Implementation of Dimension Reduction Technique for Face Recognition
Rashmi Gupta¹, Sunil Kumar² and Rajiv Kapoor³
¹Maharaja Agarsen Institute of Technology/Electronics, Delhi, India,
Email: rashmig71@yahoo.com
²Maharaja Agarsen Institute of Technology/Electronics, Delhi, India,
Email: s_mathurec@yahoo.co.in
³Delhi Technology University /Electronics, Delhi, India
Email: Rajivkapoor@dce.ac.in

Abstract—In this paper, we implemented a technique that recognizes a person by comparing characteristics of unknown faces with known faces database. The computational method used in this approach is motivated by both Physiology and Information Theory. Here, we treat the problem domain to be intrinsically two dimensional recognition problem, taking advantage of the fact that faces generally are upright, and they can be represented by a few postures and also of the fact that creating three-dimensional models requires specialized hardware and high computation time. The significant features are known as ‘Eigen faces’ because they are the ‘Principal Components’ (Eigenvectors) of the set of faces such as eyes, ears or nose.

Index Terms—dimension reduction, principal component analysis, Eigenfaces, information theory

I. INTRODUCTION

The technique is based on Information Theory approach that decomposes the face images into a small set of characteristic feature images called ‘Eigenfaces’ which may be thought of as the Principal Components of the initial training set of face image [1],[2],[8]. Recognition is performed by projecting a new face into the subspace spanned by the ‘Eigenfaces’ (‘face space”) and then classifying the face by comparing its position in the face space with the position of the known individuals[3].

Automatically learning and later recognizing new faces is possible within this framework. Recognition under widely varying conditions can be achieved by training a limited number of characteristic views (e.g. a “straight on” view, a 45° view and a profile view). This approach has the advantage over other face recognition techniques in its speed and computational simplicity, learning capacity and insensitivity to small and gradual changes in the face image.

II. PRINCIPAL COMPONENT ANALYSIS

Principal component analysis (PCA) is a vector space transform used to reduce multidimensional data sets to lower dimensions for analysis [5]-[7]. Depending on field of application, it is also named the discrete Karhunen-Loève transform (KLT), Hotelling transform or proper orthogonal decomposition (POD)[4]. PCA is the simplest of the true eigenvector-based multivariate analyses. Often, its operation can be thought of as revealing the internal structure of the data in a way which best explains the variance in the data.

III. PROPERTIES AND LIMITATIONS OF PCA

PCA is theoretically the optimal linear scheme, in terms of least mean square error, for compressing a set of high dimensional vectors into a set of lower dimensional vectors and then reconstructing the original set.

It is a non-parametric analysis and the answer is unique and independent of any hypothesis about data probability distribution. PCA compressions often incur loss of information. The PCA is limited by the assumptions made in its derivation. These assumptions are:

1. Assumption on Linearity: We assumed the observed data set to be linear combinations of certain basis.
2. Assumption on the statistical importance of mean and covariance: PCA uses the eigenvectors of the covariance matrix and it only finds the independent axes of the data under the Gaussian assumption. When PCA is used for clustering, its main limitation is that it does not account for class difference since it makes no use of the class label of the feature vector. There is no guarantee that the directions of maximum variance will contain good features for discrimination.
3. Assumption that large variances have important dynamics: It is only when we believe that the observed data has a high signal-to-noise ratio that the principal components with larger variance correspond to interesting dynamics and lower ones correspond to noise.

IV. COMPUTING PCA USING THE COVARIANCE METHOD

The goal is to transform a given data set \( X \) of dimension \( M \) to an alternative data set \( Y \) of smaller dimension \( L \). Equivalently, we are seeking to find the matrix \( Y \), where \( Y \) is the Karhunen-Loève transform (KLT) of matrix \( X \):

\[
Y = KLT \{X\}
\]
A. Organize the data set
Suppose we have data comprising a set of observations of \( M \) variables, and we want to reduce the data so that each observation can be described with only \( L \) variables, \( L < M \). Suppose further, that the data are arranged as a set of \( N \) data vectors \( X_1, \ldots, X_N \) with each \( X_N \) representing a single grouped observation of the \( M \) variables. The column vectors has to be written as \( X_1, \ldots, X_N \), each of which has \( M \) rows. Place the column vectors into a single matrix \( X \) of dimensions \( M \times N \).

B. Calculate the empirical mean
Find the empirical mean along each dimension \( m = 1, \ldots, M \). Place the calculated mean values into an empirical mean vector \( u \) of dimensions \( M \times 1 \).

