figure 1. (a) Panoramic image taken with a spherical mirror. (b) The same image unwarped to a cylindrical image.

$$Q = X^TX = \begin{bmatrix} \mathbf{x}_0^T \\ \mathbf{x}_1^T \\ \dots \\ \mathbf{x}_{N-1}^T \end{bmatrix} \begin{bmatrix} \mathbf{x}_0 & \mathbf{x}_1 & \dots & \mathbf{x}_{N-1} \end{bmatrix} . \tag{1}$$

Since the eigenvectors $\mathbf{v}'_i$ are the solution of $X^TX\mathbf{v}'_i = \lambda'_i\mathbf{v}'_i$, we can calculate the eigenvectors of XX^T by $XX^TX\mathbf{v}'_i = \lambda'_iX\mathbf{v}'_i$ [2]. In this way, we derive the eigenvectors $\mathbf{v}_i$ of the covariance matrix just by projecting the $\mathbf{v}'_i$ on the set of images,

$$\mathbf{v}_i = \frac{1}{\sqrt{\lambda'_i}} X\mathbf{v}'_i .$$

Uenohara and Kanade [14] showed that in the case of the image set consisting of rotated examples of one original image, Q is a symmetric Toeplitz matrix. Since Q is also circulant, its eigenvectors $\mathbf{v}'_i$ are not dependent on the contents of the images [14]. Q is of the form

$$Q = \begin{bmatrix} q_0 & q_1 & \dots & q_{N-2} & q_{N-1} \\ q_{N-1} & q_0 & q_1 & \dots & q_{N-2} \\ q_{N-2} & q_{N-1} & q_0 & q_1 & \dots \\ \dots & \dots & \dots & \dots & \dots \\ q_1 & \dots & q_{N-2} & q_{N-1} & q_0 \end{bmatrix} .$$

It can be derived from the *shift theorem* [5], that the eigenvectors of a general circulant matrix are the N basis vectors from the Fourier matrix $F = [\mathbf{v}'_0, \mathbf{v}'_1, \ldots, \mathbf{v}'_{N-1}]$

$$\mathbf{v}'_k = \left[1, w^k, w^{2k}, \ldots w^{(N-1)k}\right]^T, \quad k = 0, \ldots, N-1;$$

where $w = e^{-2\pi j/N}$, $j = \sqrt{-1}$.

The eigenvalues can be calculated simply by retrieving the magnitude of the DFT of one row of Q;

$$\lambda'_k = \sum_{i=0}^{N-1} q_i e^{-2\pi j \frac{ik}{N}} \ .$$

This interesting property also emphasizes the central point of the Fourier analysis, as it indicates that the Fourier basis diagonalizes every periodic constant coefficient operator, in our case the circular shift operator [13]. In other words, all basis functions of the Fourier transform are eigenvectors of the circular shift operator [5].

Since $q_i = q_{N-i}$, our matrix is circulant symmetric, and therefore we can choose an appropriate set of real–valued orthogonal eigenvectors. As it turns out, the proper basis are the cosine functions from the real and the sine functions from the imaginary part of the Fourier matrix [13].

We can therefore compute the eigensystem of Q just by first computing the autocorrelation vector $[q_1, q_2, \ldots, q_{N-1}]$, and then by calculating the λ'_i values, which should be afterwards sorted by decreasing magnitude. The eigenvectors $\mathbf{v}'_i$ corresponding to k largest eigenvalues can then be easily selected from the corresponding basis vectors of the Discrete cosine transform (DCT) [14]:

$$v'_{km} = \cos\left[\frac{\pi(2m+1)k}{2N}\right]; \quad \begin{array}{rcl} m & = & 0, \ldots, N-1 \\ k & = & 0, \ldots, N-1 \end{array}$$

Thus, with the help of the DCT, it is possible to compute the basis vectors much more efficiently.

3 An extension to a set of rotated panoramic images acquired at different locations

When dealing with the problem of localization of a mobile robot [7], we need to encode P images from P locations, each of them being rotated N times (Fig. 2). In this case, we cannot directly apply the previous approach to the calculation of eigenvectors of circulant matrices, since the inner product matrix A,

$$A = X^T X = \begin{bmatrix} Q_{11} & Q_{12} & \ldots & Q_{1P} \\ Q_{21} & Q_{22} & \ldots & Q_{2P} \\ \ldots & \ldots & \ldots & \ldots \\ Q_{P1} & Q_{P2} & \ldots & Q_{PP} \end{bmatrix},$$

is composed of several circulant blocks Q_{ij}, which are, in general, not symmetric. However, as we will show, it is still possible to calculate the eigenvectors without performing the SVD decomposition of A.

