EigenSegments: A spatio-temporal decomposition of an ensemble of images

Shai Avidan *

The Interdisciplinary Center
Kanfei Nesharim,
Herzliya, Israel
avidan@idc.ac.il

Abstract. Eigensegments combine image segmentation and PCA to obtain a spatio-temporal decomposition of an ensemble of images. The image plane is *spatially* decomposed into temporally correlated regions. Each region is independently decomposed *temporally* using Principal Component Analysis (PCA). Thus, each image is modeled by several low-dimensional segment-spaces, instead of a single high-dimensional image-space. Experiments show the proposed method gives better classification results, gives smaller reconstruction errors, can handle local changes in appearance and is faster to compute. Results for faces and vehicles are shown.

1 Introduction

View-based representation is often used in problems such as object recognition, object detection or image coding. In such an approach 3D objects are represented by a collection of images and therefor a compact representation of ensembles of images is crucial. In this scheme, images are considered as points in the high-dimensional image-space and an ensemble of images of a class of objects forms a non-linear image-manifold in this space. The question is what is the best representation of the manifold for the above mentioned problems.

The first possibility is to approximate the image-manifold with a linear subspace, as was suggested for the case of upright, frontal faces. This subspace is taken to be the leading principal components of the ensemble of images and the key finding of [11, 10] was that the intrinsic dimensionality of this linear PCA-space is much lower than the dimensionality of the image-space. This gives a very compact approximation to a large number of images in terms of a small number of orthogonal basis images, termed "eigenimages".

There exists a constant tension between the desire to increase the number of principal components, to improve reconstruction quality, as well as object recognition and detection capabilites, and the desire to keep that number low to avoid modeling noise and to maintain the computational efficiency of the algorithm. To overcome this hurdle several authors proposed to approximate the manifold with several PCA-spaces [6, 5, 4, 1, 2]. This means that instead of having a single fixed PCA-space to represent all the images in the manifold, the

* This work was done while the author was with MobilEye Vision Technologies

manifold is broken into several regions and each region is approximated with a different PCA-space. Note that the high-dimensional image-manifold is broken into regions, not the $2D$ image-plane. This is where we set in.

Our approach is orthogonal to the above mentioned methods. Instead of taking an image to be a point in the image-space we *spatially* segment the image into several segments and work in each segment-space independently. The spatial segmentation is carried out once, during the training phase, and is based on clustering together pixels that exhibit similar *temporal* behavior. All the methods developed to work in the image-space should work, as is, in each of the segment-spaces as well. Moreover, since our decomposition breaks the high-dimensional image-space into several lower-dimensional segment-spaces the number of samples (per space) is denser and a better approximation, to each of the segment-manifolds, can be obtained.

Our approach is somewhat related to the work on eigenfeatures [8] where facial features such as eyes, nose and mouth were manually selected for face detection and recognition. Instead of manually characterizing special features, such as eyes or nose, we define the special features to be a collection of pixels that exhibit similar temporal behavior. Our work is also related to the work of [4, 5] who used Factor Analysis to approximate the image-manifold by several PCAs. We, on the other hand, focus on the spatial decomposition of the image-plane, as well as the representation of an image by several low-dimensional segment-spaces, instead of a single high-dimensional image-space.

This spatio-temporal decomposition has several advantages. It is as fast and simple to compute as the PCA approach but yet if offers some of the advantages of the mixture of PCA-spaces. Each segment-space can be viewed as an approximation to the original image-space where all the pixels outside the segment are ignored. We also found empirically that our method gives better classification results and can only speculate that the additional spatial information help improve the classification results. Our approach is also better than PCA and PCA-mixture approaches in handling occlusions. Occlusions will only affect some of the segments and we will be able to determine if a mis-fit between an image and our model is local (and hence can be classified as an occlusion and ignored) or is global and act accordingly. Finally, it is natural to extend our approach to handle Level-of-Details PCA, where each segment is approximated by a different number of principal components, say to maintain a predefined reconstruction error.

