Facial Recognition Using Eigenfaces

Grant Timmerman and Danny Vance
Math 136

May 20, 2013

Abstract

This paper examines the algorithm of using eigenfaces in recognizing faces in static images. By performing a mathematical process called principal component analysis (PCA) on a large set of images depicting different faces, we can discover the core "ingredients" - the eigenfaces, derived from statistical analysis of many pictures of faces. Combined with the average, the set of eigenfaces allows us to compose any human face. Remarkably, it does not take many eigenfaces combined together to achieve a fair approximation of most faces. Also, since a person's face can be recorded with a spanning set of eigenfaces plus the average face, rather than recording each individual pixel of each face, much less space is needed for each person's face. In other word, this algorithm can be applied in image data compression. This paper aims to determine the effectiveness of this technique with regards to accuracy, computational speed, and data compressions as well as to consider its drawbacks that make the technique ineffective in recognizing faces.

Key Words: Eigenfaces, Eigenvectors, Principal Component Analysis (PCA)

1 Introduction

Human face recognition is a difficult problem in computer vision. Human faces are complex, natural objects that tend to not have easily identifiable edges and features. Because of this, it is difficult to develop a mathematical model of the face that can be used as prior knowledge when analyzing a particular image.

Although facial recognition is a high level visual problem, there is quite a bit of structured information that we can take advantage of. By finding the most relevant pieces of information in a group of faces, we can characterize each face into these important features, which are called "eigenfaces". The process of recognizing a face is achieved through projecting the face's image into the subspace spanned by the eigenfaces ("face space") and then classifying the face by comparing its position in face space with the positions of known individuals.

2 Procedure

2.1 Eigenface Initialization

2.1.1 Setup images

Acquire an initial set of face images (the training set) in grayscale, cropped to only contain the face, and then rescaled to set dimensions. The photos must be centered and of the same size.

