Design and Evaluation of a Eigenface-based Face Recognition System

Shiyu Luo (SLN8503), Qianying Yu(QYJ4977), Xinda Huang(XHV2881)

Abstract

In this report we describe a face recognition based on PCA. The method consists of two phase: in the training phase, training face image vectors are decomposed into a small set of characteristic eigenvectors via Principal Component Analysis (PCA); in the testing phase, probe (testing) images are represented by a weighted sum of these eigenvectors and these weights are matched against those of enrolled individuals (training data). The performance of the system is then evaluated by the examination of the genuine and imposter distributions, the ROC curve, and the CMC curve.

Introduction

Faces are our major focus of attention in social activity, playing a significant role in identity and emotion. Humans ability to identify faces is remarkable. Computational models of face recognition are of special interest particularly because they can be applied to various problems, including criminal investigation, image and video processing, security systems, and human-computer interactions.

For 2-D face recognition problem, the problem with the image representation is its high dimensionality. A 2-D $p \times q$ grayscale images span a $m = pq$-dimensional vector space, so an image with 100×100 pixels span a 10000-dimensional image space already, not to mention today's much bigger RGB images. However, dimensions are not equally important for us. The PCA was proposed to turn a set of possible correlated variables into a smaller set of uncorrelated variables. The idea is that a high-dimensional dataset is often described by correlated variables, and therefore only a few dimensions account for most of the information. The PCA method finds the directions with the greatest variance in the data. Later this method is applied to face recognition by Turk.

In this report we follow closely the paper published by Matthew Turk and Alex Pentland [1]. The scheme discussed by paper is based on the PCA method that decomposes face images into a small set of characteristic vectors called"eigenfaces", i.e. the principal components of face image space; they are not necessarily corresponding to human-recognizable features such as eyes, noses and ears. An individual face is characterized by a weighted sum of the eigenface features, and so to recognize a face it is only necessary to compare its weights with those of known individuals.

Method

Algorithm Description

Let $x_i = \{p_1, p_2, ..., p_n\}$ be a face image vector. Suppose we have M training images. 1. Compute the mean face μ

$$\mu = \frac{1}{M}\sum_{i=1}^{M} x_i$$

2. Normalize the training images by subtracting the mean face from their image vector. Let X_i be the mean centered image vector of x_i

$$X_i = x_i - \mu$$

3. Compute the covariance matrix S of training images

$$S = \frac{1}{M}\sum_{i=1}^{M}(X_i - \mu)(X_i - \mu)^T$$

4. Compute the eigenvalues λ_i and eigenvectors v_i of S

$$Sv_i = \lambda_i v_i, i = 1, 2, 3, ..., n$$

5. Order the eigenvectors descending by their eigenvalues. The k principal components are the eigenvectors are corresponding to the k largest eigenvalues.

The k principal components of an observation x is then given by

$$y = W^T(x - \mu)$$

where $W = (v_1, v_2, ..., v_k)$. We will treat y as the feature vector of image vector x. 6. The matching score of two images is defined as the distance between two images, i.e., the euclidean distance of their feature vectors. Suppose we have 2 image vector x_1 and x_2, with feature vectors W_1 and W_2, the matching score of x_1 and x_2 is

$$score(x_1, x_2) = ||W_1 - W_2||$$

Data Preparation

Images of first 50 subjects with frontal view and normal expression are used to construct the training set and testing set. For each selected subject, two such images are used, one for the training set and the other for the testing set. Each image is converted to .jpg format and is resized to $46 \times 70 = 3220$ pixels for computational efficiency. Because the PCA method requires all input vectors are of the same dimensionality, face detection and crop is not conducted since they will seriously affect the image size.

The prepared data is in folder `testing_data` and `training_data`. To reproduce the result when the number of eigenvectors is 100, please execute `main.py`.

Determination of Threshold

Since the “score” here is actually a eulicdean distance, it is critical to determine the range of scores to further determine thresholds. We first calculate the genuine and imposter scores and plot a histogram to observe the score distribution, and then we determine the range of threshold based on the observation.

Experiment

Genuine and Imposter Distribution

The distributions are shown in Figure. Blue curves and bins show the distribution of genuine scores, and orange ones show the imposter scores. The two distributions become more separable when the number of eigenvectors is no less than 75.

