3-25

MVA2007 IAPR Conference on Machine Vision Applications, May 16-18, 2007, Tokyo, JAPAN

Adaptive Modified PCA for Face Recognition

Youness Aliyari Ghassabeh
K. N. Toosi University of Technology, Tehran, Iran
y_aliyari@ee.kntu.ac.ir

Hamid Abrishami Moghaddam
K. N. Toosi University of Technology, Tehran, Iran
moghadam@saba.kntu.ac.ir

Mohammad Teshnehlab
K. N. Toosi University of Technology, Tehran, Iran
teshnehlab@eetd.lntu.ac.ir

Abstract

In many real-world applications such as face recognition and mobile robotics, we need to use an adaptive version of feature extraction techniques. In this paper, we introduce an adaptive face recognition system based on PCA algorithm. We combine Sanger's adaptive algorithm for computation of effective eigenvectors with QR decomposition algorithm where used to estimate associated eigenvalues. By normalizing extracted feature vectors, we construct a new more effective feature subspace and used it for on-line face recognition. Experimental results on Yale face data base demonstrated the effectiveness of proposed system in real-time face recognition applications.

1. Introduction

Feature extraction for face representation is one of the central issues to face recognition system. Among various solutions to the problem, the most successful seems to be those appearance-based approaches, which generally operate directly on images or appearances of face objects and process the image as two-dimensional patterns. The main trend in feature extraction has been representing the data in a lower dimensional space computed through a linear or non-linear transformation satisfying certain properties. Statistical techniques have been widely used for face recognition and in facial analysis to extract the abstract features of the face patterns. Principal component analysis (PCA) [1] and linear discriminant analysis (LDA) [2] are two main techniques used for data reduction and feature extraction in the appearance-based approaches. Eigen-faces and fisher-faces [3] built based on these two techniques, have been proved to be very successful. LDA algorithm selects features that are most effective for class separability while PCA selects features important for class representation. The typical implementation of these two techniques assumes that a complete dataset for training is given in advance, and learning is carried out in one batch. However, when we conduct PCA /LDA learning over datasets in real-world applications, we often confront difficult situations where a complete set of training samples is not given in advance. Actually in most cases such as on-line face recognition and mobile robotics, data are presented as a stream. Therefore, the need for dimensionality reduction, in real time applications motivated researchers to introduce adaptive versions of PCA and LDA. Sanger [4] proposed an adaptive algorithm for computation of eigenvectors in decreasing order.

In this paper we introduce an adaptive version of modified PCA (MPCA) [5] and called it adaptive modified PCA (AMPCA). We use sanger's algorithm [4] in order to estimate eigenvectors adaptively and using adaptive QR [6] algorithm, we estimate eigenvalues in each iteration. Then we combine these two algorithms in parallel and propose a new adaptive face recognition system.

The organization of the paper is as follows. The next section provides a brief review of feature extraction using PCA algorithm and reviewed its modified version called MPCA. In section 3, we present our proposed new adaptive face recognition system which is called adaptive modified PCA (AMPCA). Section 4, is devoted to simulations and experimental results. Finally, concluding remarks are given in section 5.

2. Feature extraction

Given a set of centered input vectors $\mathbf{x}_1, \mathbf{x}_2, \ldots\ldots, \mathbf{x}_N$ of n variables, a data matrix $\mathbf{x}$ is defined with each vector forming a column of it. We look for a matrix $\mathbf{A}$ such that:

$$\mathbf{S} = \mathbf{A}^T \mathbf{X} \tag{1}$$

Where columns of $\mathbf{S}$ are decorrelated. Consider a training set with the following parameters, the face training set contains M images, number of pixels in every image and numbers of classes are equal to n and N respectively. Therefore, If $\mathbf{x}_1, \mathbf{x}_2, \ldots\ldots, \mathbf{x}_M$ denotes vectorized images, $\mathbf{x}_i \in \Re^n \ , i = 1,2,\ldots,M$ belong to N classes. The total covariance matrix is defined as:

$$\mathbf{\Sigma} = \frac{1}{M}\sum_{i=1}^{M}(\mathbf{x}_i - \mathbf{m})(\mathbf{x}_i - \mathbf{m})^T = \mathbf{x}\mathbf{x}^T \tag{2}$$

