FACE DETECTION AND RECOGNITION SYSTEM IN MATLAB

Savita[1], Mr.Ajay Dagar[2]
[1]M.Tech (ECE) HOD, Electronics & Communication Engineering, [2]WCTM, Farrukh Nagar, Gurgaon

Abstract: In this paper we describe our implementation algorithms for face detection and recognition in color images under Matlab. For face detection, we trained a feed forword neural network to perform skin segmentation, followed by the eyes detection, face alignment, lips detection and face delimination. The eyes were detected by analysing the chrominance and the angle between neighbouring pixals, and the results were used to perform alignment. The lips were detected based on the analysis of the Red Color component intensity in the lower face region. Finally the faces were determined using the eyes and lips positions. The algorithms were run on faces 1999 dataset. The proposed method achieved 96.9%, 89% and 94% correct detection rate of face, eyes and lips respectively. The correctness rate of the face recognition algorithm was 70.7%. Therefore, an ongoing struggle exists to purpose an effective, face recognition system with high precision and acceptable processing time.

Keywords: Images processing, skin segmentation, Eyes detection, Lips detection, Face recognition.

I. INTRODUCTION

System for face recognition is consisted of two parts: hardware and software. This system is used for automatic recognition users or confirmation of password. Government officials or firms and some private organization are using this system for face recognition especially for identification face by video cameras like input parameter or for biometrics system for checking identity using cameras and 3D scanners. Face recognition is an easy and straightforward task for humans, but it seems to be a challenge for computing systems due to the several environmental variables: face position and its distance from the cameras, lighting conditions and shadows on the face, and some individual aspects, such as beards and hairstyles. However, some individual face features hardly change regardless of environment, and the face recognition is performed by analyzing the patterns found in different images of the same individual. Facial recognition in humans is more complex and precise than in any automatic system, because it involves the context of the social environment and the acquire knowledge throughout their lives. In this paper, we present techniques for face detection and recognition under software Matlab. Face detection process is implemented by using a neural network and the analyzing the chrominance and standard deviation of the color of neighbouring pixels on each input face. As any other computing systems, face recognition systems are developed to speed up the execution of tasks that could be performed by humans, with the benefits of a higher performance of the machines when running repetitive tasks that require attention and patience.

II. LITERATURE SURVEY

Face detection has been a typical active research domain for decades because it can be applied in many fields. In recent years, face detection algorithm access a significant development, but in comparison with the humans visual systems is still a big gap because it has lots of variations of image appearance, such as pose variation (front, non-front), occlusion, image orientation illuminating condition and facial expression. A lot of research is going on in the area of human face detection at present. Many researches have proposed different approaches to address face detection problem such as:-

Paola Campadelli, Raffaller Lanzaratti proposed a face detection algorithm the key idea being to determine roughly the skin regions of a 2D color image and searching for eyes through them. The technic is based on a Support Vector Machine trained to separate sub images representing eyes from others. The algorithm can be used in face image database management systems both as a first step of a person identification, and to discriminate the images on the basis of the number of faces in them. Lamiaa Mostafa and Sherif Abdelazeem combined two algorithms for face detection to achieve better detection rates. The two algorithms are skin detection and neural networks. Wang Chuan-Xu and Li Xue proposed a face detection method employing BP network combined with Gabor Wavelet transform. Jizeng Wang and Hongmei Yang proposed a hierarchical face detection method by using the tamplate matching algorithm and 2DPCA algorithm. The methods includes two different classifiers. The first one is called rough classifier which filtrates the most of the non-face. The second one is a core classier, which uses 2DPCA algorithm to detect the face based on the result from the first classifier. Min-Quan Jing and Ling-Hwei proposed a method to detect a face with different poses under various environments. On the basis of skin color information, skin regions are first extracted from an input image. Yongqui Tu, Faling Vi and et al proposed a face detector has been designed using multi-classifier combination method.

Shamla Mantri et al (2011) proposed to label a Self-Organizing Map (SOM) to measure image similarity. To manage this goal, the author feed facial images associated to the regions of interest into the neural network. At the end of the learning step, each neural unit is tunes to a particular facial image prototype.