2 Principal Component Analysis

Principal Component Analysis (PCA) is a statistical method for dimensionality reduction. Let $\mathbf{A} = [A_1 ... A_N]$ be the, so called, design matrix of M rows and N columns, where each column A_i represents an image of M pixels (in vector form). Further assume that all the columns of $\mathbf{A}$ are zero-mean (or else subtract the mean image from all the columns of $\mathbf{A}$). Then the principal components are obtained by solving the following eigenvalue problem:

$$C = UDU^T \tag{1}$$

where $\mathbf{C}$ is the covariance matrix $\mathbf{C} = \frac{1}{N}\mathbf{A}\mathbf{A}^T$, $\mathbf{U}$ is the eigenvectors matrix and $\mathbf{D}$ is the diagonal matrix of eigenvalues. PCA maximizes the variance of the input data samples and the eigenvalues measure the variance of the data in the directions of the principal components. The principal components form an orthonormal basis of the columns of $\mathbf{A}$ and if the effective rank of $\mathbf{A}$ is much smaller than N then we can approximate the column space of $\mathbf{A}$ with $K << N$ principal components. Formally, $A_i^{rec} = \mathbf{U_K^T}\mathbf{U_K}A_i$, is an approximate reconstruction of A_i where $\mathbf{U}_K$ are the first K eigenvectors, and the coefficients $x_i = \mathbf{U_K^T}A_i$ are obtained by projecting the column A_i on the principal components.

3 EigenSegments

An ensemble of images can be approximated by its leading principal components. This is done by stacking the images (in vector form) in a design matrix $\mathbf{A}$ and taking the leading eigenvectors of the covariance matrix $\mathbf{C} = \frac{1}{N}\mathbf{A}\mathbf{A}^T$. The leading principal components are the leading eigenvectors of the covariance matrix $\mathbf{C}$ and they form a basis that approximates the space of all the columns of the design matrix $\mathbf{A}$. But instead of looking at the *columns* of $\mathbf{A}$ we look at the *rows* of $\mathbf{A}$. Each row in $\mathbf{A}$ gives the intensity profile of a particular pixel, i.e., each row represents the intensity values that a particular pixel takes in the different images in the ensemble. If two pixels come from the same region of the face they are likely to have the same intensity values and hence have a strong temporal correlation. We wish to find this correlations and segment the image plane into regions of pixels that have similar temporal behavior. This approach broadly falls under the category of Factor Analysis [3]that seeks to find a low-dimensional representation that captures the correlations between features.

Let $p_{(x_1,y_1)}$ and $p_{(x_2,y_2)}$ be the intensity profiles of two pixels (x_1, y_1) and (x_2, y_2), respectively. Then $p_{(x_1,y_1)}$ and $p_{(x_2,y_2)}$ are N-dimensional vectors (where N is the number of images) where $p_{(x_1,y_1)}(i) \;\; 1 \leq i \leq N$ is the intensity value of pixel (x_1, y_1) in image i. Two intensity profiles are said to be correlated if the dot-product $< p_{(x_1,y_1)}, p_{(x_2,y_2)} >$ is approaching 1 and are uncorrelated if the dot-product is approaching 0. The rows of the design matrix $\mathbf{A}$ are the intensity profiles and hence running a clustering algorithm on the rows of the matrix $\mathbf{A}$ will produce clusters of temporally correlated pixels. In particular, we used the k-means algorithm on the rows of the matrix $\mathbf{A}$ but any method of Factor Analysis can be used.As a result, the image-plane is segmented into several (possibly non-continuous) segments of temporally correlated pixels. Each segment can then be approximated by its own mean segment and set of leading principal components. Put formally, define A_i^s as the pixels in image i that belong to segment s and the design matrix of this segment is given by $\mathbf{A}^s = [A_1^s, A_2^s, ..., A_N^s]$. All the variants for PCA can be applied to each $\mathbf{A}^s$ independently.