Where $\mathbf{m}$ in (2) is the total mean of the training set. Where $\mathbf{x}$ is defined as:

$$\mathbf{X} = [\mathbf{x}_1 - \mathbf{m}, \mathbf{x}_2 - \mathbf{m}, \ldots, \mathbf{x}_M - \mathbf{m}] \in \Re^{n \times M} \tag{3}$$

PCA algorithm is used to find a sup-space whose basis vectors correspond to the maximum-variance direction in the original space. This can be achieved by computing the eigenvectors of the covariance matrix. The above statement can be written as:

$$\mathbf{\Sigma\varphi} = \mathbf{\varphi\Lambda} \tag{4}$$

Where in (4), $\mathbf{\Lambda}$ denotes a diagonal matrix whose diagonal elements are eigenvalues of $\mathbf{\Sigma}$ and columns of $\mathbf{\varphi}$

are eigenvectors of $\mathbf{\Sigma}$. Because of huge size of the covariance matrix, it is not time and cost efficient to compute eigenvectors using (5). Instead, we construct a matrix $\boldsymbol{B} = \boldsymbol{X'}\boldsymbol{X} \in \Re^{M\times M}$ where $M << n$. By solving the following equation $\mathbf{B}\boldsymbol{\varphi}_\mathbf{B} = \boldsymbol{\varphi}_\mathbf{B}\mathbf{\Lambda}_\mathbf{r}$, where $\Lambda_r = diag[\lambda_1, \lambda_2, ..., \lambda_r] \in \Re^{r\times r} (\lambda_1 \geq \lambda_2 \geq ... \geq \lambda_r)$ and $\boldsymbol{\varphi}_\mathbf{B} \in \Re^{\mathrm{M}\times\mathrm{r}}$, we can find r largest eigenvectors of $\mathbf{B}$. Using the transformation given by:

$$\boldsymbol{\varphi}_\mathbf{r} = \mathbf{X}\boldsymbol{\varphi}_\mathbf{B}\mathbf{\Lambda}^{-1/2} \quad (5)$$

It is possible to compute the r largest eigenvectors of $\boldsymbol{\varphi}$. Therefore, the optimal feature vectors are acquired by projecting $\mathbf{x}$ into largest eigenvector of covariance matrix. Transformation to PCA subspace can be written as follows:

$$\mathbf{y} = \boldsymbol{\varphi}_\mathbf{r}(\mathbf{x_i} - \mathbf{m}) \quad (i = 1,2,...,M) \quad (6)$$

After projection of centralized training images into the PCA subspace, every training face image can be expressed by a linear combination of the basis vector of the PCA subspace as follow:

$$\mathbf{x}_i - \mathbf{m} = \sum_{k=1}^{r} y_{ik}\boldsymbol{\varphi}_k \qquad (i = 1,2,...M) \quad (7)$$

where $\boldsymbol{\varphi}_k$s denote r largest eigenvectors and y_{ik} s are associated weights for projection of input vector $\mathbf{x_i}$ into the $\boldsymbol{\varphi}_k$ s. It is easily proved that:

$$E(y_{i1}y_{i1}) \geq E(y_{i2}y_{i2}) \geq ... \geq E(y_{ir}y_{ir}) \quad (8)$$

From (8), it is apparent that when input images are expressed in PCA subspace, the eigenvectors related to the large eigenvalues have larger weight than others and these eigenvectors are empirically regarded to represent the changes in the illumination [5,7]. Therefore the influence of the illumination should be reduced before using these eigenvectors to calculate the feature space. Influence reduction of the eigenvectors corresponding to the large eigenvalues is done by normalizing the *j*th element of the *i-th* feature vector $y_i\,(y_{ij})$ with respect to its standard deviation, $\sqrt{\lambda_k}$. Hence, the new feature vector $\boldsymbol{y}_i^T$ is rewritten as:

