Archit Dhanani

adhanani

12-659 Section A2

Face Recognition using Principal Component Analysis

November 28, 2016

Introduction

The major focus of this project is Face Recognition using image processing. Reliability and accuracy of face recognition from the image is a topic of extensive research. Face recognition is an integral part of many sophisticated devices such as authentication, access control and other security systems. This project uses unsupervised machine learning algorithm to implement face recognition.

Face detection means to detect whether there is a face in the given image and then in face recognition we try to identify the person.

1. Background

There are many algorithms to recognize a face in the image such as PCA (Principal Component Analysis), ICA (Independent Component Analysis), LBP (Local Binary Pattern), etc. The solution should also consider the possible variation in the images such as illumination changes, pose variation, clustering, Occlusion and so on.

Component Analysis is a technique to identify specific feature from the data using an unsupervised approach. The goal of this method is to process a high dimensional data and representing it in a low dimensional sub space by reduced dimensionality, data compression and data ordering. This helps to improve the speed of execution. This is technique is widely used in the applications where the data is noisy, high dimensionally and where the data itself is very huge.

PCA is the common form of the Factor analysis. The image is represented in terms of features in different vectors. Vectors are then converted to principal components. These principal components can be represented in linear combinations of the original features. They are independent and each is represented by a eigen values to minimize the variations in the images.

We try to compute the eigen faces. Eigen Faces looks like a bland androgynous average human face. When properly weighted, eigen faces can be summed together to create an approximate gray scale rendering of a human face. This help to give a fair likeness of most people's faces. Since number of Principal Components are less than that of the original number of features in the data, this helps to reduce the unnecessary computation. Eigen Faces is the way to implement data compression to faces for identification.

The drawbacks of this methods are

1. It cannot recognize the faces when there is high variation in the contrast and brightness.
2. It cannot recognize the face when the head orientation is different for different images.
3. Also, the algorithm fails when there is a drastic change in the expression of the person.

2. Implementation

The steps involved in solving the problem will be as follows:

1. Any Gray scale image of the faces will be sufficient for the training set.

 [3]

2. Formulating the eigen faces according the data sets available by finding the average training set is defined by

   $\Psi = (1/M) \sum_{i=1}^{M} \Gamma_i$,

   Where,

   Ψ: Mean value of the data set

   Γ_i: Vectors of each pixel in the Data sets

3. Finding the covariance of the standard deviation.

   Covariance of the data set = $(X- \Psi)^T * (X- \Psi)$

4. Using the covariance matrix, we calculate the eigen values and the eigen vector. For this matlab has its own inbuilt function eig().

   V = eig(covariance)

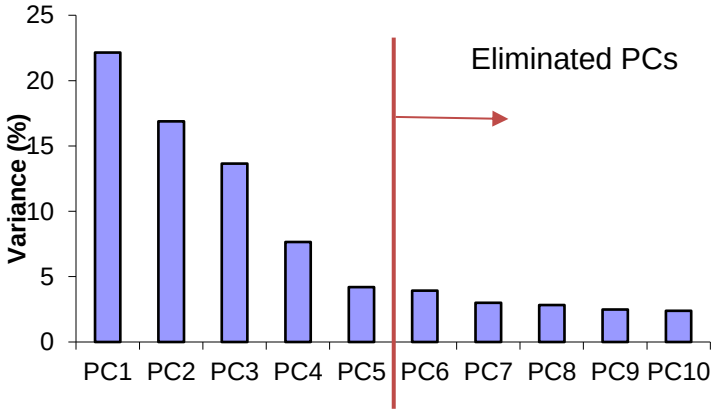
5. Using the eigen vectors to form the eigen face.

$$\mathbf{x} \rightarrow (\underbrace{(\mathbf{x}-\overline{\mathbf{x}})\cdot\mathbf{v_1}}_{a_1}, \underbrace{(\mathbf{x}-\overline{\mathbf{x}})\cdot\mathbf{v_2}}_{a_2}, \ldots, \underbrace{(\mathbf{x}-\overline{\mathbf{x}})\cdot\mathbf{v_K}}_{a_K})$$

$$\mathbf{x} \approx \overline{\mathbf{x}} + a_1\mathbf{v_1} + a_2\mathbf{v_2} + \ldots + a_K\mathbf{v_K}$$

   Ie Eigenfaces = x_bar * v

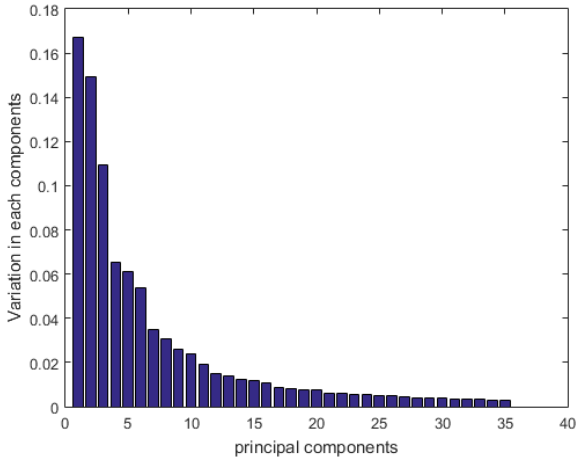
6. The principal components(PCs) with lower variation are eliminated and this helps to reduce the dimensions in the problem.



7. Now the test image is converted into the given formulation.
8. If the deviation of the test image as per the new dimensionality fits in the boundaries given for the original image, then the algorithm recognizes the images. [1]

Results

- After calculating eigen vector, only 35 eigen vectors show the variance greater than 90%.



- These vectors are used to find the eigen faces. The following is the eigen faces with the greatest variance.

- By reconstructing the images used to train the model, I got the initial image back.

- The accuracy of the model is 80% when tested on the 30 random images from the data set.

Conclusion

- Principal Component Analysis helped to understand the reduction in dimensionality of the dataset.
- One of the drawbacks of the method is that it requires all the data set in same size.
- Nearest neighbor algorithm is used for classification of the images.
- There are many other algorithms used to classify the images like logistic regression, Linear Discriminant Analysis(LDA), etc. I will try to implement these algorithms to improve the accuracy of the model.

References

1. Lindsay I Smith, “A Tutorial on Principal Component Analysis” 2002
2. Jonathon Shlens, “A Tutorial on Principal Component Analysis”
3. Belhumeur, P., P.Hespanha, J., Kriegman, D.: Eigenfaces vs. fisherfaces: recognition using class specific linear projection. IEEE Transactions on Pattern Analysis and Machine Intelligence 19 (1997) 711–720