Mohammod Abdul Kashem et al (2011) investigated that face recognition has received substantial attention from researches in biometrics, patterns recognition field and computer vision communities.

Navneet Jindal et al (2013) give an idea of face detection from a long database of face images with different backgrounds is not an easy task.

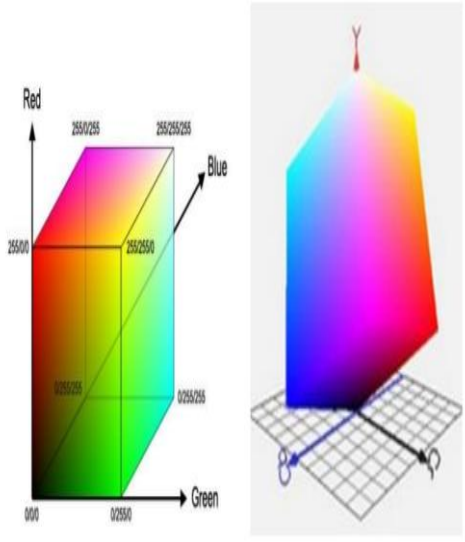
Cunjian Chan et al (2013) analyzed that the facial makeup has the ability to alter the appearance of a person. Such an alteration can degrade the accuracy of automated face recognition systems, as well as that of methods estimating age and beauty from faces.

III. COLOR SEGMENTATION

Several color spaces have been created to represent digital images. Including RGB and YCbCr, which were used in this work.

In RGB space, pixels are represented by the intensity of their Components Red, Green and Blue. Thus, pixal values in RGB can be illustrated in Fig. 1(a), in which each dimension measures its component intensity. White and Black are represented by the maximum and minimum values of three components, respectively.

The YCbCr color space is divided into the luminance and Chrominance components and it is useful in image compression applications and transmission of video signals. The luminance (Y component) measures the density of light intensity at a certain point in the image, and the chrominance (Cb and Cr Components) measure the color values. The YCbCr space is illustrated in Figure 1(b), in which the X and Y axes correspond to the chrominance, and the Z axes corresponds to the luminance.



(a) (b)

Fig.1(a) RGB Color Space;(b)YCbCr Color Space

The following equation converts RGB values into YCbCr :

$$\begin{bmatrix} Y \\ Cb \\ Cr \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.169 & -0.331 & 0.500 \\ 0.500 & -0.419 & 0.081 \end{bmatrix} \times \begin{bmatrix} R \\ G \\ B \end{bmatrix}$$

IV. IMAGE PROCESSING AND NEURAL NETWORKS IN MATLAB

Image files can be manipulated in Matlab through the Image Processing Toolbox, which includes the functions imread and imwrite, used to open and save image files, respectively.

Imread allows different syntaxes, and the most used in this work was A= imread (file name, fmt), in which A receives the color matrix of the image filename. Fmt. The color matrix size matches the image dimensions, and each position stores the color of a pixal. For binary images, the color matrix values are limited to 0and 1, which represent black and white, respectively. For color images, the matrix in RGB is tridimenstional, with integer values in the range 0.255.

The function in write also allows different syntaxes, including imwrite (A, filename. Fmt), in which the color matrix. A is stored in the image file filename. Fmt. The color matrices can be manipulated using basic matrix operations such as addition, subtraction and multiplication, or specific image processing functions, such as erosion and dilation. Neural networks can be created and managed in Matlab through the Neural Network Toolbox, which includes several training functions such as feedforwardnet and cascadeforwardnet, used to feed-forward and cascade-forward neural network, respectively. The network created in Matlab can be set on the number of hidden layers, training function, number of epochs, learning errors, training rate and output format.

V. FACE DETECTION

Face detection is a pre-requisite for face recognition performing, since at that stage, the faces will be delimited and the essential face features will be extracted.

The structure of the face recognition system consists of a wide range of tasks.

- Detention and tracking of images like faces.
- Align, normalize facial images
- Recognition.

