Face Recognition for Single & Different Facial Expressions

Kedar Yadav[1] Dr. Namit Gupta[2]
[1]PG Scholar [2]Assistant Professor & Head
[1,2]Department of Electronics & Instrumentation Engineering
[1,2]Shri Vaishnav Institute of Technology & Science, Indore, Madhya Pradesh, India

Abstract— In this paper presents and analyses the performance of Principal Component Analysis (PCA) based technique for face recognition. The face is our main hub of thought in common intercourse, in concert a major part in assigning identity and feeling. We can distinguish thousands of faces learned throughout our lifetime and identity recognizable faces at a look even after years of division. This skill is give up healthy, regardless of large changes in the visual incentive due to presentation conditions, appearance, aging, and distraction such as spectacles, beards, changes in hairstyle. Through human faces are complex in shape, face recognitions is not difficult for a human brain whereas for a computer this job is not easy. We think recognition of human faces with two facial appearance: single and differential. The images that are captured previously constitute the training set. From these images eigenface are calculated. The image that is going to be recognized through our system is mapped to the same Eigen spaces. Next I used classification technique namely distance based used to classify the images as recognized or non-recognized. Currently I get result for the single facial expression now I am operational for dissimilar facial appearance.

Key words: Eigen Faces, Principal Component Analysis, Face Recognition

I. Introduction

Face is an essential element of focus of our daily life. We convey our identity and emotion through our face and different expressions of face respectively. The human ability to recognize face is remarkable. Through human faces are complex in shape, face recognition is not difficult for a human brain whereas for a computer this job is not easy. The complexity of recognition is prominent and several algorithms are reported in literature that could achieve the recognition with high degree of accuracy. Face recognition system is widely used in different areas that include (a) criminal record and identification, (b) robot vision, (c) security system, (d) human computer interaction, (e) image and field processing.

Face recognition system is divided into two categories, (a) appearance based and (b) component based. For appearance based, we think the holistic characteristic or the complete face image as our aspect for detection. On the additional hand, in component based face detection, we judge geometrical association of different components of face such as eye, nose, lip etc. as the features of a recognition system. Principal component analysis (PCA) is a quick and well-organized system that is extensively used for appearance based face recognition. This technique is also used for dimensionality reduction in different areas that include image processing, signal processing and data mining. This technique is sometimes also called eigenface. The eigenface approach is chosen for this study considering its capability of recognizing real time images where orientation, illumination and distortion are continuously changing. This work focuses on how the image with real time attributes affect the recognition feature of eigenface technique. Our primary objective for this research is to minimize the complexity in calculation for bigger matrices. For example, if we have 120 picture with the size of (180×200), we will have a very big number while calculating the one dimensional vector from 2D matrix (by calculating 180×200×120) which is a very big number. By using the eigenvectors, we could minimize the use of all the images and reduce them for example 40 pictures which will also bring down our total calculation to (180×200×40). Though we are using lesser amount of data, we will still get the same level of accuracy. Beside we could make the size even smaller by changing the order of matrix multiplication which in turn reduces the principal components, and the end we could work only on (40×120) matrix with the same level of accuracy.

Rest of the paper is organized as follows. Section 2describe the methodologies used in this research in detail. Section 3 describes about the system and execution flows of different components in the system. Section 4 presents and analyzes the result. Finally section 5 concludes and gives direction of future research.

II. Methodologies

Our face recognition system consists of several steps. Each of the steps is described in detail in below:

A. Initialization & Finding Principal Components

At first we take images. These images are nothing but the matrix which has pixel intensity at different rows and columns. This image could be viewed as a vector also. If an image has height, h and width, w, then we could formulate this image as w vectors, where each vector has h dimensions. The rows of the images are placed one after another like the figure below:

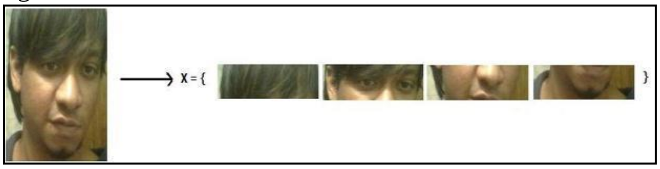

Fig. 1: Formation of the Faces Vector from Faces Images

The vector which is represents our image and this image has a certain space so this is called image space. If we have N images, we have image space dimension as N× w× h. In this image space all images are represented by w by h pixels. These images under same image space look like each other. They have two eyes, a nose, a mouth etc. located at the same image space.