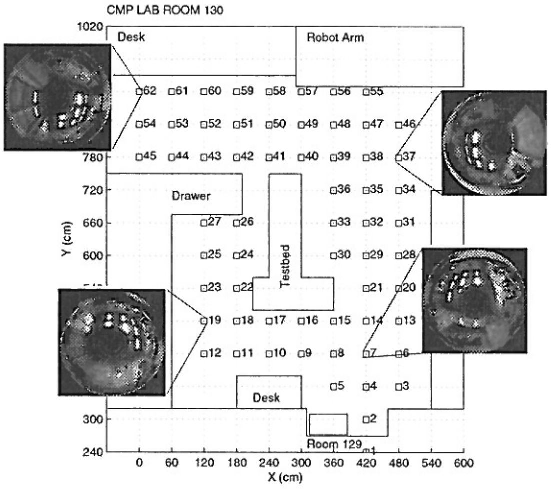


Figure 2. Map of $P = 62$ locations, where images were taken for training the robot. Images shown correspond to locations 7, 19, 37 and 62, respectively.

We have to solve the eigenvalue problem

$$A\mathbf{w}' = \mu\mathbf{w}', \tag{2}$$

where $(\mu, \mathbf{w}')$ is the eigenpair of A. The fact that the matrix blocks Q_{ij} of A are circulant matrices is crucial. As it was already mentioned, every circulant matrix can be diagonalized in the same basis by Fourier matrix F. Consequently all the matrices Q_{ij} have the same set of eigenvectors $\mathbf{v}'_k$, $k = 1 \ldots N$. We shall find the eigenvectors $\mathbf{w}'$ of A among the vectors of the form

$$\mathbf{w}'_k = \left[\alpha_{k1}\mathbf{v}'^T_k, \alpha_{k2}\mathbf{v}'^T_k, \ldots \alpha_{kP}\mathbf{v}'^T_k\right]^T \quad k = 1 \ \ldots \ N \ . \tag{3}$$

Equation (2) can be rewritten blockwise as

$$\sum_{j=1}^{P} Q_{ij}(\alpha_{kj}\mathbf{v}'_k) = \mu\alpha_{ki}\mathbf{v}'_k, \quad i = 1 \ \ldots \ P \ .$$

Since $\mathbf{v}'_k$ is an eigenvector of every Q_{ij} the equations simplify to

$$\sum_{j=1}^{P} \alpha_{kj}\lambda'^k_{ij}\mathbf{v}'_k = \mu\alpha_{ki}\mathbf{v}'_k, \quad i = 1 \ \ldots \ P,$$

where λ'^k_{ij} is an eigenvalue of Q_{ij} corresponding to $\mathbf{v}'_k$. This implies a new eigenvalue problem

$$\Lambda\boldsymbol{\alpha}_k = \mu\boldsymbol{\alpha}_k \ , \tag{4}$$

where

$$\Lambda = \begin{bmatrix} \lambda'^{k}_{11} & \lambda'^{k}_{12} & \dots & \lambda'^{k}_{1P} \\ \lambda'^{k}_{21} & \lambda'^{k}_{22} & \dots & \lambda'^{k}_{2P} \\ \dots & \dots & \dots & \dots \\ \lambda'^{k}_{P1} & \lambda'^{k}_{P2} & \dots & \lambda'^{k}_{PP} \end{bmatrix}$$

and

$$\boldsymbol{\alpha}_k = [\alpha_{k1}, \alpha_{k2}, \dots \alpha_{kP}]^T \ .$$

Since A is block-symmetric, Λ is symmetric and we have P linearly independent eigenvectors $\boldsymbol{\alpha}_k$ which provide P linearly independent eigenvectors $\mathbf{w}'_k$ in (3). Since the same procedure can be performed for every $\mathbf{v}'_k$, we can obtain $N \cdot P$ linearly independent eigenvectors of A.

It is therefore possible to solve the eigenproblem using N decompositions of order P. Since P is usually small in comparison to the total number of images $P{\cdot}N$, this method offers a similar improvement as the method in [14].

However, by looking at the properties of the circulant matrices one can deduce, that this method works only if we use the complex Fourier basis as the eigenvector set for the circulant matrix. In fact, this set of basis vectors is the only common eigenspace for all the submatrices Q_{ij} from A.