$$\boldsymbol{y}_i^{\;T} = \left(\frac{y_{i1}}{\sqrt{\lambda_1}}, \frac{y_{i2}}{\sqrt{\lambda_2}}, \;...\;, \frac{y_{ir}}{\sqrt{\lambda_r}}\right) \quad (9)$$

These normalized feature vectors are used to construct the new feature subspace.

3. New adaptive face recognition system

In this paper, it is assumed that we confront with a sequence of face images, so all face images are not available at first and during on-line training each images present. We used Sanger's adaptive algorithm in order to estimate eigenvectors and QR decomposition algorithm for adaptive estimation of related eigenvalues. Two algorithms work independently and simultaneously.

3.1 Sanger's algorithm

Sanger [4] proved that the algorithm:

$$\mathbf{C(t+1)} = \mathbf{C(t)} + \gamma(t)(\mathbf{y(t)x(t)}^T - LT[\mathbf{y(t)y(t)}^T]\mathbf{C(t)}) \quad (10)$$

Converge to the matrix T whose rows are the first r eigenvectors of $\mathbf{\Sigma} = E(\mathbf{xx}^T)$ in descending eigenvalue order. In (10), $\mathbf{C}$ is an $r \times n$ matrix, $\mathbf{y(t)} = \mathbf{C(t)x(t)}$, LT[.] set all entries of its matrix argument which are above the diagonal to zero, $\mathbf{x}$ is bounded input vector with autocorrelation matrix $\mathbf{\Sigma}$. $\gamma(t)$ is learning rate. Using (10), it is quite easy to compute r larger eigenvectors adaptively.

3.2 QR decomposition

Consider a $N \times N$ matrix $\mathbf{\Sigma}$, whose eigenvalues are desired. Let $\mathbf{\Sigma_0}$ be equal to $\mathbf{\Sigma}$. Then given $\mathbf{\Sigma_k}$, find a unitary matrix $\mathbf{Q_k}$ and an upper triangular matrix $\boldsymbol{R_k}$ such that $\mathbf{\Sigma_k} = \mathbf{Q_k R_k}$. The next matrix in the iterative sequence is formed as follows:

$$\mathbf{\Sigma_{k+1}} = \mathbf{Q_{k+1}R_{k+1}} = \mathbf{R_k Q_k} \quad (11)$$

The sequence of matrices $\mathbf{\Sigma_k}$ will eventually converge to a diagonal matrix, whose diagonal entries are the eigenvalues of the original matrix $\mathbf{\Sigma}$, arranged in order of magnitudes. Sharman [6] introduced an adaptive version of QR algorithm in order to estimate eigenvalues in on-line applications. Proposed algorithm is as follows:

$$\mathbf{\Sigma_{k+1}} = \mathbf{T_k}^T\mathbf{\Sigma_0 T_k} \;\;, \mathbf{T_k} = \mathbf{Q_0 Q_1 ... Q_k} \quad (12)$$

In adaptive applications the initial matrix Σ_0 can be recursively updated with new data as follows:

$$\mathbf{\Sigma_{0,\mathit{t}+1}} = (1 - a_t)\mathbf{\Sigma_{0,\mathit{t}}} + a_t\mathbf{x}_{t+1}\mathbf{x}_{t+1}^T \quad (13)$$

Where a_t is a constant or time-varying weighting factor and $\mathbf{x_t}$ is the new available data for improving estimation of the covariance matrix. If covariance matrix update occurs when the QR iteration has reached step k, then (12) can be updated as:

$$\mathbf{\Sigma_{k,t+1}} = \mathbf{T}_{k-1,t}^T\mathbf{\Sigma_{0,t+1}T_{k-1,t}} \quad (14)$$

And combining (13) and (14), we will have:

$$\mathbf{y_t} = \mathbf{T_{k-1,t}}^T\mathbf{x_t}\mathbf{\Sigma_{k,t+1}} = (1 - a_t)\mathbf{\Sigma_{k,t}} + \mathbf{y_t y_t}^T a_t \quad (15)$$

Using (15), it is possible to adaptively compute eigenvalues.