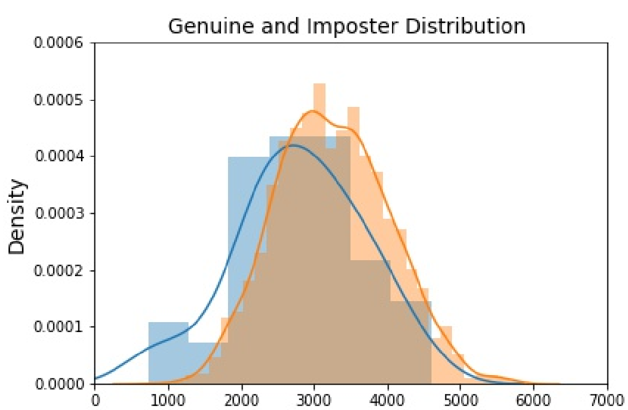

Figure 1: Genuine and imposter distribution, 10 eigenvectors

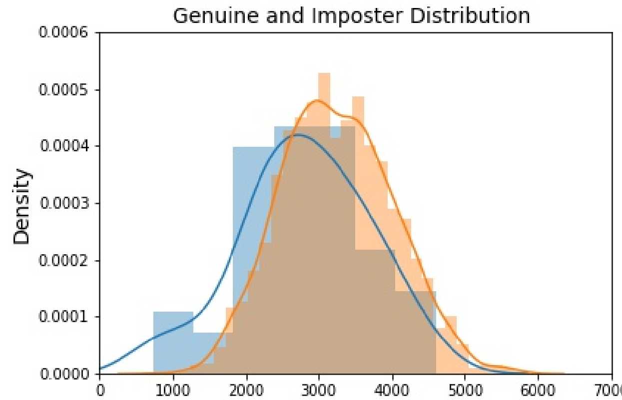

Figure 2: Genuine and imposter distribution, 50 eigenvectors

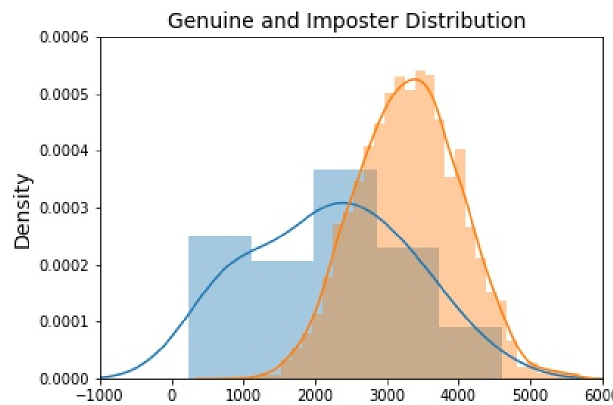

Figure 3: Genuine and imposter distributions, 75 eigenvectors

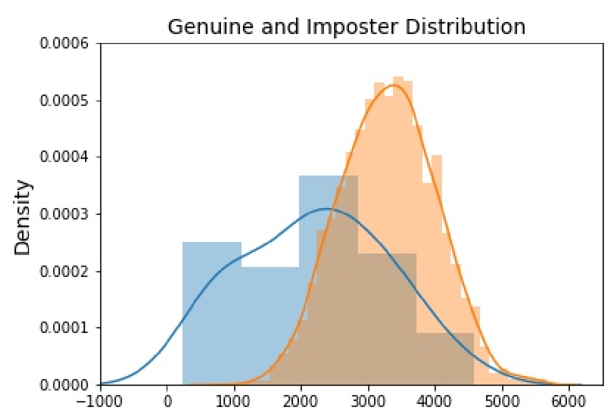

Figure 4: Genuine and imposter distributions, 100 eigenvectors

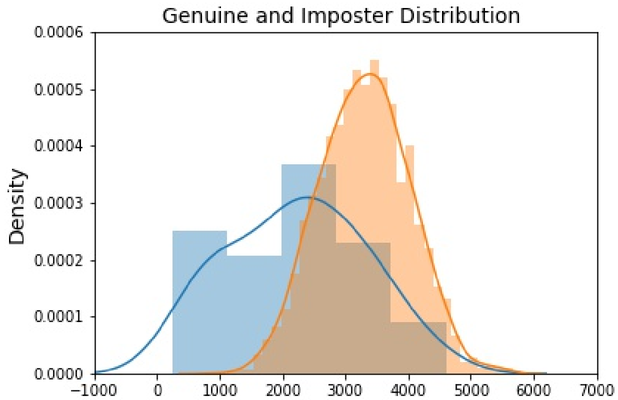

Figure 5: Genuine and imposter distributions, 150 eigenvectors

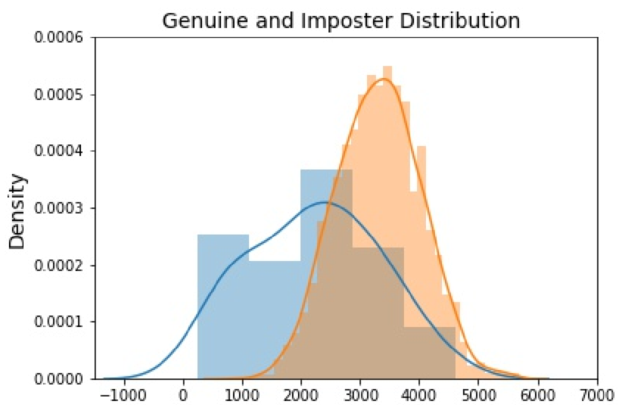

Figure 6: Genuine and imposter distributions, 200 eigenvectors

To illustrate the result of genuine and imposter distribution clearly, we truncate the proportion from 95% to 100%,

ROC Curve

The ROC curves with different numbers of eigenvectors are shown in Figure 7, 8, 9, 10, 11, and 12. The performence of the system is improved greatly with the number of eigenvectos used increases when it is less than 75. When the number of eigenvectors used is greater than 75, the improvement is insifignificant. It shows that 75 eigenvectors is sufficient to represent the image space, given the augmented data.