Now we will build the face space from the image space. The main task of building a face space is to describe the face images. The basis vector of this space is called principal component. The dimension of the face space will be M× w× h. In the face space all pixel is not relevant and each pixel depends on the neighbors. So the dimension of face space is less than the dimension of the image space. We could find the principal components of the face by finding the eigenvectors of the covariance matrix of the set of face image. This eigenvectors are basically a set of feature which characterize to the maximum variations between face images. Each of this images that comes from the image space contribute more of less to the eigenface. So we can display eigenvector as a sort of ghostly faces which we call eigenface. Actually eigenface do not exist in real world. We could not say we can build or create eigenface of a particular image face which is in the image space. Eigen face actually is an imaginary face which is a combination of all the image with in a particular image space.

Presents eight eigenface from a sample image database presented in fig.2

Fig. 2: Eigenface from the Image Database

We present the mathematical formulation of eigenfaces below.

1) We obtain N training images I1,I2,….IN. Each of these images have dimension w × h. Convert these images into vector space by concatenation. After the concatenation a matrix is converted to a vector.
2) Represent each image I, which its corresponding vector λ_i .

$$\begin{bmatrix} B_{11} & B_{12} & . & B_{1h} \\ : & : & : & : \\ B_{w1} & B_{w2} & : & B_{wh} \end{bmatrix} \xrightarrow[\text{concatenation}]{} \begin{bmatrix} B_{11} \\ : \\ B_{1h} \\ : \\ B_{wh} \end{bmatrix} \triangleq \lambda_i$$

3) Calculate the mean face vector ω by the following equation

$$\omega = \frac{1}{N}\sum_{i=1}^{N} \lambda_i$$

Subtract the mean face, ω from each face vector, λ_i to get a set of vector μ_i.

$$\mu_i = \lambda_i - \varpi$$

The purpose of subtracting the mean image from each image vector is to keep only the distinguishing features from each face by removing the common information.

Find the covariance matrix C by the following equation:

$$C = A^T A \text{ Where, } A = [\mu_1, \mu_2, \ldots\ldots \mu_N]$$

Find the eigenvalues and eigenvectors for the covariance matrix C. Sort the eigenvectors according to the eigenvalues. Take the first M eigenvectors that have higher eigenvalues. Now each eigenvector will have N×1 dimension. Let us name those eigenvectors as η_i.for I=1, 2…M.

B. *Projection of New Face to Eigenface*

When a new image is encountered, calculate the set of weights based on the new or input image and the M eigenface by projecting the input image onto each of the eigenface. The mathematical formulation is given below:

Let us consider the new image as I_{new}

Find out the M eigenface components, Ψ_l,, by projecting the new image

$$\Psi_l = \gamma_l^T (I_{new} - \varpi)$$

Where

$$\gamma_l = \sum_{k=1}^{N} \eta_{lk}\, \mu_k \text{ , for } l=1, 2\ldots.M$$

Create a new feature vector, Ω_{new} for the new image by concatenating eigenface component, Ψ_1

$$\Omega_{new} = [\Psi_1, \Psi_2, \ldots \ldots . \Psi_N]$$

C. *Face Recognition by Classification Algorithms*

The last step of the face recognition system is to identify the new face to be recognized or not recognized. If the face is recognized the system will tell the person's name for whom the face has been recognized. In the other word, if we have N persons in the image database, we say that there are N classes where each individual person representing a class. There are two algorithms used for classification, one is distance based and the other one is neural network based classification.

The distance based classifier works in the following way for each image in the image database, find out the feature vector Ω_i , for N persons where i=1, 2…N. The procedure will be same that is discussed for the new image in the earlier sub section .Comparison is done by the Euclidian distance between two features, Ω_{new} and Ω_i , if the distance is less than some predefined threshold, t, we say that the image is recognized. The class of the new image will be one that has the least Euclidian distance with the new image, providing this distance is less than the threshold.

III. SYSTEM DESCRIPTION

The system by using MATLAB 2017a (version 9.2). MATLAB is a high level language for technical computing. It also mathematical operation solving for linear equation, Fourier analysis, 2d and 3d graphics data. These are follows:

1) *Image Reading*

MATLAB can easily read an image and convert the image in a certain matrix. We can use the image matrix for our related work. MATLAB can read an image of 8 bit up to 32 bit.

2) *Image Conversion from RGB to Grayscale*

MATLAB can convert an image from RGB to grayscale. This computational task can be done by MATLAB command. If RGB image in 32 bit it represent red for 8 bit, green for 8 bit, blue for 8 bit.