3.1 Complex eigenspace of spinning images

Once we estimated the eigenvectors $\mathbf{w}$' of the inner product matrix A, we can calculate the eigenvectors for our set of images as follows,

$$\mathbf{w}_i = \frac{1}{\sqrt{\mu_i}} X \mathbf{w}'_i \ .$$

We then sort the eigenvectors according to their eigenvalues μ_i. To illustrate the background of the formation of the eigenvectors, we show in Fig. 3 the images of the vectors $\mathbf{w}'$ and $\mathbf{w}$. As a comparison, in Fig. 4 we depict the corresponding eigenvectors calculated using SVD.

As λ_i indicate the variance that each eigenvector encompasses, we can use only $K, K << \min(n, N \cdot P)$ eigenvectors in the final representation. Since the eigenvectors come in symmetric pairs, we store just one of a pair. A good estimate of the compressing efficiency of the eigenspace is the energy of the eigenspace, which can be calculated as

$$\frac{\sum_{i=0}^{K-1} \lambda_i}{\sum_{j=0}^{S-1} \lambda_j} \cdot 100\% \ ,$$

where S is the total number of eigenvectors.

To represent each training image in the eigenspace, we have to estimate its coefficients by projecting the image on the basis. Since our eigenvectors are complex, the coefficient vectors are also complex, and can be viewed as points in the complex eigen–subspace. In Fig. 5 we can see the coefficients for an image and its rotated siblings. The rotation (shift) of the image results in the change of angle of

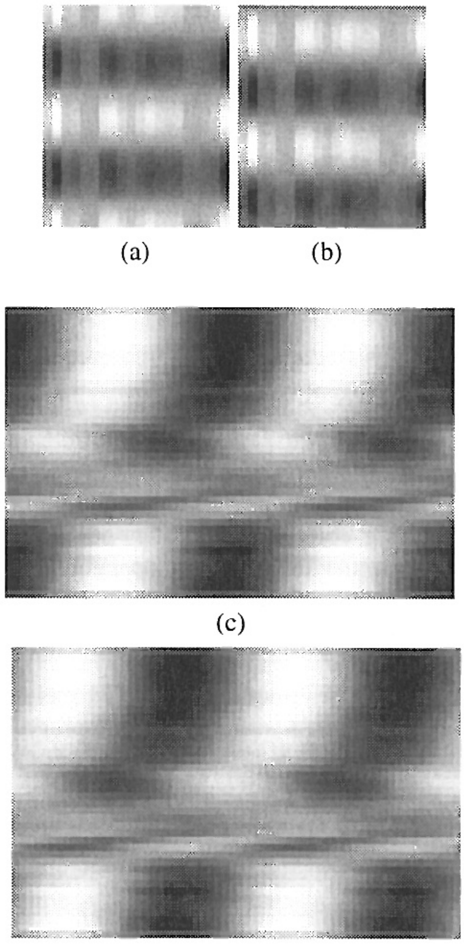

(a) (b) (c) (d)

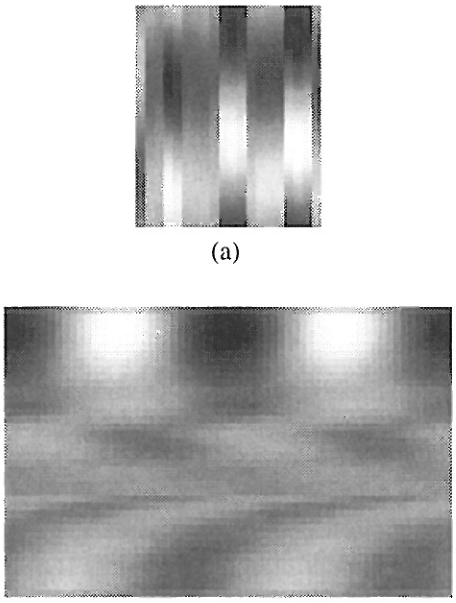
Figure 3. (a) Real and (b) imaginary part of one of the $\mathbf{w}'$ vectors. (c) Real and (d) imaginary part of one of the $\mathbf{w}$ vectors.

the coefficient vector in the complex eigenspace, while the magnitudes of the coefficients do not change.

Because of this property, it is not necessary to store the coefficients for all the images. In fact, if we have just one representative coefficient for one viewpoint, all the others can be generated on the fly with a simple rotation in the complex plane.

