Face Recognition with Eigenfaces

by

Haonan Wu

N18859539

New York University

Dec, 2017

E.D Wong

Table of Contents

Chapter 1

Introduction

The Principal Component Analysis (PCA) was independently proposed by Karl Pearson (1901) and Harold Hotelling (1933) to turn a set of possibly 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 meaningful dimensions account for most of the information. The PCA method finds the directions with the greatest variance in the data, called principal components.

This project is to implement PCA by Python.

Chapter 2

Environment Description

- Window 10
- Virtual Studio 2015
- Python 2.7
  (1) numpy
  (2) matplotlib

Chapter 3

Algorithmic Description

step 1 Let $X = \{x_1, x_2, \ldots, x_n\}$ be the training set's matrix with $x_i \in \{[0, 255]\}^d$.
d is the size of the image. In this case, $d = 45045$.

step 2 Compute the mean

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

.

step 3 Compute the the Covariance Matrix

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

. The size of S in this case is 9×9.

step 4 Compute the eigenvalues λ_i and eigenvectors v_i of

$$Sv_i = \lambda_i v_i, i = 1, 2, \ldots, n$$

step 5 Order the eigenvectors descending by their eigenvalue. The k principal components are the eigenvectors corresponding to the k largest eigenvalues.
In this case, instead of using the integer k to truncate the eigenvalue, I use the propotion of λ_i in $\sum \lambda_i$ to truncate the eigenvalue. And the threshold is 0.7.
In other word, if the sum of kept eigenvalues is larger than 0.7, we throw the rest

one.

step 6 The k principal components of the test set x are then given by:

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

where $W = (v_1, v_2, \ldots, v_k)$.

Finding the nearest neighbor between the projected training images and the projected test image.

Chapter 4

Result

(a) test image (b) result image (c) test image (d) result image

(e) test image (f) result image

Figure 4.1: Label 1

For the first face, all three test image are detected correctly. But if the threshold for the k largest eigenvalues is larger than 0.8, the centerlight image will be detected incorrectly.

(a) test image (b) result image

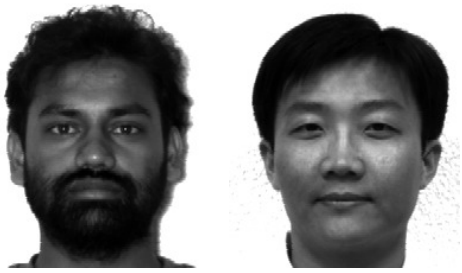
Figure 4.2: Label 2

For 2nd face, the test image is also the train image, it should be correctly.

(a) test image (b) result image

Figure 4.3: Label 3

For the 3rd face, the test image is also the train image, it should be correctly.



(a) test image (b) result image

Figure 4.4: Label 7

(a) test image (b) result image

Figure 4.5: Label 7

(a) test image (b) result image

Figure 4.6: Label 7

For the 7th face, only the training image can be detected. And I have tried every possible threshold for truncating k largest eigenvalues. And I can't predicted correctly.

And I also use the happy image or the centerlight image to train the PCA, and only the training image can be detected, and another two of the images can't be predicted correctly.

(a) test image (b) result image

Figure 4.7: Label 10

The test image is also the train image, it should be correctly.

(a) test image (b) result image

Figure 4.8: Label 11

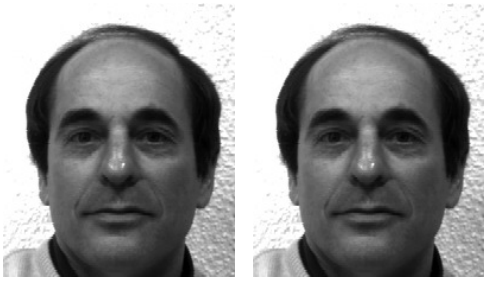
(a) test image (b) result image

Figure 4.9: Label 11

(a) test image (b) result image

Figure 4.10: Label 11

For the 11th face, all three test image are detected correctly.

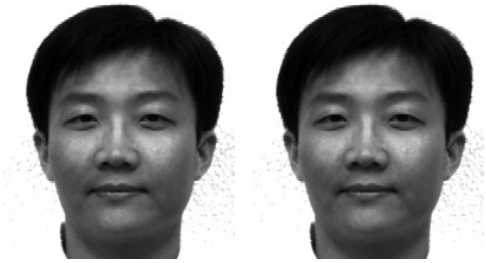


(a) test image (b) result image

Figure 4.11: Label 12

The test image is also the train image, it should be correctly.

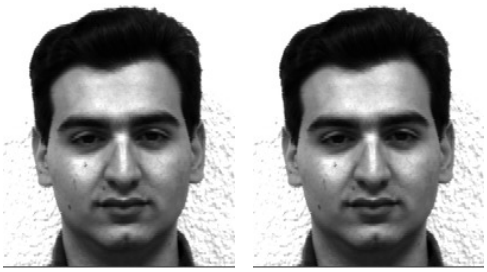
(a) test image (b) result image (c) test image (d) result image



(e) test image (f) result image

Figure 4.12: Label 14

For the 11th face, all three test image are detected correctly.



(a) test image (b) result image

Figure 4.13: Label 15

For the 15th face, the test image is also the train image, it should be correctly.

(a) test image (b) result image

Figure 4.14: Label 0

The test image is an apple, and I get a result and I don't know what kind of result is better.