Early Face Recognition using Eigenfaces

Ritu Ann Roy George
B170106EC
B-Batch
NIT Calicut

***Abstract*—This objective of this assignment is to understand and implement Eigen faces method, the foundation of facial recognition systems. The Yale Face database is utilised in this experiment.**

I. Introduction

Face Recognition is one of the most popular applications of computer vision. Principal Component Analysis approach used for dimensionality reduction in face recognition systems, was first developed by Sirovich and Kirby in 1987 and implemented as EigenFaces by Turk and Alex Pentland in face classification in 1991 .

II. PCA and Eigenfaces

Principal Component Analysis, PCA, is one of the most famous dimension reduction techniques in computer vision. PCA is a statistical procedure that uses an orthogonal transformation to convert a set of observations of possibly correlated variables into a set of values of linearly uncorrelated variables called principal components.In a multi-dimensional data, it finds the dimensions that are most useful and contain the most information. thereby extracting essential information from data by reducing the dimensions.

Eigenfaces are an application of PCA. A set of eigenfaces can be generated by performing PCA on a large set of images depicting different human faces. They are used for applications like Face Recognition and Facial Landmark Detection.

In other words, Eigenfaces are images that can be added to a mean (average) face to create new facial images. We can write this mathematically as

$$F = F_m + \sum_{i=1}^{n} \alpha_i F_i \tag{1}$$

where
F is a new face, F_m is the mean or the average face, F_i is an EigenFace and α_i are scalar multipliers we choose to create new faces which can be positive or negative.

III. Algorithm

A. *Read and Vectorize the Input*

Extract the images from the Yale Face Database and flatten each image into a row vector for easy computation. Here the first image is set aside to act as the test image. The rest are utilised to obtain the eigenfaces.

B. *Mean Face*

Obtain the average of all the images. This vector symbolises the mean of the training dataset.

$$F_m = \frac{1}{n} \sum_{i=1}^{n} F_i \tag{2}$$

C. *Principal Component Analysis*

- Compute Covariance Matrix C

$$C = \frac{1}{n} \sum_{i=1}^{n} (X_i - \mu)(X_i - \mu)^T \tag{3}$$

- Obtain eigenvalue and eigenvector

$$Cv_i = \lambda_i v_i \tag{4}$$

  where
  v_i is the eigenvector and λ_i is the eigenvalue.

- Set principal components for reducing dimensions
  Arrange the eigenvalue and the associated eigenvectors in the decreasing order of eigenvalues and choose an arbitrary number of principal components (here n=4 and n = 20 are taken to compare the observations). This is done to obtain the Eigenfaces with the highest eigenvalues.
  The inbuilt function in OpenCV cv2.PCACompute() can also be utilised for the same.

D. *Reconstruction*

Given a new image, find the coefficients to project it on to the space using eigenfaces.

- Vectorize the image and subtract mean vector from it
- Project onto Principal Components
  The dot product of the mean subtracted vector with each of the eigenvectors gives the weight α_i

- Reconstruct the face vector
  Multiply each weight to the eigenfaces and sum them all together. Finally, add the average face vector to this sum.
- Reshape into original dimension to plot the reconstructed image.

E. Recognition

Given a new test image, recognise the image with the closest image in the database. For this, Obtain the weights of the test image and the training image dataset by projecting them onto the principal components. Find the training image whose weights share the minimum euclidean distance with that of the test image.

IV. Observations

Fig. 1: Mean Face

Fig. 2: Test Image

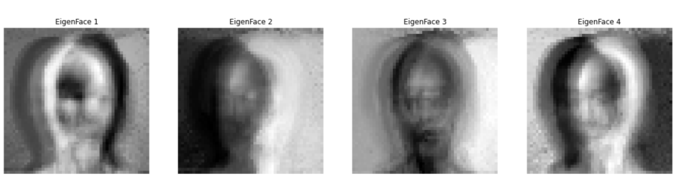

Fig. 3: The 4 Eigenfaces from PCA computation

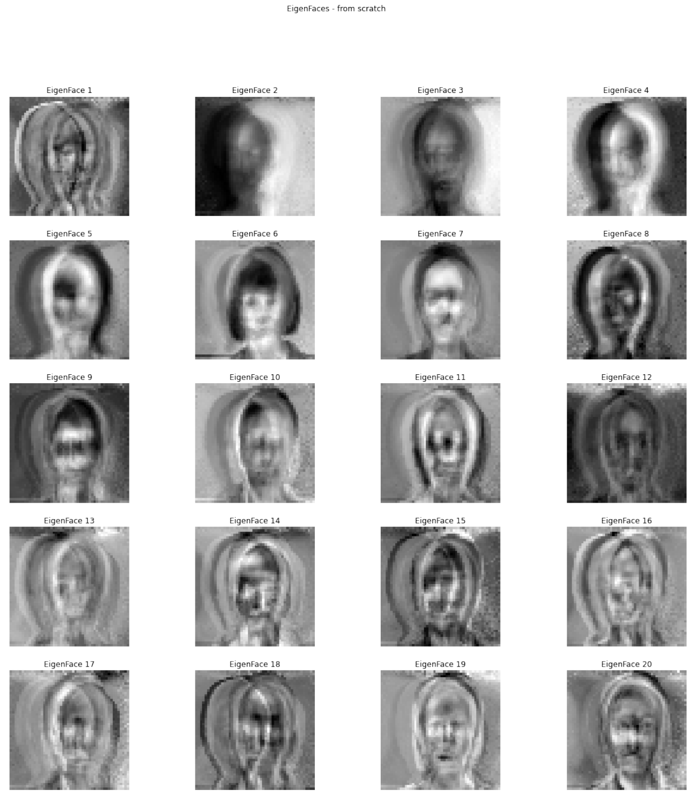

Fig. 4: The 20 Eigenfaces from PCA computation

Fig. 5: The 20 Eigenfaces from cv2.PCACompute()

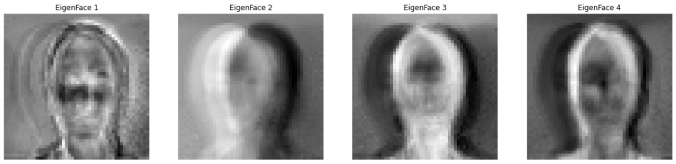

Fig. 6: The 4 Eigenfaces from cv2.PCACompute()

Fig. 7: Reconstructed Image using 4 Eigenfaces

Fig. 8: Reconstructed Image using 20 Eigenfaces

Fig. 9: Recognition (using 4 Eigenfaces)
Predicted image with the closest similarity to test image

Fig. 10: Recognition (using 20 Eigenfaces)
Predicted image with the closest similarity to test image

V. Inference

- The more the number of principal components, better is the result for both reconstruction and recognition.
- No knowledge of geometry and reflectance of faces is required
- Data compression is achieved by the low-dimensional eigenvector representation
  Thus recognition is simple and efficient compared to other matching approaches.
- The drawback of Eigenfaces is its lack of discriminant power.
- May not be robust when dealing with extreme variations in expression and background.

VI. Result

An important application of computer vision, Face Recognition, is implemented using Eigenfaces.