# Human Face Charecterization using Eigenfaces : Midterm report for CS 395T

Kevin Ng and Suresh Subbiah
*The University of Texas at Austin*
*Austin, Texas 78712 USA*

March 22, 2000

**Abstract**

In this project we will approach the problems of human face charecterization and recognition from the perspective of making suitable projections of our large–dimensional data (the face images) so as to utilize their inherent redundancy to reduce the complexity and size of the data set. We use the Eigenface approach of [1] as our base case

## 1 Introduction

The basic goal of this project is to utlize the inherent redundancy of human face image bitmaps to reduce the size and complexity of the data set. This reduction in complexity is achieved by making projections of the initial data set onto "suitable" subspaces. The choice of the suitable subset depends on the many factors such as the (a) the intended application of the data set, (b) the inherent inhomogeneity of the data set and (c) the objective criterion used to determine the subspace. We focus our attention on face charecterization and face recognition. In face charecterization the basic idea is to find an optimal basis to represent a new test image. An image can be represented by a row vector of size $n^2$, where the original image is stored in grayscale of size $n \times n$ pixels. In [1] the authors have found that the eigenbasis of a matrix formed by collecting the row vectors of an initial set of images is an excellent starting point to represent a new image. The authors have found that with a base set of 115 images it took approximately 40 coefficients to get a "good" approximation of an image. The reduction in complexity is enormous!! Initially we have $2^{14}$ integers to represent an image, finally we have approximately $\approx 40$ floats to represent the same image (assuming the eigenpictures have been determined). This technique is simple and popular. It is also called Principal Component Analysis(PCA). This approach is discussed in greater detail in [2] and [3]. The primary emphasis here is to choose a basis such that the projections of the data set onto this basis retain maximum information (i.e. images have large components along the basis directions). In face recognition the primary objective is to be able to distinguish between similar images of distinct subjects. One popular approach for recognition is based on using Fisher's Linear Discriminant. Here the projection that maximizes the ratio of seperation between cluster means to intra–cluster variance is chosen. This approach is discussed in [4]. We intend to modify this approach possibly by choosing a projection which maximizes inter–cluster seperation without regard to intra–cluster variance.

# 2 Eigenfaces

In this section we outline the basic procedure to construct and use Eigenfaces and the progress we have made in implementing the eigenfaces algorithm.

## 2.1 Procedure

Consider a set of $M$ images each represented by a bitmap of $n \times n$ pixel values in grayscale. Each of the $M$ images can be visualised as $n^2 \times 1$ column vector. Let the $i^{th}$ image vector be represented by $\bar{\mathbf{x}}_i$. We form a mean image $\mathbf{m}$, which is the average of all the $M$ images. The data set has to be centered around the origin, so we subtract $\mathbf{m}$ from the $\bar{\mathbf{x}}_i$ to obtain

$$\mathbf{x}_i = \bar{\mathbf{x}}_i - \mathbf{m} \tag{1}$$

We form a covariance matrix C defined as

$$C \equiv \frac{1}{M} \sum_{i=1}^{M} \mathbf{x}_i \mathbf{x}_i^T, \tag{2}$$

where $\mathbf{x}_i^T$ is a $1 \times n^2$ row vector (transpose of $\mathbf{x}_i$). The matrix is of dimension $n^2 \times n^2$ and of rank $M$. We determine the $M$ eigenvectors of $C$ which correspond to non-zero eigenvalues. (There are only M non-zero eigenvalues). In order to avoid diagonalising a large matrix like $C$ there is a trick which reduces the computation significantly. A $n^2 \times M$ matrix $A \equiv [\mathbf{x}_1, \mathbf{x}_2, ...\mathbf{x}_M]$ is constructed. We form a $M \times M$ matrix $B$ defined as

$$B = A^T A \tag{3}$$

The eigenvalues of $B$ and $C$ are identical. The eigenvectors of $C$ may be obtained from the eigenvectors of $B$ by pre–multiplying them with $A$. We denote the non-zero eigenvectors of $C$ as $\mathbf{\Lambda}_i$. The eigenvalues are given by $\lambda_i$. Note that the eigenvalues are indexed in decreasing order, i.e. $\lambda_1 > \lambda_2 > ... > \lambda_M$. Once the eigenvectors are known any image can be represesnted in the eigenbasis. To find the projection of an image along each eigenvector, we take the dot product of that image with the eigenvector. Thus an entire image can be represented by less than $M$ numbers (since the eigenvectors corresponding to smaller eigenvalues do not have a significant projection of the test image along them).