3.3 Proposed adaptive face recognition system

We combined two above algorithms in order to estimate effective eigenvectors for presenting data and associated eigenvalues. With arrival of each training image, sanger's algorithm using (10) update effective eigenvectors and simultaneously QR algorithm using (15) update eigenvalues. In other words, we don't need to have all training face images ahead of time, instead we consider training face images as a sequence of random vectors which with arrival of every vector, using (10) and (15) estimation

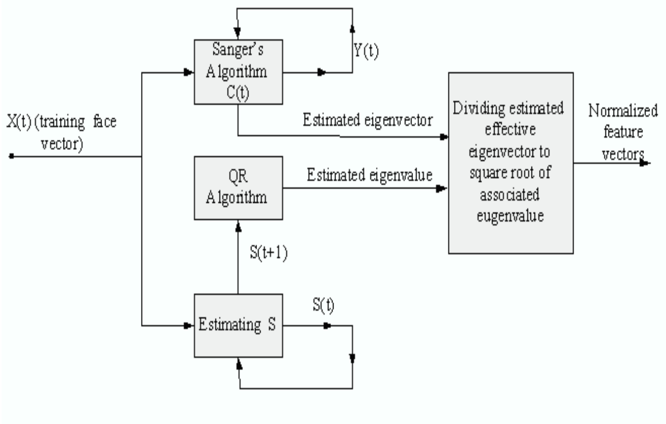

Figure 1. Proposed adaptive face recognition system

of eigenvectors and eigenvalues updated. It is obvious that (10) and (15) work independently and don't effect on each other, this characteristic make it appropriate to implement these algorithm simultaneously. Finally at the end of training process, using (9), we find normalized feature space. Figure 1 shows a simple block diagram for estimating normalized eigenvectors. As figure 1 shows final output of the system is normalized feature vectors which demonstrated effective directions for presenting face images while reducing influence of illumination. Proposed face recognition system uses preprocessed training images. Preprocessing includes histogram equalization and mean centering. Hence mean vector of training face images is not accessible ahead of time, we used (16) in order to estimate mean vector adaptively.

$$\mathbf{m_{k+1}} = \mathbf{m_k} + \eta_{k+1}(\mathbf{x_{k+1}} - \mathbf{m_k}) \qquad (16)$$

in (16) $\mathbf{m_k}$ is estimation of mean vector until k-th iteration, $\mathbf{x_{k+1}}$ is input vector at time k+1 and η_{k+1} is learning rate. Figure 2 shows preprocessing block diagram. It is clear from figure 2 that in preprocessing part raw images cropped, size reduced and histogram equalized and then adaptively mean centered and changes to vectors.