The eigensegment approach has nice computational and memory properties. Observe that the collection of the first K principal components of all the S segments occupy the same memory as the first K principal components of the entire image. This is because the number of pixels in all the segments is equal to the number of pixels in the image. The only additional memory requirement is to store the segmentation map that assigns each pixel to a different segment. If we assume we have S segments, each with K principal components then the eigensegment representation is a collection of S K-dimensional points. If we stack them together we have that each image is represented by a SK-dimensional feature vector. This SK-dimensional feature vector is not as optimal as the one obtained by taking the first SK principal components. But it is S times faster to compute because of the additional dot-products required in traditional PCA. This property becomes increasingly important in applications of object detection where exhaustive search over the entire image is often performed and hence speed-ups are crucial.

Another interesting property we found empirically is that the spatial decomposition is often as powerful, if not more powerful, then the temporal decomposition for the purpose of object classification. In the experimental section we demonstrate that good classification scores are obtained if an image ensemble is decomposed spatially and each segment is approximated by its mean intensity value (this can be thought of as the 0-th principal component). This is particularly encouraging because this spatial decomposition can be achieved in a single pass over the test image, compared to the multiple dot-products needed in case of using several principal components.

4 Experiments

We performed a number of experiments to demonstrate the potential of the spatio-temporal decomposition. We show results on faces and vehicles and address issues of image reconstruction and object classification.

4.1 Experiments with Face images

We demonstrate our results on both the Olivetti [9] and the Weizmann [7] face databases. The Olivetti database contains 10 images of 40 subjects, taken over a period of about two years. The size of the original images is 112 × 92 pixels but we reduced their size by half, to increase the speed of computations. The Weizmann database contains about 28 subjects under various pose, lighting and facial expressions. There is a total of 840 images. The original size of the images is 512 × 352 pixels, but we reduced their size to be 32 × 22.

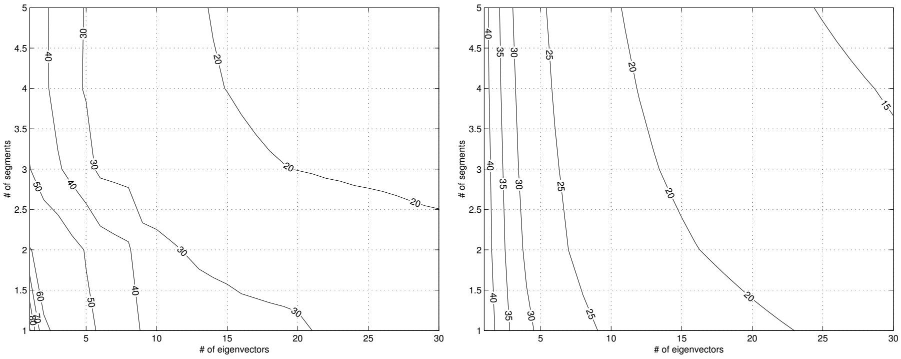
EigenSegment representation In this experiment we measured the reconstruction error of eigensegment representation vs. the usual eigenimage representation. We tested it on both databases by exhaustively computing the reconstruction error for all combinations of up to 5 segments and up to 30 eigenvectors per segment. Figure 2 shows the contour maps for both databases. Each contour in the figure represent an iso-reconstruction error contour in the spatio-temporal space. As can be seen, there is a tradeoff between the number of eigenvectors and the number of segments. For example, In the Olivetti database one segment with 21 eigenvectors has the same reconstruction error (in intensity values), for this image ensemble, as a 5-segment 5-eigenvector combination. The advantage of the latter is that it takes less memory to store the principal components and it takes less time (5 scans of the image, compared to 21 scans in the usual PCA approach) to compute the projection of an image into the eigensegment

Fig. 1. Some sample images from the Olivetti database.

space, albeit at the price of representing an image as a 25-dimensional vector in eigensegment representation compared to a 21-dimensional vector in PCA. The Weizmann database is much more diverse than the Olivetti database (in terms of facial expressions, illumination and head pose) and hence the tradeoff between the number of segments and the number of principal components is lower, but still a factor of 2 in speedup can be achieved.