## 2.2 Implementation and Progress

We felt that Eigenfaces was a good starting point for the project since PCA is basic building block of most projection algorithms. We do all our analysis using MATLAB. We have primarily used the "Faces database" maintained by L. Sirovich at $h$ttp://camelot.mssm.edu/databases/databases.html for our analysis. This database contains $M = 143$, full face images of $M$ distinct Caucasian males. Each image is a grayscale of $128 \times 128$ pixel values( n = 128).If the image is in PGM (Portable Gray Map) ascii format then each pixel value is directly represented in the image file in ascii. We use Matlab to read in such a file and produce an image. There are some problems in that some of the images from the database are not represented cleanly using Matlab. The same file viewed using "xv" gives a clean image. This problem occured only for a few images and we have to modify our image viewing code in Matlab slightly. Figure 1. shows a typical image from this database.

We calculate the average face $m$, it is shown in Figure 2. In Figure 3 we show the first face with the mean face subtracted from it.

Once we have the zero centered data, we digonalize the covariance matrix using the trick described in the previous section. The eigenvalues are all positive. The largest eigenvalues are about four orders of magnitude larger than the smallest eigenvalues. The largest 3 eigenvalues are 42.7, 10.6 and 6.5. The smallest three eigenvalues are 0.043, 0.086 and 0.081. As mentioned in [1] the projection of a typical face onto $\mathbf{x}_i$ is roughly proportional to $\sqrt{\lambda_i}$. Thus a typical image has very low wightage in the lower eigenimages. Figures $4 - 6$ show the first three eigenfaces.

For comparision purposes we show the eigenface corresponding to the smallest eigenvalue in Figure 7.

The next three figures show the reconstruction of the first face using 40, 120 and 143 eigenvectors respectively. Compare these figures with Figure 3. We see that there is no significant improvement is going from 40 to 120 eigenvectors. But we do get perfect reconstruction with all the 143 eigenfaces. This is an indication of the correctness of our result.

## 2.3 Discussion of results

We have yet undertaken a quantitative analysis about the quality of our results. That will be the next phase of our project. Generally we notice that our reconstruction with 40 eigenfaces is not as high quality as the ones in [1]. This may be because we have used full face images and not carefully cropped features of images. We were consistently having problems with normalization of the images, since each step outlined in the previous section takes the pixel values away from the original [-127,128] range. An additional confusion arose from the fact that Matlab is capable of doing some rescaling and it can plot images where the pixel values are floats! We are not sure exactly what rescale/smoothing algorithm they use.

Also we have spent some time on providing a nice GUI interface for our project. We have built the GUI windows using the Matlab toolkit. The window allows one to load images from diffrent directories, performs all the required calculations and the one has a choice of a) viewing the original images, b) the mean image, c) the original - mean image, d) eigenfaces or e) reconstructed images. Right now the degree of reconstruction cannot be specified from the GUI, but otherwise all the images can be accessed from the interface.

## 3 Future work

As suggested by Prof. Dhillon we intend to pursue the following directions

- Quantitative estimate of the accuracy of our reconstruction.
- Testing our algorith with some larger database like the AR face database from Purdue.
- A face recognition algorithm in which we form clusters which will maximize distance between cluster means. We plan to use some type of k–means algorithm to obtain the clusters. We will quantify how useful this approach is for face recognition.
- If time permits we try diffrent ways of visualizing data clusters in the large dimensional space, so that we get an overall view of the dataset using only a few different views.

## References


[1] L. Sirovich and M. Kirby, " A low–dimensional procedure for charecterization of human faces," *J. Opt. Soc. Amerr. A*, vol. 4, no. 3, pp. 519-524, 1987.

[2] L. Sirovich and M. Kirby, " Application of the Karhunen-Loeve procedure for the characterization of human faces," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 12, no. 1, pp. 103-108, 1990.

[3] M. Turk and A. Pentland, " Eigenfaces for recognition," *J. for Cognitive Neuroscience*, vol. 3, no. 1, pp. 71-86, 1991.

[4] P. Belhumeur, J. Hespanha and D. Kriegman, " Eigenfaces vs Fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Analysis Machine Intelligence*, vol. 19, no. 7, pp. 711-720, 1997.

Figure 1: Typical face from database

Figure 2: Mean of 143 faces

Figure 3: Image - mean of first face

Figure 4: Dominant eigenface

Figure 5: Second most dominant eigenface

Figure 6: Third dominant eigenface

Figure 7: The last eigenface, corresponding to the $143^{rd}$ eigenvalue.

Figure 8: Reconstruction using 40 eigenfaces

Figure 9: Reconstruction using 120 eigenfaces

Figure 10: Reconstruction using all 143 eigenfaces