Face Recognition using Fisherfaces

Ritu Ann Roy George
B170106EC
B-Batch
NIT Calicut

***Abstract*—This objective of this assignment is to implement the Fisherfaces method for face recognition and compare its performance with the Eigenfaces method. The Yale Face database is utilised in this experiment.**

I. Drawbacks of Eigenfaces

A drawback of this approach is that the scatter being maximized is due not only to the between-class scatter that is useful for classification, but also to the within-class scatter that, for classification purposes, is unwanted information. Much of the variation from one image to the next is due to illumination changes. Thus if PCA is presented with images of faces under varying illumination, the projection matrix W_{opt} will contain principal components (i.e., Eigenfaces) which retain, in the projected feature space, the variation due lighting. Consequently, the points in the projected space will not be well clustered, and worse, the classes may be smeared together

II. Linear Subspaces

For each face, use three or more images taken under different lighting directions to construct a 3D basis for the linear subspace. Note that the three basis vectors have the same dimensionality as the training images and can be thought of as basis images. To perform recognition, we simply compute the distance of a new image to each linear subspace and choose the face corresponding to the shortest distance. We call this recognition scheme the Linear Subspace method.

III. Drawbacks of Linear Subspaces

Athough the Linear Subspace algorithm would achieve error free classification under any lighting conditions (provided there is no noise or shadowing), there are several compelling reasons to look elsewhere. First, due to selfshadowing, specularities, and facial expressions, some regions in images of the face have variability that does not agree with the linear subspace model. Given enough images of faces, we should be able to learn which regions are good for recognition and which regions are not. Second, to recognize a test image, we must measure the distance to the linear subspace for each person. While this is an improvement over a correlation scheme that needs a large number of images to represent the variability of each class, it is computationally expensive. Finally, from a storage standpoint, the Linear Subspace algorithm must keep three images in memory for every person.

IV. Fisherfaces

The Linear Subspace algorithm takes advantage of the fact that, under admittedly idealized conditions, the variation within class lies in a linear subspace of the image space. Hence, the classes are convex, and, therefore, linearly separable. One can perform dimensionality reduction using linear projection and still preserve linear separability. This is a strong argument in favor of using linear methods for dimensionality reduction in the face recognition problem, at least when one seeks insensitivity to lighting conditions. Since the learning set is labeled, it makes sense to use this information to build a more reliable method for reducing the dimensionality of the feature space. Here we argue that using class specific linear methods for dimensionality reduction and simple classifiers in the reduced feature space, one may get better recognition rates than with either the Linear Subspace method or the Eigenface method.

Fisher's Linear Discriminant (FLD) is an example of a class specific method, in the sense that it tries to "shape" the scatter in order to make it more reliable for classification. This method selects W in such a way that the ratio of the between-class scatter and the withinclass scatter is maximized.

V. Fisherface Algorithm

Let the between-class scatter matrix be defined as:

$$S_B = \sum_{i=1}^{c} N_i(\mu_i - \mu)(\mu_i - \mu)^T \tag{1}$$

and the within-class scatter matrix be defined as:

$$S_W = \sum_{i=1}^{c} \sum_{x_k \in X_i} (x_k - \mu_i)(x_k - \mu_i)^T \tag{2}$$

where μ_i is the mean image of class X_i, and N_i is the number of samples in class X_i. If S_W is non singular, the optimal projection W_{opt} is chosen as the matrix with orthonormal columns which maximizes 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, i.e.,

$$\begin{gathered} W_{opt} = argmax_W \frac{W^T S_B W}{W^T S_W W} \\ W_{opt} = [w_1 w_2 ... w_m] \end{gathered} \tag{3}$$