Face segmentation we segmented the Olivetti database into five segments. Figure 3 shows the result of the segmentation. As can be seen, the segments correspond to natural facial features such as eyes, nose, forehead and hair. This was achieved without enforcing spatial continuity on the various segments. We did note however that as the image ensemble becomes more diverse (in terms of facial expressions, illumination and head pose) the fines of the spatial segmentation deteriorates.

Handling local changes in appearance In another experiment we analyzed the effect of local changes in appearance on our representation. Our learning set consisted of 40 images, one per subject, from the Olivetti database. We then focused on one of the subjects that had glasses in the learning set and took a different image of him, without the glasses. We then measured the reconstruction quality of eigensegment decomposition vs. the reconstruction quality of eigenimage decomposition. Figure 4 shows the results. One interesting observation we



(a) Olivetti database

(b) Weizmann database

Fig. 2. Tradeoff between number of segments and number of eigenvectors. The contour labels measure the reconstruction error (in intensity values) per pixel. For example, In the Olivetti database one segment with 21 eigenvectors has the same reconstruction error, for this image ensemble, as a 5-segment 5-eigenvector combination. See text for details.

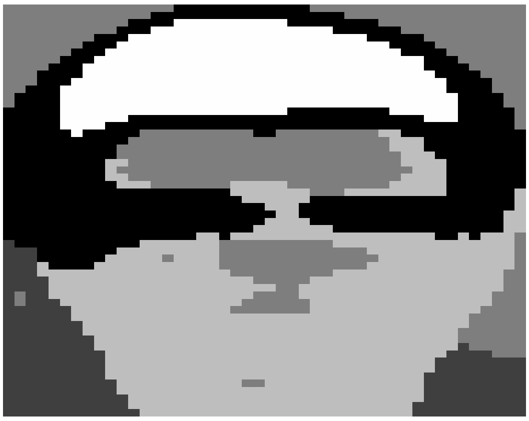


Fig. 3. Segmentation of the Olivetti image ensemble into 5 segments.

can make is that the eigensegment representation can go beyond PCA in terms of representation power. In our case we have only 40 images to learn from and so we are limited to only 40 principal components to represent every new face (even of the same subject). However, by breaking the image into segments we can approximate each segment with it own set of 40 principal components. At the limit, if the number of pixels in the segment goes below 40, we can have a perfect reconstruction.

Eigensegments for recognition In this experiment we tested the classification power of the eigensegment representation, compared to that of the usual eigenspace approach. The Olivetti database was divide into a learning set that contains the first 5 images of every subject and a testing set that contains the last 5 images of every subject. The learning database of 200 images was then decomposed using various combinations of segments and principal components. Both learned images and test images were projected into feature space, according to the particular spatio-temporal decomposition we choose, and a nearest-neighbour algorithm was used to classify each test image. Table 1 compares the results using the vanilla eigenspace vs. the eigensegments approach with either 3, 5 or 20 segments. As can be seen, the best results (189 correct classifications out of 200) are obtained using the segment-mean decomposition, that is the feature vector of the image is taken to be the mean intensity value of each segment. In addition to being the most accurate it is the fastest to compute since only a single scan of the image is needed to compute the mean intensity value of each segment, as opposed to the multiple dot-products required in the usual PCA approach.