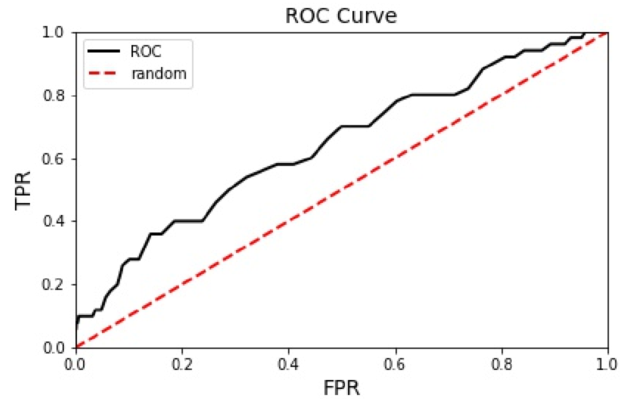

Figure 7: ROC curve, 10 eigenvectors

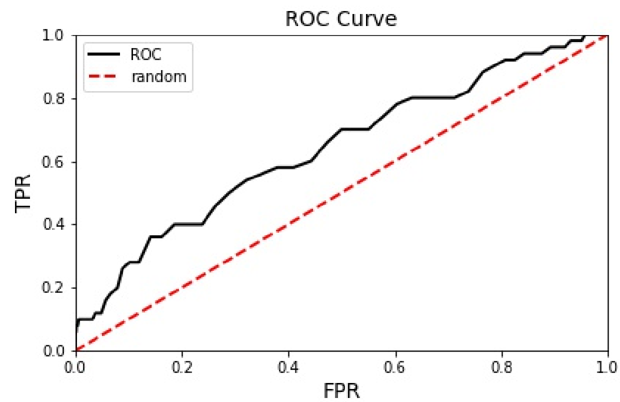

Figure 8: ROC curve, 50 eigenvectors

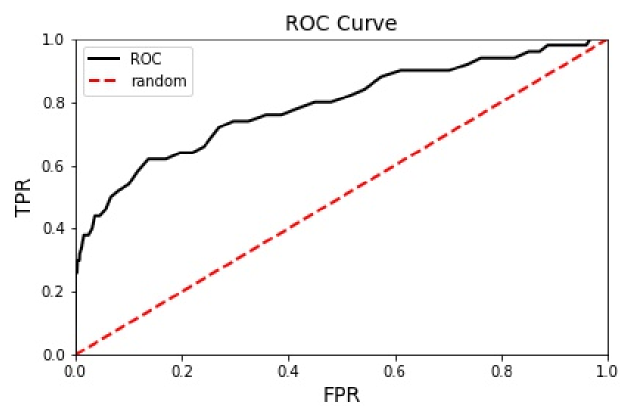

Figure 9: ROC curve, 75 eigenvectors

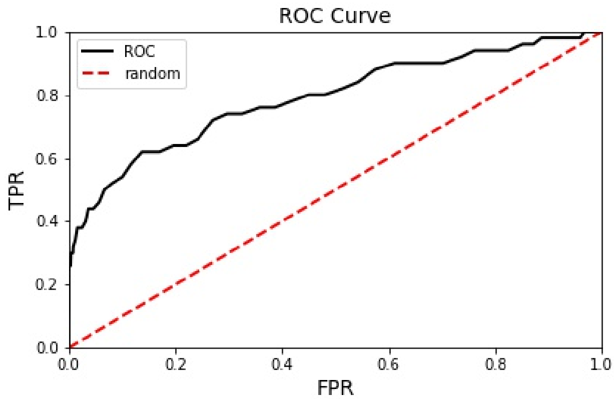

Figure 10: ROC curve, 100 eigenvectors

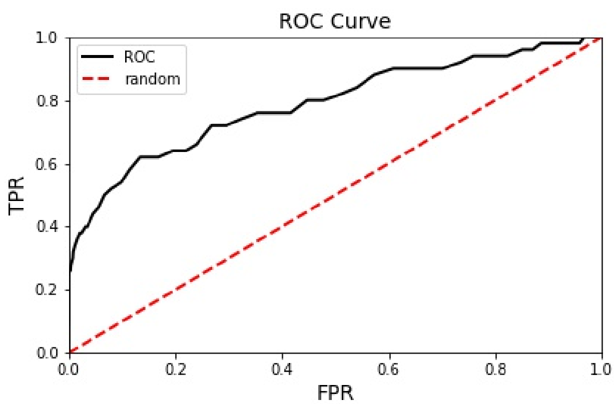

Figure 11: ROC curve, 150 eigenvectors

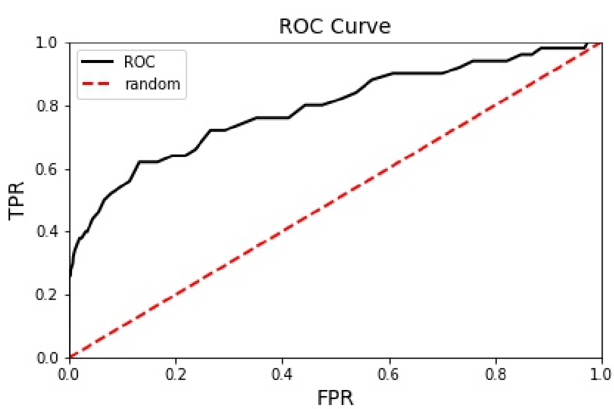

Figure 12: ROC curve, 200 eigenvectors

CMC Curve

CMC Curves with different numbers of eigenvectors are shown in Figure 13, 14, 15, 16, 17, and 18. Overall, the accuracy of the system increases as the number of eigenvectors used increases, especially when it is under 100 eigenvectors. When it is greater than 100, the improvement in accuracy is insignificant. It shows that 100 eigenvectors is sufficient to represent the image space, given the augmented data.

