Structured Face Recognition with Images

Miguel Valdez
Boston University CS Department

mvg35@bu.edu

Abstract

Images in a data structured setting can easily be though as array of numbers that in theory can be compared to different arrays of numbers. With this consideration in mind, how hard is it to use the mathematical context of an image to identify the difference between those two images. In this project we tackle this question using different mathematical procedures in order to create a useful and reliable face identifier.

1 Background

1.1 Method

Eigenfaces is a method of face recognition based on Principal Component Analysis. PCA is a method of transforming a number of correlated variables into a smaller number of uncorrelated variables. PCA decomposes an image into a set of eigenvectors. The PCA basis vectors are learnt by unsupervised training. PCA can be applied to the task of face recognition by converting the pixels of an image into a number of eigenface feature vectors, which can be compared to measure the similarity of two faces in the images.

Training images where taken and sized 64x48 loaded in grayscale, only the face is shown. Each image is converted into a column vector and then the images are loaded into a n by m matrix, n being the number of pixels in each image and m is the number of images.

Main idea behind eigenfaces is to suppose a α is a $N^2 * 1$ vector corresponding to $N * N$ face image I.

The idea is to represent α into a low-dimensional space:

1.2 Training

Steps are
Calculating the mean of images
Subtracting the mean from the images to get a new image with the new propagation
Calculate the eigenvectors and eigenvalues of new image
Retain only the eigenvectors with the largest eigenvalues
Project the mean-shifted images into the eigenspace

Each face (minus the mean) ω i in the training set can be represented as a linear combination of the best K eigenvectors:
$\omega_i - mean = \sum_{j=1}^{k} w_j u_j, (w_j = u_j^T)$

1.3 Classification

The similarity can be calculated by finding the Euclidean distance between their corresponding feature vectors and the smaller the distance between the feature vectors, the more similar the faces.

Submitted to 31st Conference on Neural Information Processing Systems (NIPS 2017). Do not distribute.

2 Results

2.1 Cases

Testing cases where chosen as: one picture that goes along the testing data, a picture of a random subject which shouldn't go along the testing data, and an extra picture of another subject which shouldn't go along either.

Figure 1: Different cases for testing. My portrait. Random Portrait. Artist Portrait

Results where as expected, origianl picture getting a result of .20 while other two struggles to surpass the .05 mark. Making this method quite reliable under the implemented variances.

Figure 2: Results after algorithm

2.2 Improvementes

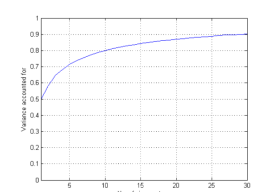

Figure 3: Results after algorithm

Increasing the number of eigenvectors generally increases recognition accuracy but also increases computational cost. Note, however, that using too many principal components does not necessarily always lead to higher accuracy, since we eventually reach a point of diminishing returns.

A big motivation for this project is security and ways in which this technology can be applied. In future changes, it would be nice to implement a dual system that implements OpenCV functionalities in order to simulate this process with a live camera feed or similar. This would allow for further applications.

References

[1] M. Turk and A. Pentland, (1991) "Eigenfaces for Recognition", Journal of Cognitive Neuroscience, vol. pp. 71-86 $http://www.vision.jhu.edu/teaching/vision08/Handouts/case_study_pca1.pdf$