# PCs	1 Segment	3 Segments	5 Segments
1	117	158	169
3	169	189	185
5	176	182	186
10	184	183	187
20	185	187	187

# Segments	Correct classification
1	126
3	158
5	164
10	178
20	189

Table 1. The table on the left compares PCA vs. eigensegments for different numbers of principal components (PCs). Classification rates (for 200 test images) for different numbers of segments and eigenvectors are shown. The case of 1 segment is equivalent to the usual eigenimage approach. The table on the right shows the classification results where each segment is approximated by its mean intensity value (which can be seen as the 0-th principal component). In both cases, the Olivetti database was used with 200 images for learning and 200 images for testing. Note that the spatial decomposition where every segment is approximated by its mean intensity value gives the best results. See text for further details.

4.2 Experiments with vehicles images

This experiment demonstrates the use of segments for the use of vehicle recognition using Support Vector Machine (SVM) [12]. We collected a set of 2832

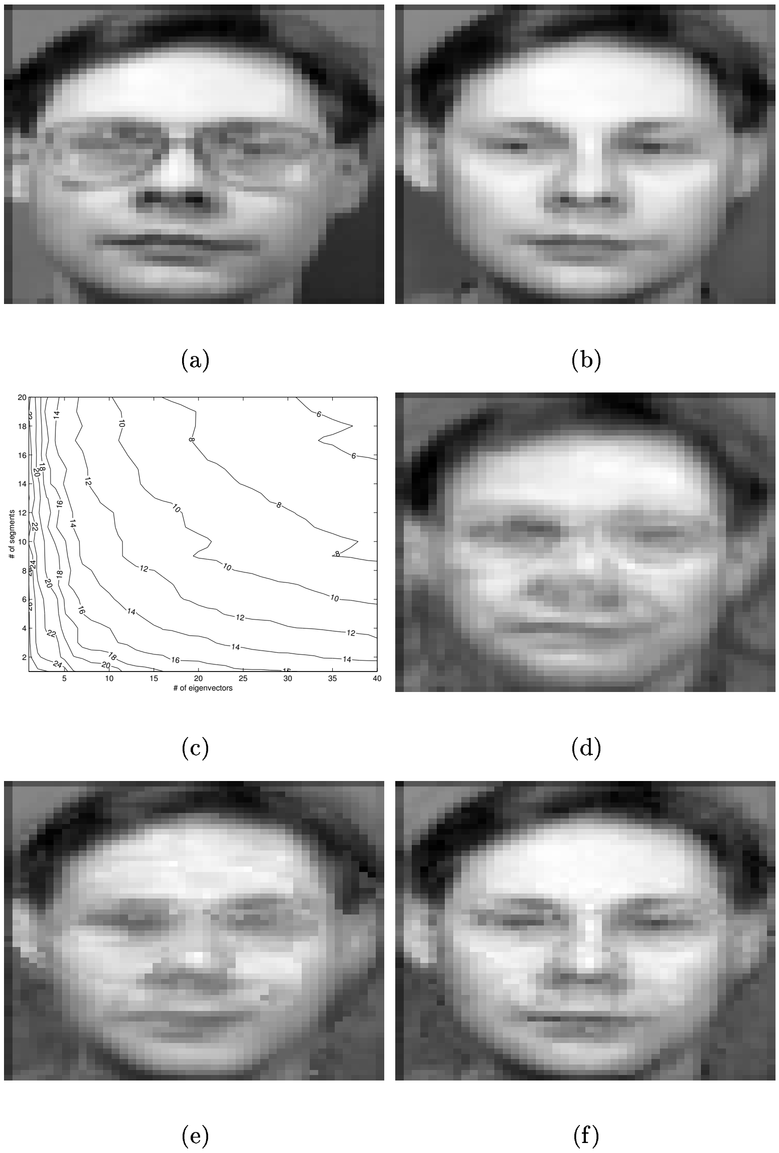

(a)
(b)
(c)
(d)
(e)
(f)