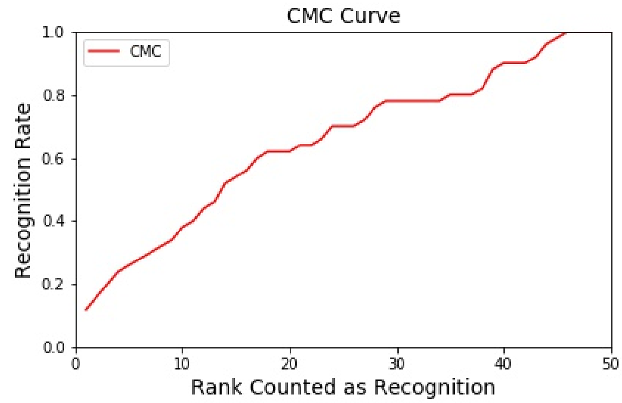

Figure 13: CMC Curve, 10 eigenvectors

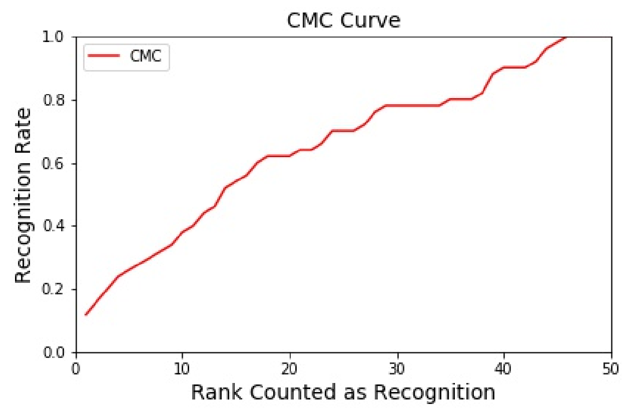

Figure 14: CMC Curve, 50 eigenvectors

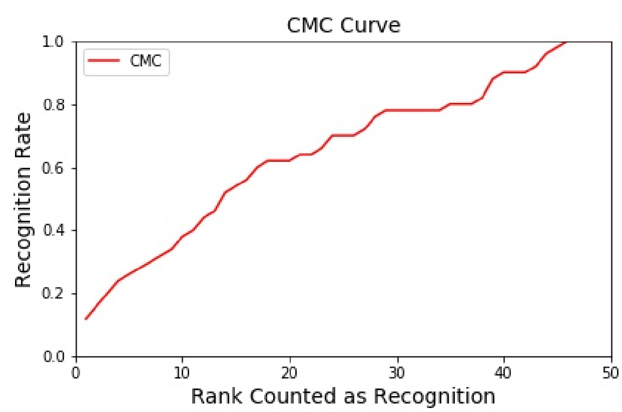

Figure 15: CMC Curve, 75 eigenvectors

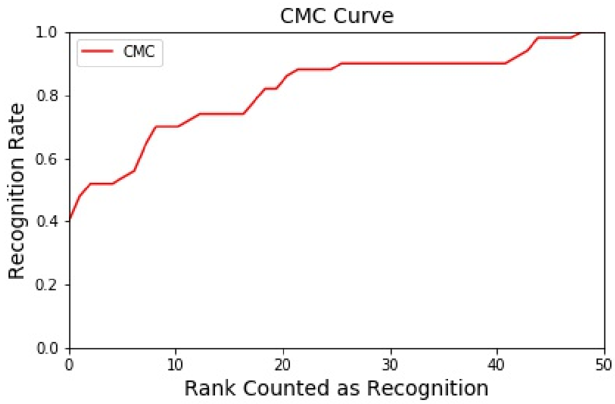

Figure 16: CMC Curve, 100 eigenvectors

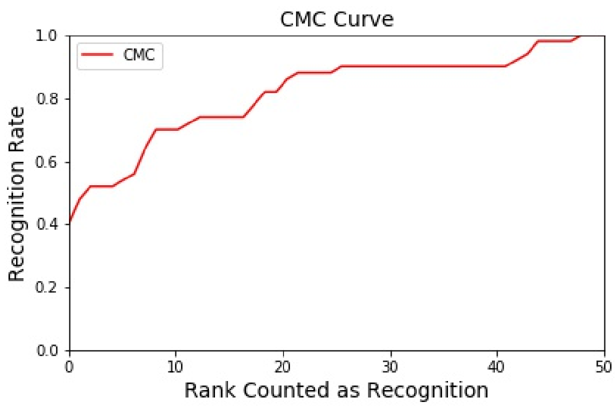

Figure 17: CMC Curve, 150 eigenvectors

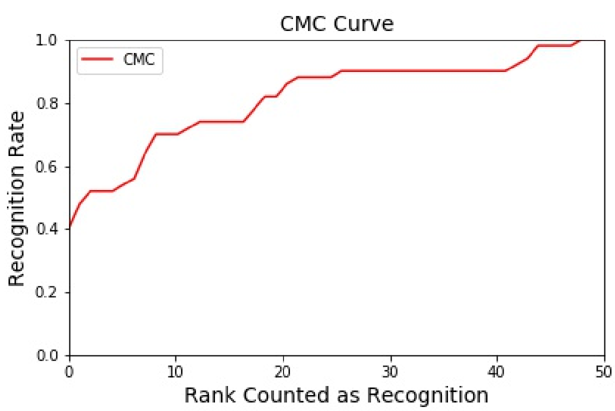

Figure 18: CMC Curve, 200 eigenvectors

Conclusion

In this report, PCA is used for dimensionality reduction to ensure computational efficiency and noise surpression. The experiment results shows that based on our data, 100 eigenvectors are sufficient to represent the image space.

Acknowledgement

Sincerest thanks to our course instructor Professor Xin Chen and our course TA Andi Zang, for making this project successful.

References

Turk, Matthew, and Alex Pentland. "Eigenfaces for recognition." Journal of cognitive neuroscience 3.1 (1991): 71-86.