Face Recognition using PCA and LDA

Aniket Solanki – 16BCP001, Joel Macwan – 16BCP017, Karan Lohia – 16BCP019,

Jinal Kothari – 16BCP021, Smit Sanghavi – 16BCP044, Rajvi Abhangi - 16BCP071

**Abstract—Face recognition is a classical problem which has been around for a long time. For face recognition, the dataset is preprocessed to make it easier for a classifier to process it. PCA is a statistical approach used for reducing the number of variables in face recognition. In PCA, every image in the training set is represented as a linear combination of weighted eigenvectors called eigenfaces. These eigenvectors are obtained from a covariance matrix of a training image set. The weights are found out after selecting a set of most relevant Eigenfaces. Recognition is performed by projecting a test image onto the subspace spanned by the eigenfaces and then classification is done by measuring minimum Euclidean distance.**

**Keywords—Face Recognition, PCA, Eigenvalue, Eigenvector, Eigenfaces, Euclidean distance, Covariance, Fisherfaces, LDA**

1. **INTRODUCTION**

Over the last ten years or so, face recognition has become a popular area of research in computer vision and one of the most successful applications of image analysis and understanding.

An automated face recognition system needs to overcome several difficulties. Lighting condition is another major problem for face recognition. The same person seen under different lighting conditions can appear dramatically different. Facial expression will also make a variation. A reliable face recognition system should be accurate, efficient and invariant to change. For an accurate face recognition system, the accuracy should be over 79%. Efficiency is

critical for a real-time face recognition system. Users cannot tolerate a slow system to recognize a person or wait for

the result of searching. Besides, a face recognition system should overcome the rotational and intensity changes. The system should work properly even the person has little head rotation or under moderate variation in lighting direction, brightness. Otherwise, the system can only be used under some specific conditions which makes it inflexible.

Face recognition can be broken down into the following tasks:

1. Data pre-processing

2. Detect face in the image

3. Analyze facial features

4. Compare against known features

5. Make a prediction

1. **IMPLEMENTATION**

**PCA based Face Recognition**

Step1: prepare the training faces Obtain face images I1, I2, I3, I4 , . . . . . . IM (training faces). The face images must be centered and of the same size.

Step 2: Prepare the data set .Each face image in the database is transformed into a vector and placed into a training set S. Each image is transformed into a vector of size MN × 1 and placed into the set. For simplicity, the face images are assumed to be of size N × N resulting in a point in dimensional space.



Step 3: compute the average face vector The average face vector (Ψ) has to be calculated by using the following formula:

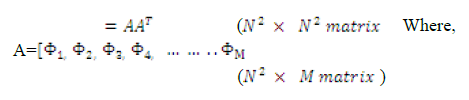


Step 4: Subtract the average face vector The average face vector is subtracted from the original faces and the result stored in the variable ,



Step 5: Calculate the covariance matrix. We obtain the covariance matrix C in the following manner,





Step 6: Calculate the eigenvectors and eigenvalues of the covariance matrix. The covariance matrix C has a dimensionality of N2 X N2 , so one would have N2 eigenface and eigenvalues. Computationally, this is not very efficient as most of those eigenfaces are not useful for our task. In general, PCA is used to describe a large dimensional space with a relative small set of vectors .

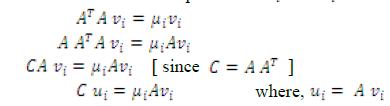
Step 6.1: consider the matrix (M × M matrix)



Step 6.2: compute eigenvectors of



where,



Thus, finally we can obtain eigenvectors as follows:



Step 7: Keep only K eigenvectors (corresponding to the K largest eigenvalues) Eigenfaces with low eigenvalues can be omitted, as they explain only a small part of Characteristic features of the faces.

Next we have to project the training sample into the Eigenface space. The feature weight for the training images can be calculated by the following formula:



Where,ui is the ith Eigenfaces and i=1, 2, 3 . . . . . .K. The weight is obtained as above form a vector as follows:





a)Sample Eigen Faces

**Classification techniques**

a) Read the test image and separate face from it.

b) Calculate the feature vector of the test face using Eigen faces. Each value would represent a weight and would be saved on a vector

c) Compute the average distance (Euclidean distance) between test feature vector and all the training feature vectors. Mathematically, recognition is finding the minimum Euclidean distance , between a testing point and a training point given in the following equation



Where, i = 1, 2, 3. . . . . . K. The Euclidean distance between two weight vectors thus provides a measurement of similarity between the corresponding images.

d) The face class with minimum Euclidian distance shows similarity to test image .

**PCA+LDA Based Face Recognition**

Linear Discriminant Analysis utilizes fischer space method to searches the directions for maximum discrimination of classes in addition to dimensionality reduction .

To accomplish this goal, within-class and between-class scatter matrices are defined .

Consider a set of N sample images (x1, x2, x3…xN), taking values in an n-dimensional image space, and assume that each image belongs to one of C classes {X1, X2,…..,XC}.

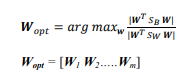
Within class scatter matrix is calculated as



5) Between class scatter matrix is calculated



(6) where µi is the mean image of class Xi, and Ni is the number of samples in class Xi. In LDA, the optimal projection Wopt is chosen to maximize the ratio of the determinant of the between-class scatter matrix of the projected samples to the determinant of the within-class scatter matrix of the projected samples .



Wopt = [W1 W2…..Wm] where w is the set of generalized eigenvectors of SB and SW corresponding to the m largest generalized eigenvalues λ, such that



Where ( i = 1,2,..m) Also, there should be eigenvectors corresponding to at most K largest eigenvalues.

**Classification Techniques**

In order to recognize any image from dataset, Euclidian distance can be used.

2) L2 norm: The Distance measurement, L2 norm is also known as the Euclidean norm or the Euclidean distance when its square root is calculated. It sums up the squared difference between two values. The L2 norm of an image X and an image Y is:



1. **RESULTS**

1) For PCA,

We, tested on 500 images of our class dataset.

Optimal scaling parameters of Haar cascade for dataset image detection are found to

be 1.3 and 5.

Dimensions of Image are kept

128 X 128.

Still classifier is not able to detect faces from many sketches.

We obtained accuracy of 70%

2) For PCA+LDA,

Tested on 500 images of our class

dataset.

Optimal scaling parameters of Haar cascade for dataset image detection are found to be 1.3 and 5.

Dimensions of Image are kept 128

X 128.

We obtained accuracy of 82%

1. **CONCLUSION**

• It is evident from the results that results of

PCA+LDA approach are far better than

implementing only PCA.

• In the above methods,we were able to

detect faces and classified them based on

the facial expression.

1. **REFERENCES**

[1] [Juwei Lu, Plataniotis, K. N., &amp;

Venetsanopoulos, A. N. (2003). Face

recognition using LDA-based algorithms. IEEE Transactions on Neural Networks, 14(1), 195–200.

[2] Zhao, W., Krishnaswamy, A., Chellappa, R., Swets, D. L., &amp; Weng, J. 1998-Springer. Discriminant Analysis of Principal Components

for Face Recognition.

[3] Wang, X., &amp; Tang, X. (2006). Random Sampling for Subspace Face Recognition. International Journal of Computer Vision, 70(1), 91–104.

[4] Belhumeur, P. N., Hespanha, J. P., &amp; Kriegman, D. J. (1996). Eigenfaces vs. Fisherfaces: Recognition using class specific linear projection. Computer Vision — ECCV.