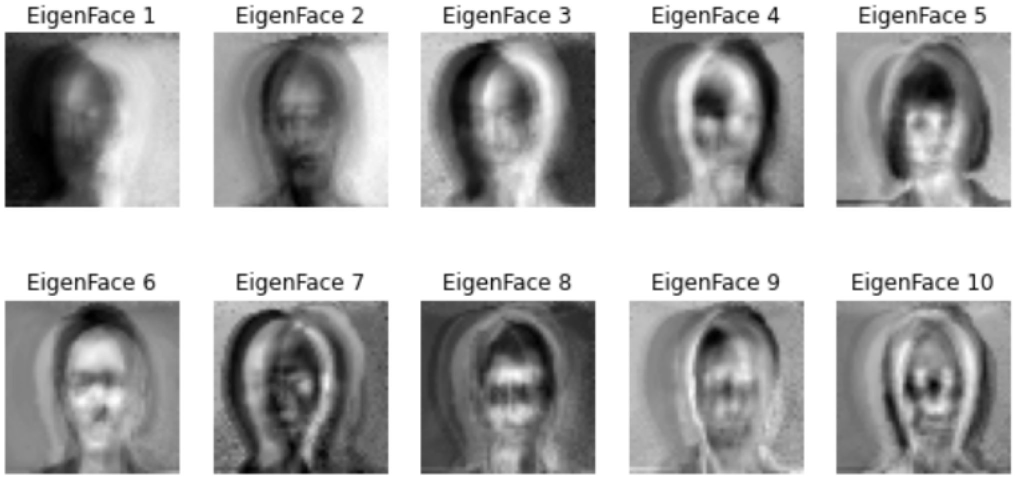

Fig. 1: 20 Eigenfaces

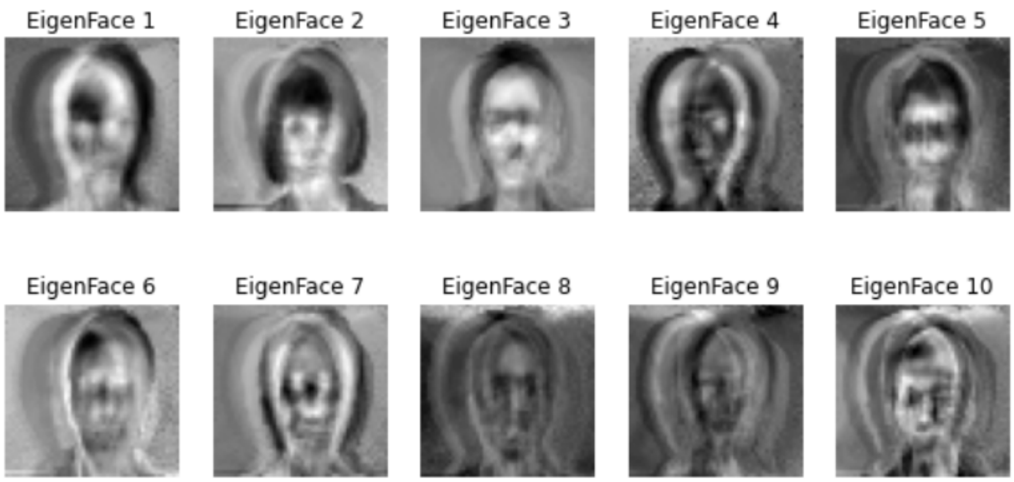

Fig. 2: 20 Eigenfaces after removing the first 3 principal components

where $w_i|i = 1, 2, ..., m$ is the set of generalized eigenvectors of S_B and S_W corresponding to the m largest generalized eigenvalues $\lambda_i|i = 1, 2, ..., m$,i.e.,

$$S_B w_i = \lambda_i S_W w_i, i = 1, 2, ..., m \tag{4}$$

Note that there are at most $c - 1$ nonzero generalized eigenvalues, and so and upper bound on m is $c - 1$, where c is the number of classes.

The Fisherfaces method, projects the image set to a lower dimensional space so that the resulting within-class scatter matrix SW is nonsingular. This is achieved by using PCA to reduce the dimension of the feature space to $N - c$, and then applying the standard FLD to reduce the dimension to $c - 1$.