Fig. 4. Reconstruction quality in the presence of local appearance changes. The subject has glasses in the database image (a) but not in the test image (b). Image (c) shows a contour map of reconstruction error for a various number of segments and PCs. Images (d-f) show the reconstructed test image using different combinations of segments and PCs. Image (d) uses 1 segment and 40 PCs, Image (e) uses 10 segments and 20 PCs and Image (f) uses 20 segments and 40 PCs. Image (d) is the best PCA can do because there are only 40 images in the database. Image (e) is as good as image (d) but takes half the time to compute and image (f) is better than the best reconstruction PCA can do. We can keep improving the reconstruction by increasing the number of segments.

images of vehicles and 7757 images of non-vehicles, see Figure 5. All the images are rescaled to be 20×20 pixels in size and the mean intensity value is set to 127 (in the range [0..255]) to compensate for variations in illumination and color. We compared three spatio-temporal decompositions of the image ensemble. In the first case we decomposed the images into 4 segments and approximated each segment with 30 principal components resulting in a 120-dimensions feature vector. In the second case we decomposed the images into 6 segments and approximated each segment with 8 principal components resulting in a 48-dimensional feature vector. In the third case we decomposed the image into 50 segments and took the mean intensity value of each segment, resulting in a 50-dimensional feature vector. The first case requires 30 dot-products of the test image, the second case requires 8 dot-products, and the last case requires a single scan of the test image to compute the mean intensity value of each segment. In every case we fed the feature vectors to a second-order homogenous polynomial SVM. The trained SVM was then tested on a set of 4292 images of vehicles and 9589 images of non-vehicles. The Receiver Operator Characteristics (ROC) curve is shown in figure 6. Each curve represent the percentage of correct classifications of vehicles against false classifications. So, for example, in the third method there is correct classification on 99.29% of the vehicles at the price of wrongly classifying 15.26% of the non-vehicles as vehicles. By changing the SVM threshold we can achieve a classification score of 95.90% on vehicles at the price of wrongly classifying only 3.15% of the non-vehicles as vehicles. The first two methods, that involve a combination of eigenvectors and segmentation perform far worst. The third method, that takes the mean intensity value of the different segments, produced the best results, in terms of classification power, at the lowest CPU time as it requires only a single scan of the test image.

5 Conclusions

We have shown a method for a spatio-temporal decomposition of an ensemble of images. The ensemble is first decomposed spatially into regions of pixels that exhibit a similar temporal behavior and each region is then approximated by a number of principal components. In the experiments we conducted we found, empirically, that this spatio-temporal decomposition can give the same reconstruction errors as the usual eigenimages approach, but at about half the computational price. For the purpose of object classification we found that the mean-segment decomposition, where every segment is approximated by its mean intensity value gives the best result both for face and vehicle classification.

6 Acknowledgment

I would like to thank the research memebers of MobilEye Vision Technologies for their good advice and for allowing me to use some of their data in my experiments.

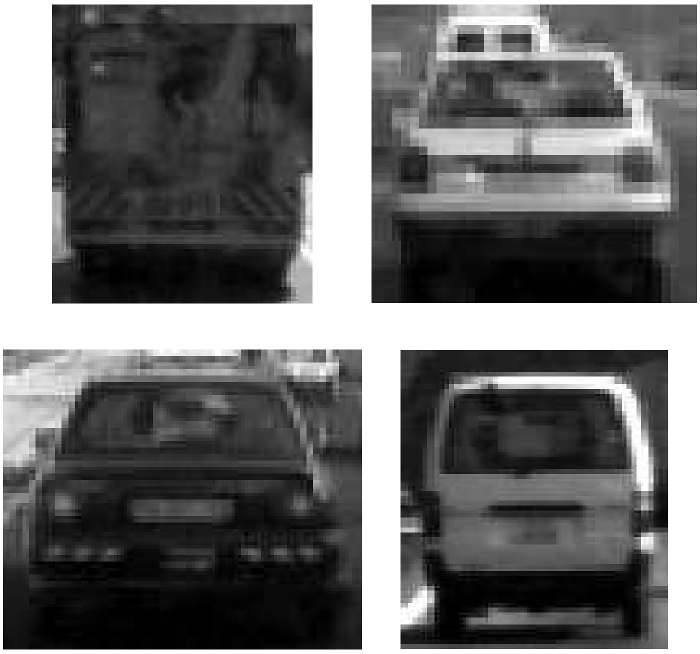


Fig. 5. Some sample images from the vehicle database.