3) *Image Resize*

MATLAB command can be used to resize a certain image in to any size that MATLAB allow.

4) *Convert Matrix to 1 Dimensional Vector*

We can use certain technique to convert a matrix to 1 dimensional vector which helps to compute the desired output.

5) *Matrix Transpose*

In MATLAB we can easily transpose a certain matrix. In our system, at first of the system take the images. These images are captured by webcam or other capturing source. We calculate the mean image and eigenface.

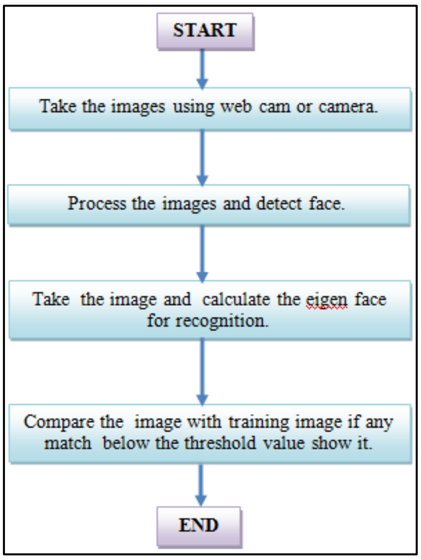

Fig. 3: Basic Flow Diagram of the Face Recognition System

Than we take the input image that will be detected later on. We process the input image and compare the input image with training image set if any match below the threshold value than we can say it is recognized otherwise not. The detail flow of execution is given in fig. 4

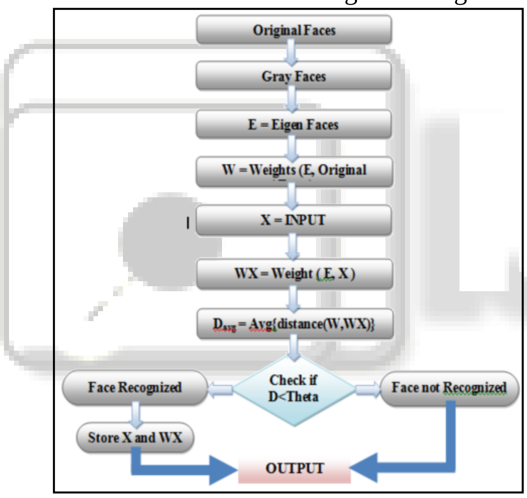

Fig. 4: Detail Flow of Execution in the System

IV. RESULTS & ANALYSIS OF RESULTS.

In total, six test cases were chosen to test the performance of face recognition system. All the images were in same size. The training set images were captured such that the person's faces were frontal with minimal head tilt and head decant varied expressions. Tests were taken in well illuminate area to recognize, at test, the known images. We have to do 2 type of testing. 1. Face image with single facial expression.2. Face image with different facial expression.

Fig. 5 and Fig. 6 present some sample image for single and different facial expression respectively.

Fig. 5: Sample Dataset for Single Facial Expression

Fig. 6: Sample Data Set For Different Facial Expression

We Apply Algorithms For Six Test Cases The Test Cases Are Described Below.

1) Test Case 1: Test Case 1 Is Measuring The Impact Of Having Large Number Of Training Images Of A Small Group Of Persons. Altogether 30 Images Of 3 Persons Were Taken To Create The Training Set. Different Facial Expression Of Each Person Is Taken For This Case.
2) Test Case 2: Test Case 2 Is Measuring The Impact Of Having Large Number Of Training Images From A Large Group Of Persons. Altogether 50 Images Of 50 Persons Were Taken To Create The Training Set. The Images Are For Single Facial Expression.
3) Test Case 3: Test Case 3 Is Measuring The Impact Of Having Small Number Of Persons In The Training Set. In Total 6 Images From Each Of The 2 Persons Were Considered To Create The Training Set. So The Total Number Of Images Is 12.
4) Test Case 4: Test Case 4 Is Measuring The Impact Of Having Small Number Of Images In The Training Set. In Total 20 Images From 20 Persons Were Considered To Create The Training Set.
5) Test Case 5: Test Case 5 Is Measuring The Impact Of Having Very Small Number Of Images In The Training Set. Altogether 3 Images Of 3 Persons Were Taken /To Create The Training Set.
6) Test Case 6: Test Case 6 Is Measuring The Impact Of Having Small Number Of Images In The Training Set. Altogether 3 Images Of 10 Persons Were Taken To Create The Training Set.