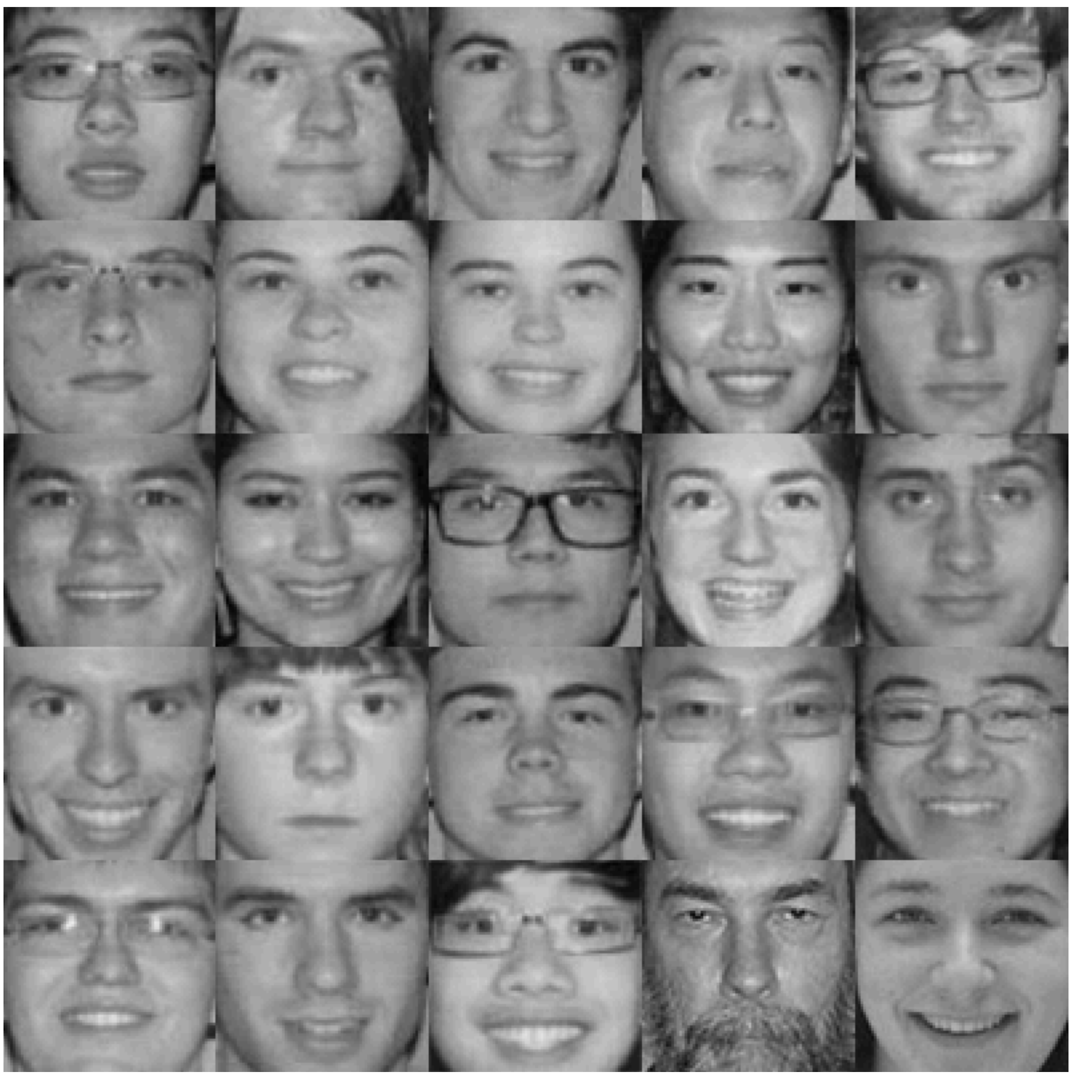

Figure 1: The face images we used as our training set.

2.1.2 Compute the Average Face

If the photos in the training set are each m by n, we can represent each photo as a flattened mn by 1 matrix (we'll call it v) that stores values for the brightness of each pixel in one very large list. The average face W is simply the arithmetic average of all of our M image vectors:

$$W = \frac{1}{M}\sum_{i=1}^{M} v_i$$

Here is the result from our training set:

Figure 2: The average face of the training set.

2.1.3 Find Difference Vectors

Subtract the average face vector W from each photo vector v to get a difference vector: $D_i = v_i - W$ where D_i is the ith difference vector.

We can align all of our M difference vectors into an mn by M matrix A where $A = \{D_1,\ D_2,\ D_3, \ldots\ D_M\}$ and each difference vector D_i represents the ith column of A.

Figure 3: The difference faces of the training set

As you can see, the luminosity of the pictures severely affects the difference vectors. Also note that Figure 3 does not fully represent the difference vectors because some values are negative yet are represented as black in this picture.

2.1.4 Find Eigenfaces

The set of difference vectors are very large and too computationally intensive if used. This is where principal component analysis comes in. We seek a set of M orthonormal vectors u_n and their associated eigenvalues λ_k, which best describes the data. These are the eigenvectors and eigenvalues for the covariance matrix defined by the following.

$$C = \frac{1}{M}\sum_{n=1}^{M} D_i {D_i}^T = AA^T$$

However the matrix C is N^2 by N^2, which is infeasible for large image sizes. Fortunately we can just compute the eigenvectors by first solving the much smaller M by M matrix and taking linear combinations of the resulting vectors. Here are the results of finding the eigenvalues for the smaller matrix:

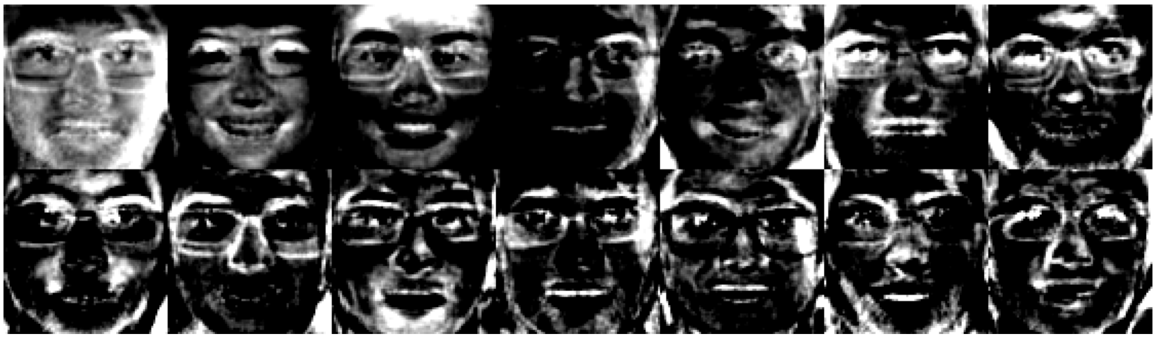

Figure 4: The eigenfaces of the training set. The most important facial component is on the top left and the least important facial component is on the bottom right.