More formally,

$$W_{opt}^T = W_{fld}^T W_{pca}^T \tag{5}$$

where

$$W_{pca} = argmax_W \mid W^T S_T W \mid$$
$$W_{fld} = argmax_W \frac{\mid W^T W_{pca}^T S_B W_{pca} W \mid}{\mid W^T W_{pca}^T S_W W_{pca} W \mid} \tag{6}$$

Note that the optimization for W_{pca} is performed over $n \times (N - c)$ matrices with orthonormal columns, while the optimization for W_{fld} is performed over $(N - c) \times m$ matrices with orthonormal columns. In computing W_{pca} , we have thrown away only the smallest $c - 1$ principal components.

Taken to an extreme, we can maximize the between-class scatter of the projected samples subject to the constraint that the within-class scatter is zero, i.e.,

$$argmax_{W \in \omega} \mid W^T S_B W \mid \tag{7}$$

where ω is the set of $n \times m$ matrices with orthonormal columns contained in the kernel of S_W

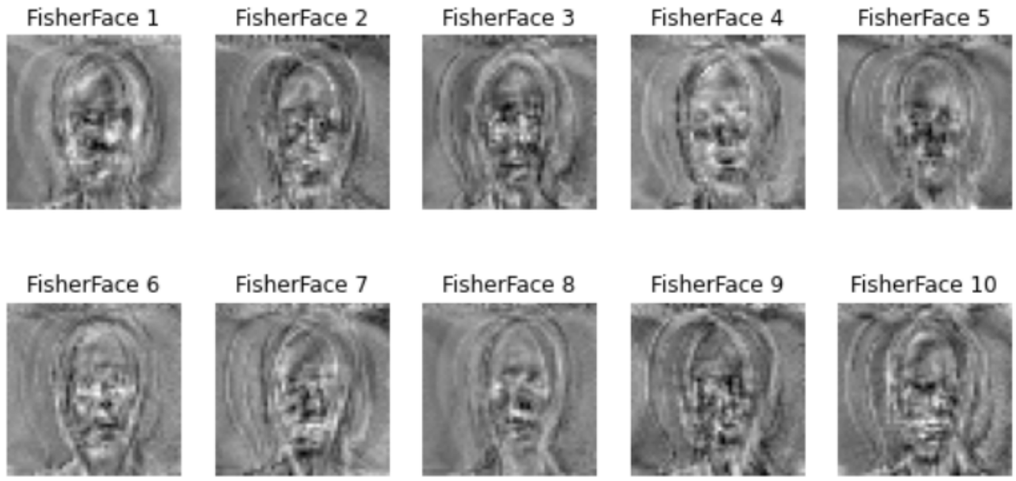

Fig. 3: 20 Fisherfaces

Fig. 4: 20 Fisherfaces after removing the first 3 principal components

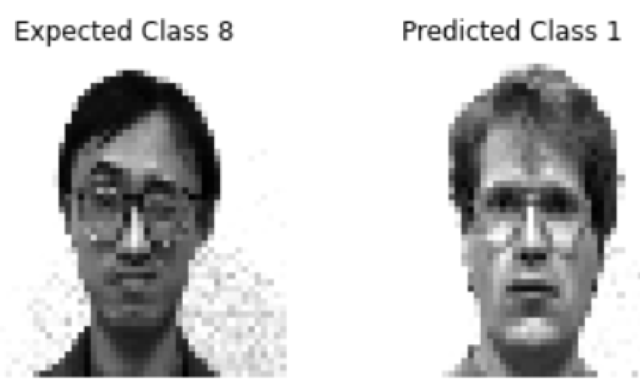

Fig. 5: Incorrect Prediction

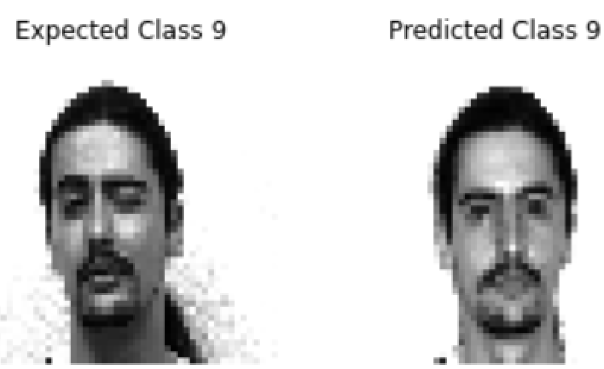

Fig. 6: Correct Prediction

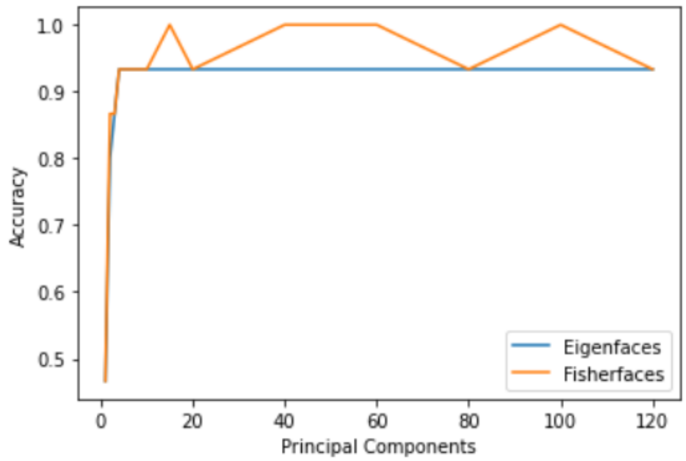

Fig. 7: Accuracy Plot

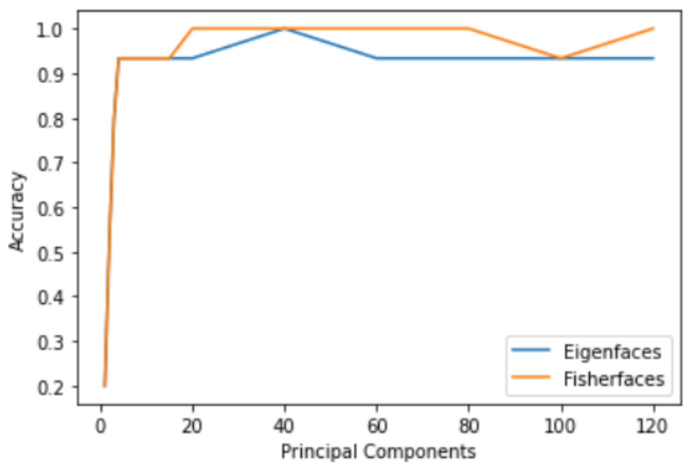

Fig. 8: Accuracy Plot (after removing the first 3 principal components)

VI. Procedure

1) Read and Vectorize the image database and group the images into various classes (each class for each subject). Extract the images from the Yale Face Database and flatten each image into a row vector for easy computation. Here the last image from each of the 15 classes is set aside to act as the test image. The rest are utilised as the training data set.
2) Perform Principle Component Analysis and obtain the eigenvalues and eigenvectors.
3) Extract the first n principal components which are arranged in the descending order of eigenvalues. Let's call this set E.
4) Compute the projections of the images with E. Let's call this P.
5) Using P, find the within class scatter and between class scatter and hence compute the corresponding fisherface eigenvalues and eigenvectors. Let F denote the eigenvectors thus obtained.
6) Using F and P, compute the projections of the dataset images on the fisherfaces.
7) Compute the eigenface and fisherface projections for the test images.
8) Using the concept of Gaussian probability, compute the probability of the test image projections being similar to those of the training images. The one with the highest similarity is passed as the predicted class.
9) Compute the accuracy for various dimensions and observe the change in the accuracy.
10) Repeat the same after removing the first 3 principal eigenvectors and observe the change in the accuracy.

VII. Inference

- In the Eigenface method, removing the first three principal components results in better performance
- The Fisherface method had error rates lower than the Eigenface method and required less computation time.
- The advantages of the Fisherface method are noticeable especially at higher dimensions.

VIII. Result

Face Recognition is performed by Bayes Implementaion on Eigenfaces.