Let $\mathbf{x}_i$ be the i-th image and $\mathbf{x}_r;\ r = i{+}1, \dots, (i{+}N{-}1)$ a set of its rotated siblings. Images are evenly rotated, each by $2\pi/N$ radians regarding to its predecessor. The coefficient q_{ij} of the image $\mathbf{x}_i$ can be calculated as $< \mathbf{x}_i, \mathbf{e}_j >$. Let the j–th coefficient of the image $i + 1$ be $q_{(i+1)j}$. The angle between the two coefficients is

$$\phi_j = \arctan \frac{Re(q_{(i+1)j} - q_{ij})}{Im(q_{(i+1)j} - q_{ij})} \ .$$



(a)

(b)

Figure 4. (a) One of the eigenvectors v' of Q calculated by EVD. (b) Corresponding eigenvector v.

Every other coefficient q_{rj} can be now calculated as

$$q_{rj} = q_{ij} e^{2\pi\sqrt{-1}(r-i)\phi} \quad \begin{array}{rcl} r & = & i+1, \ \ldots \ (i+N-1) \\ j & = & 1, \ \ldots \ K \end{array} .$$

4 Robot localization

Once the eigenspace is built, we can interpolate the memorized coefficients in order to form a hypersurface that represents our knowledge of the training data. For mobile robot localization, the parameters to be retrieved are x and y coordinates of the robot and the relative orientation. This is achieved by projecting the momentary input image into the eigenspace and search for the nearest coefficient on the hypersurface.

Since we memorized only one coefficient vector per location, we search among the scores of N projected coefficients, i.e., we first search for the nearest neighbor of the coefficient vector corresponding to the input image and then we rotate this coefficient vector by Φ, where Φ is a vector of estimated ϕ_k values.

4.1 Using shift invariant properties of coefficients

The time complexity of the search can be quite high if the parameter space is large. It is however easy to decrease it dramatically by using a two-step search.

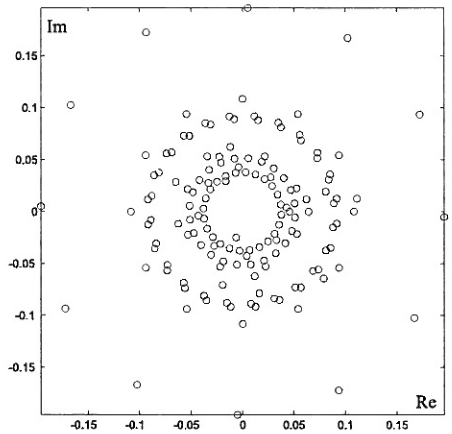


Figure 5. Complex coefficients corresponding to the first ten eigenvectors for one image of the training–set and its rotated siblings.

A closer look at the coefficients reveals that the modulus of the coefficient vectors does not change for the coefficient of the original image and its rotated siblings. If $\mathbf{q}$ is the coefficient vector of an input image, we first calculate its modulus vector $\mathbf{m}$

$$m_i = \sqrt{(Re(q_i))^2 + (Im(q_i))^2} \ .$$

In the first search we make a list of coefficients on the hypersurface which have a modulus vector close to $\mathbf{m}$. These points and their parameters are then the only candidates for the second search. As our experiments show, this strategy dramatically speeds up the algorithm. In fact, a search that would normally take around 120 seconds was completed in 8-10 seconds on average.

4.2 Results of localization

In this section we present the results of localization of a mobile robot equipped with a panoramic camera in an environment of roughly 6x9 meters. Training images were taken at 62 locations, each image was rotated 50 times, so that the interval of rotation was $7.2°$. Tests were performed on an eigenspace of 5 and 10 dimensions for a testing set of 100 test images. The results can be seen in Fig. 6.

In table 1 we compared the times required for building the eigenspace by 1) using the standard decomposition of the correlation matrix XX^T, 2) by calculating the decomposition of the inner product matrix X^TX and 3) by using our approach. The tests were made for images of dimensions 40×68, the latter being the width of the image. Each image was therefore rotated 68 times, i.e., for 40 locations we got 2720 images. Since this is also the number of image