Due note that the eigenfaces shown above are actually almost completely black and we have scaled the values by 5000%. Also note that the eigenvalues can have both negative so the visual representation is not fully representative in this visual model.

2.2 Eigenface Recognition

2.2.1 Project a photo to the face space

To project a photo vector onto the face space, we need to find the scalar coefficients that make up a reconstructed face. To do this we can simply multiply the difference vector for a particular image by the generated set of eigenfaces.

Once we have the coefficients, to create the new face we add the average face to the sum of the eigenface vectors (F_n) times their corresponding coefficients (k_n). The formula

$$I_r = W + \sum_{n=1}^{P} k_n F_n$$

Where P is the number of eigenvectors we choose to use.

Here are some examples of projected faces:

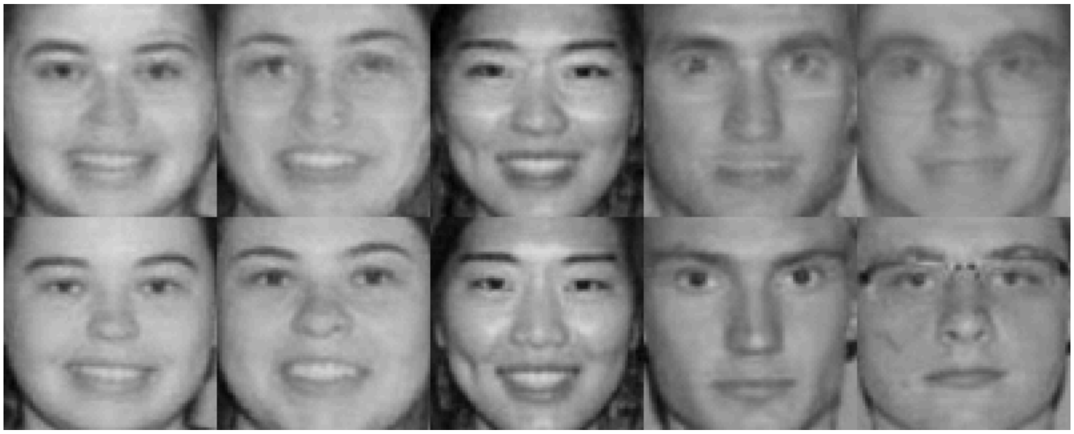

Figure 5: The reconstructed faces (top) compared to the original faces (bottom) using 14 eigenfaces.

The following are the percent errors of the reconstructed figures in Figure 5:

6.38312%	4.2757%	5.14499%	4.43696%	5.73563%

3 Drawbacks

Although this method of facial recognition is very quick and pretty accurate, there are some drawbacks, some of which include:

- Poor performance with different luminosity levels
- Poor performance with faces at different angles
- Smiles, hair, and other features can distort results
- The size of each person's head has to be normalized

4 Summary

WIP.

References

[1] Wim D'haes, Dirk van Dyck, Xavier Rodet, PCA-based branch and bound search algorithms for computing K nearest neighbors, Pattern Recognition Letters, Volume 24, Issues 9–10, June 2003, Pages 1437-1451, ISSN 0167-8655, 10.1016/S0167-8655(02)00384-7. (http://www.sciencedirect.com/science/article/pii/S0167865502003847)

[2] Alsamman, A., & Alam, M. S. (January 01, 2005). Comparative study of face recognition techniques that use joint transform correlation and principal component analysis. Applied Optics, 44, 5, 688-92.

[3] Hancock, P. J. (January 01, 2000). Evolving faces from principal components. Behavior Research Methods, Instruments, & Computers : a Journal of the Psychonomic Society, Inc, 32, 2, 327-33.

[4] Hyun-Chul Kim, Daijin Kim, Sung Yang Bang, Face recognition using the mixture-of-eigenfaces method, Pattern Recognition Letters, Volume 23, Issue 13, November 2002, Pages 1549-1558, ISSN 0167-8655, 10.1016/S0167-8655(02)00119-8. (http://www.sciencedirect.com/science/article/pii/S0167865502001198)

[5] Hancock, P. J. (January 01, 2000). Evolving faces from principal components. *Behavior Research Methods, Instruments, & Computers : a Journal of the Psychonomic Society, Inc, 32,* 2, 327-33.