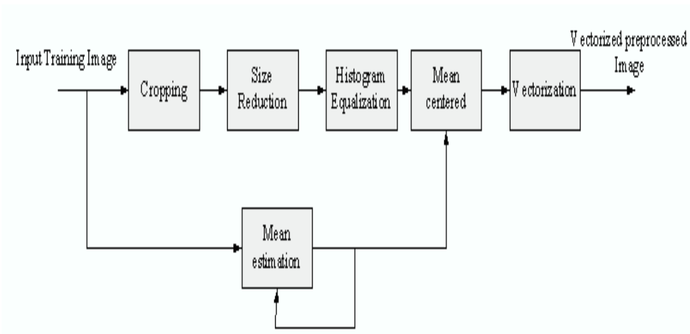

Figure 2. Preprocessing block diagram

4. Simulation results

The developed adaptive face recognition system applied to YALE face database [8]. The images are approximately 128×128 pixels and 256 grayscales. We considered 3 subjects with 60 different available images covering a wide range of illumination and poses. All images are cropped in order to remove background and resized into size of $56(\mathrm{w}) \times 56(\mathrm{h})$. Figure 3 illustrated cropped images used in this experiment. We used leave one-out method to create the training and test data. For each subject we considered 59 face images as training data and remaining 1 face images as test data after classification we changed the test data and considered another face images as test data and repeated this procedure for all 60 face images belong to every subject and computed average of misclassification as error rate. We reduced the dimensionality to 1,5,10,15,20,25,30,35,40 and 50. All test and training images pass through the preprocessing block diagram and therefore histogram equalized and mean centered.

Figure 3. Sample cropped faces used in this experiment

Figure 4 shows 30 normalized eigen-faces [7] ordered according to significance of eigenvectors. These normalized eigen-faces are output of the proposed face recognition system. For each dimensionality, we trained the system and found effective normalized eigenvectors, then we used Euclidean distance as a classifier for incoming test data. Therefore, we assigned the input test image to subject that its mean value has the lowest Euclidean distance from test image. For each subspace, we computed the error rate for both proposed adaptive system called adaptive modified PCA (where normalized features are used) and adaptive PCA (where original features are used, without normalization). We repeated this experiment 10 times and computed average of error rates and average of recognition rates. We also computed variance of error rate and variance of recognition rate for proposed algorithm. Figure 5 shows error rate according to dimensionality and compare the performance of adaptive

PCA and AMPCA, it also shows variance of error rate for adaptive modified PCA method. Figure 6 compare the recognition rate of PCA and AMPCA according to the dimensionality. It also illustrates variance of the recognition rate for AMPCA method.

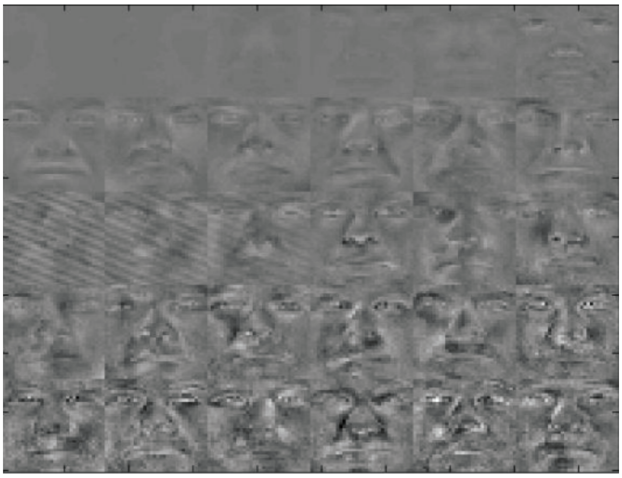

Figure 4. 30 significant normalized Eigen-faces as output of proposed adaptive face recognition system.

It is clear as the dimensionality of sub-space increase, the error rate decrease and as a result recognition rate will improve. It is deduced from figure 6 that normalization of feature vectors improves performance of classifier.