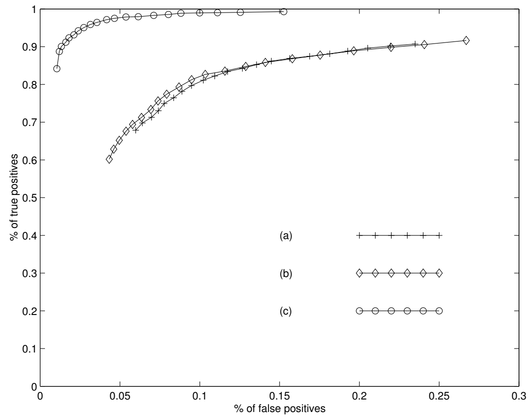


Fig. 6. Comparison of Receiver Operator Characteristics (ROC) for vehicle detection. The image ensemble was decomposed in three different ways. (a) 6 segments and 8 PCs. (b) 4 segments and 30 PCs. (c) 50 segments where each segment is approximated by its mean intensity value. As can be seen, the mean-segment decomposition is far better than the other two methods that involve both spatial decomposition and temporal decomposition. Moreover, method (c) is the fastest to compute as it requires a single scan of the test image to compute the mean intensity value of each segment. See text for further details.

References

1. C. M. Bishop and J. M. Winn. Non-linear Bayesian Image Modelling. In *European Conference on Computer Vision.* Dublin 2000.
2. C. Bregler and S. M. Omohundro. Nonlinear manifold learning for visual speech recognition. In *Fifth International Conference on Computer Vision*, pages 494-499, Boston, June 1995.
3. R. O. Duda and P. E. Hart. Pattern Classification and Scene Analysis. Wiley-Interscience publication, 1973.
4. B. J. Frey, A. Colmenarez and T. S. Huang. Mixtures of Local Linear Subspaces for Face Recognition. In *IEEE Conference on Computer Vision and Pattern Recognition*, Sant Barabara, CA, June 1998.
5. G. E. Hinton, M. Revow and P. Dayan. Recognizing handwritten digits using mixtures of linear models. In *Advances in Neural Information Processing Systems 7*, G. Tesauro, D. Touretzky and T. Leen, Eds. 19955, pp. 1015-1022, MIT Press.
6. B. Moghaddam and A. Pentland. Probabilitic visual learning for object recognition. In *IEEE Transactions on Pattern Ananlysis and Machine Inteliligence*, 19(7):696-710, 1997.
7. Y. Moses. The Weizmann facebase, ftp.idc.ac.il/Pub/Users/cs/yael/Facebase/
8. A. Pentland, B. Moghaddam and T. Starner. View-based and Modular Eigenspaces for Face Recognition. In em IEEE Conference on Computer Vision and Pattern Recognition, Seattle, WA, June, 1994.
9. F. Samaria and A. Harter. Parameterisation of a stochastic model for human face identification. In *2nd IEEE Workshop on Applications of Computer Vision* December 1994, Sarasota (Florida).
10. L. Sirovich and M. Kirby. Low-dimensional procedure for the characterization of human faces. In *Journal of the Optical Society of America 4*, 510-524.
11. M. Turk and A. Pentland. Eigenfaces for recognition. In *Journal of Cognitive Neuroscience*, vol. 3, no. 1, 1991.
12. V. Vapnik. *The Nature of Statistical Learning Theory.* Springer, N.Y., 1995.