C. Calculate the deviations from the mean
Mean subtraction is an integral part of the solution towards finding a principal component basis that minimizes the mean square error of approximating the data. Hence we proceed by centering the data as follows:
Subtract the empirical mean vector \( u \) from each column of the data matrix \( X \).

\[
B = X - uH
\]

where \( H \) is a \( 1 \times N \) row vector of all 1's:

\[
h[n] = 1 \quad \text{for} \quad n = 1, 2, \ldots, N
\]

D. Find the covariance matrix
Find the \( M \times M \) empirical covariance matrix \( C \) from the outer product of matrix \( B \) with itself:

\[
C = E[B \otimes B] = E[BB^*] = \frac{1}{N}B B^*
\]

where \( E \) is the expected value operator, \( \otimes \) is the outer product operator, and \( ^* \) is the conjugate transpose operator. Note that if \( B \) consists entirely of real numbers, which is the case in many applications, the "conjugate transpose" is the same as the regular transpose.

E. Find the eigenvectors and eigenvalues of the covariance matrix
Compute Matrix \( V \) of eigen vectors which diagonalizes the covariance matrix \( C \):

\[
V^{-1}CV = D
\]

where, \( D \) is the diagonal matrix of eigenvalues of \( C \). This step will typically involve the use of a computer-based algorithm for computing eigenvectors and eigenvalues.

Matrix \( D \) will take the form of an \( M \times M \) diagonal matrix, where

\[
D[p,q] = \lambda_m \quad \text{for} \quad p = q = m
\]

is the \( m \)th eigen value of the covariance matrix \( C \), and

\[
D[p,q] = 0 \quad \text{for} \quad p \neq q
\]

Matrix \( V \), also of dimension \( M \times M \), contains \( M \) column vectors, each of length \( M \), which represent the \( M \) eigenvectors of the covariance matrix \( C \).

F. Rearrange the eigenvectors and eigenvalues
Sort the columns of the eigenvector matrix \( V \) and eigen value matrix \( D \) in order of decreasing eigen value. Make sure to maintain the correct pairings between the columns in each matrix.

G. Compute the cumulative energy content for each eigenvector
The eigen values represent the distribution of the source data's energy among each of the eigenvectors, where the eigenvectors form a basis for the data. The cumulative energy content \( g \) for the \( m \)th eigenvector is the sum of the energy content across all of the eigenvectors from 1 through \( m \):

\[
g[m] = \sum_{q=1}^{m} D[p,q] \quad \text{for} \quad p = q \quad \text{and} \quad m = 1, \ldots, M
\]

H. Select a subset of the eigenvectors as basis vectors
Save the first \( L \) columns of \( V \) as the \( M \times L \) matrix \( W \):

\[
W[p,q] = V[p,q] \quad \text{for} \quad p = 1, \ldots, M \quad q = 1, \ldots, L
\]

I. Convert the source data to z-scores
Create an \( M \times 1 \) empirical standard deviation vector \( s \) from the square root of each element along the main diagonal of the covariance matrix \( C \):

\[
s = \{ s[m] \} = \sqrt{C[p,q]} \quad \text{for} \quad p = q = m = 1, \ldots, M
\]

Calculate the \( M \times N \) z-score matrix:

\[
Z = \frac{B}{s.H}
\]

J. Project the z-scores of the data onto the new basis
The projected vectors are the columns of the matrix

\[
Y = W^*Z = \text{KLT}(X)
\]

The columns of matrix \( Y \) represent the Karhunen-Loeve transforms (KLT) of the data vectors in the columns of matrix \( X \).
V. RESULTS

The technique is applied on database taken from school of computer science and electronic engineering. This database consists of 395 individuals and 20 images of each individual which amounted as total of 7900 images. These images are taken under different lighting conditions like artificial, mixture of tungsten and fluorescent overhead. These images contain both male and female images from various racial origins and spreads through an age of 18 to 50 years. Fig. 1 shows the results of implementation of dimension reduction technique using PCA. The concept of face space allows the ability to learn and subsequently recognize new faces in an unsupervised manner. When an Image is sufficiently close to a face-space but is not classified as one of the familiar faces, it is initially labeled as “unknown”. The computer stores the pattern vector and the corresponding unknown image. If a collection of “unknown” pattern vectors cluster in the pattern space, the presence of a new but unidentified face is postulated.

VI. CONCLUSION

The Eigen face approach for face-recognition was motivated by Information Theory. It leads to the idea of biasing the face-recognition on a small set of image features that best approximate the set of known face images, without requiring that they correspond to our intuitive image recognition model. Eigenfaces do provide a solution that is well fitted to the problem of face-recognition.

REFERENCES