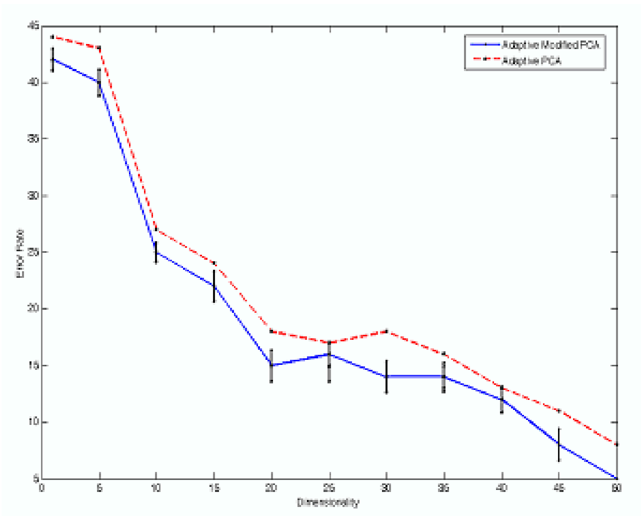

Figure 5. Error rate according to dimensionality for AMPCA and adaptive PCA.

5. Conclusion remarks

In this paper, we introduce a new adaptive version of normalized PCA and called it AMPCA. Then we construct an on-line face recognition system and trained it with a sequence of input images. Simulation results on YALE face database demonstrated the effectiveness of this new adaptive face recognition system in on-line applications. Experimental results demonstrated that for three subjects proposed system showed a good performance and as the dimensionality of eigen-space increases, accuracy of classification will improve. Adaptive nature of the proposed system, make it available for different real-time face and gesture recognition applications.

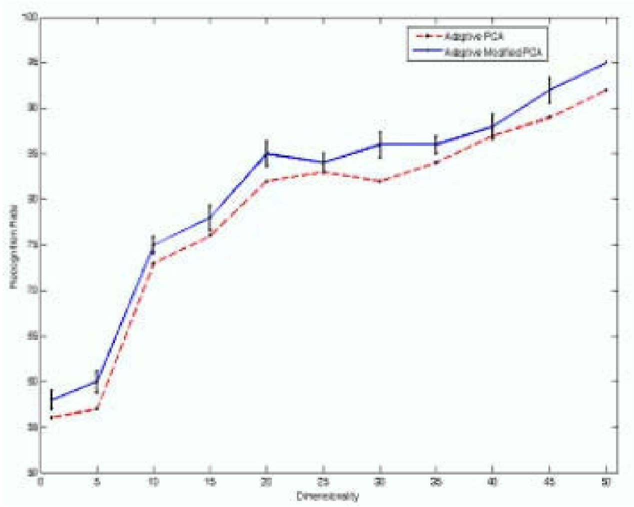

Figure 6. Recognition rate according to dimensionality for AMPCA and adaptive PCA.

Acknowledgments

This project was partially supported by Iranian telecommunication research center (ITRC)

References

[1] M. Turk and A. Pentland, "Eigenfaces for recognition", Journal Cognitive Nero-science, 3(1), 1991.

[2] D.L. Swets and J.J. Weng, "Using Discriminant Eigenfeatures for image retrieval ", IEEE Trans. Pattern Anal. Machine Intell, vol. 18, pp. 831-836, Aug. 1996.

[3] P.N. Belhumeur, J.P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection", IEEE Trans. Pattern Anal. Machine Intell., vol. 19, pp. 711-720, may 1997.

[4] T. D. Sanger, "Optimal unsupervised learning in a single – layer linear feed forward neural network", Neural Networks, Vol 2, pp. 459-473, 1989.

[5] L. Luo, M. N. S. Swamy, E. I. Plotkin, "A modified PCA algorithm for face recognition", IEEE Conf., CCECE 2003. Montreal, May 2003.

[6] K. C. Sharman, "Adaptive algorithms for estimating the covariance eigen structure", IEEE Conf., ICASSP 86, Tokyo 1986.

[7] A. Pentland, T. Starner, N. Etcoff, O. Oliyide, and M. Turk, " Experiments with eigen faces", workshop of IJCAI, 1993.

[8] Georghiades, A.S. and Belhumeur, P.N. and Kriegman, D.J., "From Few to Many: Illumination Cone Models for Face Recognition under Variable Lighting and Pose", IEEE Trans. On Pattern Anal. Mach. Intelligence, Vol. 23, pp. 643-660, 2001.