A.NORMALIZATION OF IMAGE SIZE:-

Size normalization is an important pre-processing technique in face detection and recognition. Although various effective learning-based methods have been proposed. It is usually done to change the acquired image size to a default image size.

B.TEMPLATE MATCHING:-

The template-matching compares the face candidate image with the face template, measures the level of similarity and concludes whetherit is human face or a non-face. Several enhancements have been made to optimize the template-matching algorithm for the training images given by the EE368 instructors. A multi-layer classification scheme has been implemented to avoid missing faces or having non-faces. The color space chosen for the template matching is gray because the best results have been experimentally obtained. The template matching algorithm loads the face and non-face template images, it computes the 2 Dimensional (2-D) cross-correlation or the 2-Dconvolution.

The face template is an image made by averaging all faces

on the training images.

Fig.2. Face Template Image

A few human faces are not detected if only one face template is used. The reasons for the undetected faces is due to the very different color skin or face profiles found across several subjects. Additional face templates are used to detect the missing faces.

Fig.3.Additional face template image

There are two types of face detection problems:-

- Face detection in images
- Real-time face detection

Face Detection In Images:-

Most face detection systems attempt to extract a fraction of the whole face, thereby eliminating most of the background and other areas of an individuals head such as hair that are not necessary for the face recognition task. With static images, this is after done by running a across the image. The face detection system then judges if a face is present inside the window (Brunelli and Poggio, 1993).

Real-Time face Detection:-

Real time face detection involves detection of a face from a series of frames from a video-capturing device. While the hardware requirements for such a systems are for more stringent, from a computer vision stand point, real-time face detection is actually a for simpler process than detecting a face in a static image.

Since in real-time face detection, the system is presented with a series of frames in which to detect a face, by using spatio-temperal filtering, the area of the frame that has changed can be identified and the individual detected (Wang and Adelson, 1994 and Adelson and Bergen 1986).

Fig.4.Spatio temporally filtered image

VI. ALGORITHM

The following steps are performed for the face detection process:-

- Lighting Compensation
- Color Space Transformation
- Skin Color Detection
- Connected Component and Variance-based Segmentation
- Mouth Detection
- Face Boundary
- Nose Detection
- Lip Detection
- Verfifying Weighting Eyes-Mouth Triangles.

VII. PROPOSED SYSTEM

We propose a real time face recognition system based on PCA and Mahalanobis. The main task of face recognition is to extract effective features. The proposed system uses the Eigen face method to reduce image information. Even small faces have incredible information. This method should be able to break the image, generally to represent the face image effectively rather than the image. The proposed system uses basic analysis of components to extract classification of distances different from features such as Euclidian Distance, Manhattan Distance, and Mahalanobis Distance.

The different steps for calculating the eigen face are as follows:-

A two-dimensional face image can be represented as a one-dimensional vector device by connecting each row (or Column) to an elongated vector. We denote the vector N size (=Column X rows of the image) representing the image set in which the sample was taken, assuming Γ1, Γ2, Γ3....ΓM, subtract the average. To be calculated, it is subtracted from the original face (Γ i) and the results stored in variable.

$$\Psi = \frac{1}{M}\sum_{n=1}^{M}\Gamma_n$$

Calculate the heterogeneous matrix.

The heterogeneous Matrix A is calculated according to:-

$A = \phi^T \phi$

Calculation of eigen values and eigen values of the dispersion matrix. In this step we need to compute the eigen

values of the self-vectors (eigen vectors) Xi and λi.
[ϕ]Xi=*f*i
Xi is an eigen vector and *f*i is a unique surface. New images are converted to their eigen face components.
Mahalanobis Distance:-
The Mahalanobis space is defined as the area where sample changes are along each dimension. Consequently, vector transformation from image space to feature space is performed by dividing each factor of the vector by the corresponding standard deviation. This transformation provides a dimensionless space with changes in units of each dimension. If there are X and Y curves in the unconverted PCA space, the corresponding vectors m and n are in the Mahalanobis space.
First, λi=σ2 is defined
The relation between the vectors is defined