Table I Summarizes All the Test Cases and Number of Images in Total Used For Recognition System. The Accuracy Reported Here Is The Accuracy For Distance Based Classifier.

Test Case	Image/ Person	No Of People	Total Images	Total Testing Images	Accuracy
Case 1	3	10	30	50	80%
Case2	1	50	50	70	100%
Case3	6	2	12	24	37.5%
Case4	1	20	20	40	80%
Case5	1	3	3	5	95%
Case6	3	10	30	36	83%

Table 1: Accuracy of Recognition

For The Distance Based Classifier, The Euclidian Distance Between The New Image And All The Images In Image Database Are Calculated. Then This Distance Has To Be Within Two Threshold Value. The First Threshold Value Is Used To Screen This Image To Be A Valid Face Image Or Not. For Example, If The Image Is For A Flower Or House Then The Euclidian Distance Will Be Higher Between Flower And Facial Images In The Database.

The System Will Let User Know That This Is Not A Valid Face Image As The Distance Is Greater Than The Threshold. The Next Threshold Is Used To Determine Whether The Image Falls Under The Images In The

Database. The Second Threshold Is Chosen By Trial And Error Method. Table Ii Presents Different Threshold values and corresponding mismatch rates. If we increase the threshold values we could increase the probability of recognition as a face image. However, we should be careful about not to increase this value to much such that misclassification will increase due to non-facial image classification to a facial image.

Mismatch	69%	32%	12%	1%
Threshold	0.3	0.4	0.5	0.6

Table 2: Mismatch vs. Threshold

V. Conclusion

From the tests and analyses performed in this research, we conclude with the following remarks. Training set and test images need to be taken in good, comfortable illumination settings and need to be frontal faces with minimal head tilt. Number of images in the training set is a significant factor. It impacts on defining the correct threshold value for accepting true matches and rejecting false matches. The system performs relatively well with larger training sets and reflects similar behavior irrespective of the number of persons present in those larger training set. However, increase in the training set size increases the system performance and varies the acceptance or rejection rate depending on the person group size.

References

[1] http://www.biometricscatalog.org/NSTCSubcommittee/Documents/Face%20Recognition.pdf
[2] Matthew A. Turk and Alex P.Pentland. "Eigenfaces for recognisatoin". Journal of cognitive neurosciences, Volume 3, Number 1, Nov 27, 2002. http://www.face-rec.org/algorithms/PCA/jcn.pdf
[3] Rashedur M. Rahman, Anirban Das, M. Russel, Md. Shazzad Maruf. "Face Recognition for Single and Different Facial Expressions" http://computerresearch.org/stpr/index.php/gjcst/article/view
[4] Chi-Fa Chen, Yu-Shan Tseng and Chia-Yen Chen "Combination of PCA and Wavelet Transforms for Face Recognition on 2.5D Images", Palmerston North, November 2003, Department of Electrical Engineering, I-Shou University, Kaohsiung, Taiwan.http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=4700005.
[5] Howard L.Resnikoff, Raymond O.Wells Jr. "Wavelet Analysis" The scalable structure of Information.
[6] K.P. Soman and K.I. Ram Chandran "Introduction to Wavelet "from theory to practice.
[7] Martinez, A. Kak, "PCA versus LDA", IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 23, no. 2, pp. 228-233, 2001. http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=908974.
[8] Lindsay I Smith, "A tutorial on Principal Components Analysis" February 26, 2002 http://www.cs.otago.ac.nz/cosc453/student_tutorials/principal_components.pdf.
[9] http://www.hitechprojects.com/euprojects/hn2r/ex_documen1ts/ifa/leaflets/hn2r_itf _leaflet_tue_facerecog_v10_v13.pdf
[10] K. Jain, P. Flynn, A. Ross, "Handbook of Biometrics", Springer, 2007 http://biometrics.cse.msu.edu/info.html
[11] http://www.360biometrics.com/blog/difference-between-identification-authentication.
[12] Gonzalez, R. C. and Woods, R. E. 2003. "Digital Image Processing", 2nd Edition, Addison-Wesley Publishing Company
[13] http://www.merl.com/projects/facerecognition2/.
[14] MATLAB User guide, Retrieved on 28th. June, 2010. http://www.mathworks.com/access/helpdesk/help/techdoc/.
[15] Robi Polikar, "The Story of Wavelets" Dept. of Electrical & Computer Engineering Rowan University.
[16] http:www.face-rec.org/.
[17] http:www.mathwork.com.