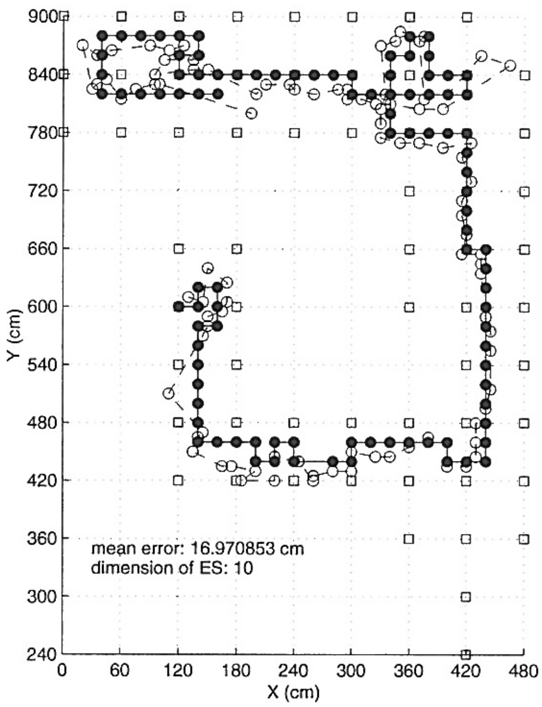


Figure 6. Results of localization for a 10 dimensional eigenspace. Black dots denote test locations. Empty dots denote estimated locations of the robot.

elements, this is the border case when the covariance matrix is of the same size as the inner product matrix, and the complexity of the SVD method reaches its upper bound.

5 Conclusion

In this paper we presented a novel approach to the problem of mobile robot localization using panoramic images. In our previous work we have shown how to overcome the problem of inplane rotation of the panoramic sensor and at the same time achieve robustness by using an eigenspace model based on all the rotated images integrated in the training set. The main limitation of our method was the high complexity of building the eigenspace. We now show that the eigenspace of such a set of images can be efficiently built by using the complex Fourier basis as the eigenvector set for the inner product matrix. The complex eigenspace representation is therefore a compound representation that integrates the Fourier basis and the eigenspace approach. Furthermore, it has some properties that enable a faster search in the coefficient set.

Since our representation enables robust estimation of eigenspace parameters, our plan is to focus now on testing the robustness of the method against noise, occlusion and variations to illumination.

locations (P)	XX^T	X^TX	CPLX
10	2507.3	55.8	16.1
20	2569.6	429.2	105.3
30	2634.8	1400.3	312.4
40	3007.7	3252.3	853.2

Table 1. Timings for building of eigenspace (in seconds).

References

[1] H. Aihara, N. Iwasa, N. Yokoya, and H. Takemura. Memory-based self-localisation using omnidirectional images. *14th ICPR*, pages 297–299. IEEE Computer Society Press, August 1998.

[2] D. Ballard. *Natural Computation*. The MIT Press, 1996.

[3] J. Gaspar, N. Winters, and J. Santos-Victor. Vision-based navigation and environmental representations with an omnidirectional camera. In *IEEE Transactions on Robotics and Automation*, 16(6):890–898, 2000.

[4] H. Ishiguro and S. Tsuji. Image-based memory of environment. In *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Systems*, pages 634–639, 1996.

[5] B. Jähne, H. Haußecker, and P. Geißler, editors. *Handbook of Computer Vision and Applications*. Academic Press, 1999.

[6] M. Jogan and A. Leonardis. Panoramic eigenimages for spatial localisation. *8-th CAIP*. Springer Verlag, September 1999.

[7] M. Jogan and A. Leonardis. Robust localization using eigenspace of spinning-images. In *IEEE Workshop on Omnidirectional Vision*, pages 37–44. IEEE Computer Society, June 2000.

[8] M. Jogan and A. Leonardis. Robust localization using panoramic view-based recognition. In *15th ICPR*, volume 4, pages 136–139. IEEE Computer Society, September 2000.

[9] A. Leonardis and H. Bischof. Robust recognition using eigenimages. *Computer Vision and Image Understanding - Special Issue on Robust Statistical Techniques in Image Understanding*, 78(1):99–118, 2000.

[10] S. Nayar and T. Poggio, editors. *Early visual learning*. Oxford University Press, 1996.

[11] S. K. Nayar, S. A. Nene, and H. Murase. Subspace methods for robot vision. *IEEE Trans. on Robotics and Automation*, 12(5):750–758, October 1996.

[12] T. Pajdla and V. Hlaváč. Zero phase representation of panoramic images for image based localization. *8-th CAIP*. Springer Verlag, September 1999.

[13] G. Strang. The discrete cosine transform. *SIAM Review*, 41(1):135–147, 1999.

[14] M. Uenohara and T. Kanade. Optimal approximation of uniformly rotated images: Relationships between karhunen-loeve expansion and discrete cosine transform. *IEEE Transactions on Image Processing*, 7(1):116–119, 1998.