$$m_i = \frac{x_i}{\sigma_i} \quad n_i = \frac{y_i}{\sigma_i}$$

$$d(x,y) = \sqrt{\sum_{i=1}^{k}(m_i - n_i)}$$

Where λi is the i[th] Eigen value corresponding to the i[th] Eigen vector.
Manhattan Distance:-
Also known as the LI-norm or the Manhattan Distance or the City Block Distance. It is defined as follows:

$$d(x,y) = |x-y| = \sum_{i=1}^{k}|x_i - y_i|$$

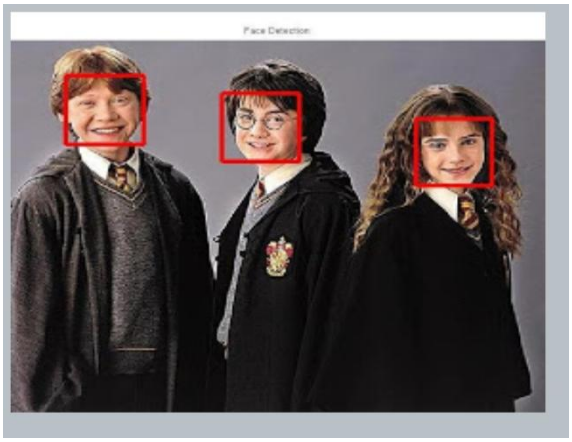
$$d(x,y) = ||x-y||^2 = \sum_{i=1}^{k}(x_i - y_i)^2$$

Face Detection Matalab Code:-

A. Face detection:

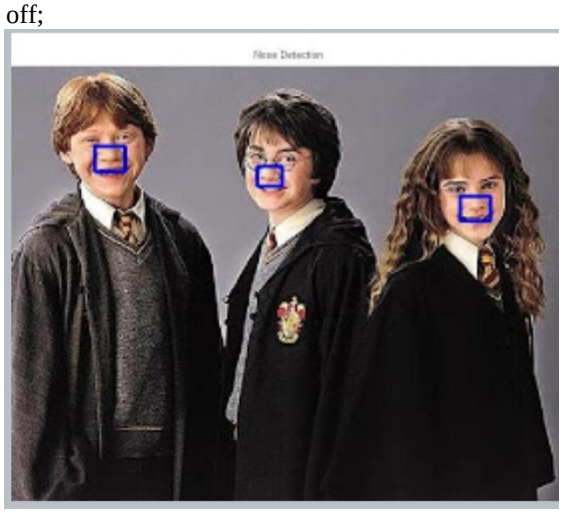
```
Clear all
clc
%Detect objects using Viola-Jones Algorithm
%To detect Face
F Detect  = vision . CascadeObjectDetector;
%Read the input image
I = imread ( 'Harry Potter.jpg' ) ;
%Returns Bounding Box values based on number of objects
BB = step (F Detect,I);
figure,
imshow (I); hold on
for i = 1: size (BB,1)
rectangle ('Position' , BB (i,:), 'LineWidth' , 5, 'Line Style', '-', 'EdgeColor', 'r');
end
title ('Face Detection');
hold off;
```

The step (Detector,1) returns Bounding Box Value that contains [x,y, Height, Width] of the objects of interest.



B. Nose Detection:-

```
%To detect Nose
Nose Detect =Vision.CascadeObjectDetector('Nose', 'MergeThreshold' ,16);
BB=Step(NoseDetect,I);
figure,
imshow(I); hold on
for i=1;size(BB,1)
rectangle('Position' ,BB(i,:), 'LineWidth' , 4, 'LineStyle', '-' , 'EdgeColor' , 'b');
end
title('Nose Detection');
hold off;
```

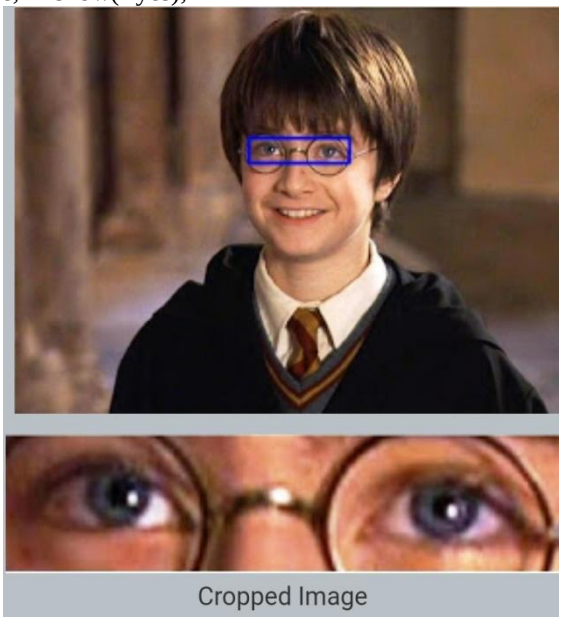


C.Mouth Detection:-

```
%To detect Mouth
Mouth Detect=vision.CascadeObjectDetector('Mouth' , 'MergeThreshold' , 16);
BB= step(MouthDetect, I);
figure,
imshow(I); hold on
for i=1: size(BB, 1)
rectangle('Position' ,BB(i,:), 'LineWidth' , 4, 'LineStyle' , '-' , 'EdgeColor' , 'r');
end
title('Mouth Detection');
hold off;
```


D. Eye Detection:-

```
% To detect Eyes
EyeDetect= vision.CascadeObjectDetector(Eye Pair Big');
% Read the input Image
I= imread('harry_potter.jpg');
BB= step(EyeDetect,I);
figure,
imshow(I);
rectangle('Position',BB,'LineWidth' , 4, 'Linestyle' , '-' , 'EdgeColor', 'b');
title('Eyes Detection');
Eyes= imcrop(I,BB);
figure, imshow(Eyes);
```



VIII. DATABASE

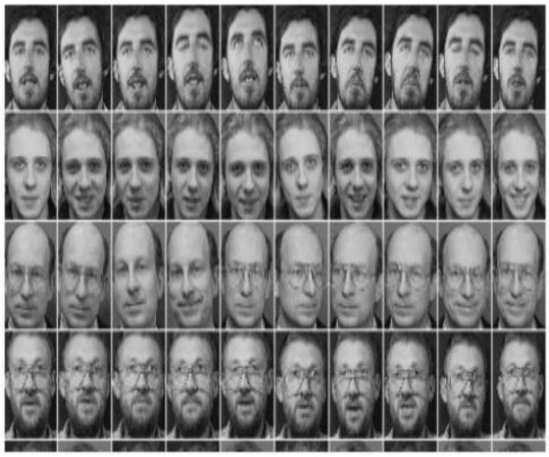
The face recognition code created for the database.Atotle of about 1200 face image were used for 78 people tested using various variations and configurations variables in the project.

The AT&T Database of Faces(ORL):-

"Face AT and T database" was formerly "face ORL database". The image consists of 40 different themes and 10 images per subject.

In some subjects, the images are taken at different times, the lighting is different depending on facial expressions (open \ closed eyes, not smile \ smile), face details (eye-glasses \ without glasses). All images were captured on a dark homogeneous background with vertical position subjects.

Each image is 92 (width) X 112 (height) pixals, 256 gray levels pixal.

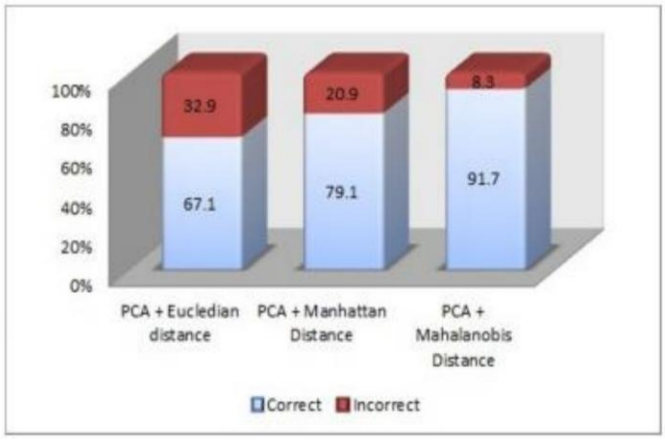


IX. RESULTS

AT and T Face Database

The results of running the codes on the AT and T Face Database are described below in Figure and Table.

Method Used	Correct	Incorrect	Recognition accuracy
PCA + Eucledian distance	286	34	89.4%
PCA + Manhattan Distance	284	36	88.8%
PCA + Mahalanobis Distance	301	19	94.1%



The system completed the query stream of 900 test images in 3 seconds, taking into account the 3ms query time slot.

X. CONCLUSION

This face recognition project gave me the opportunity to find out many of the common methods used for face recognition. In addition, we also learned that by combining two or more Teckins, the accuracy of the system can be greatly improved. In this project we developed a PCA-base face recognition system that extracts features and uses them to match different spatial matrices. The spaces used are Eucledian Distance, Manhattan Distance and Mahalanobis Distance.The results clearly show that recognition systems based on Mahalanobis distance exert better performance than

Conventional Eucledian- based workbooks. The code runtime is also fast and the response time is less than 0.2 seconds.

REFERENCES

[1] Bruner, I.S. and Tagiuri R. The perception of people. In Handbook of Social Psychology, Vol.2, G. Lindzey, Ed., Addison- Wesley, Reading, MA, 634-654. 1954.
[2] Bledsoe, W.W. The model method in facial recognition Tech. rep. PRI: 15, Panoramic research Inc., Palo Alto, CA. 1964
[3] Kelly, M.D. Visual identification of people by computer, Tech. rep. AI-130, Stanford AI project, Stanford, CA. 1970
[4] M. Turk and A. Pentland, "Eigen faces for recognition", J. Cognitive Neuroscience, Vol. 3, 71-86, 1991.
[5] Alaa Eleyan and Hasan Demir el, "Face Recognition System Based on PCA and Feed forward Neural Network". In Springer- Verlag Berlin Heidelberg 2005.
[6] Jawad Nagi, Syed Khaleel Ahmed and Farrukh Nagi, "A MATLAB based Face Recognition System using Image Processing and Neural Network" in International Colloquium on Singnal Processing and its Application in 2008.
[7] Shamla Mantri and Kalpana Bapat, "Neural Network Based Face Recognition Using Matlab" in IJCSET in Feb. 2011.
[8] Mohammod Abul Kashem, Md. Nasim Akhter, Shamim Ahmed, and Md. Mahbub Alam, "Face Recognition System Based on Principal Component Analysis (PCA) with Back on Propagation Neural Networks (BPNN)" International Journal Of Scientific and Engineering Research Volume 2, Issue 6 in 2011.
[9] Navneet Jindal and Vikas Kumar, "Enhanced Face Recognition Algorithm Using PCA with Artificial Neural Network" in International Journal of Advanced Research in Computer Science and Software Engineering in 2013.
[10] Cunjian Chan, Antitza Dant Cheva and Arun Ross, "Automatic Facial Makeup Detection with Application in Face Recognition" in IEEC in 2013.
[11] Guodong Guo Guowang Mu Yum Fu BBN Technologies. Thomas S. Huang UIUC "Human Age Estimation Using Bio-Inspired feature" 978-1-4244-3991-1/09/$25.00 ©2009 IEEE.
[12] Sujata G. Bhele and V. H. Mankar 2 "A Review Paper on Face Recognition Techniques" ISSN: 2278-1323 International Journal of Advanced Research in Computer Engineering and Technology (IJARCET) Volume 1, Issue 8, October 2012.
[13] Volker Blanz and Thomas Vetter, Member, IEEE "Face Recognition Based on Fitting a 3D Morphable Model" IEEE Transaction On Pattern Analysis And Machine Intelligence, Vol. 25, No.9, September 2003.

