International Journal of Engineering & Technology, 7 (4.10) (2018) 55-58

International Journal of Engineering & Technology

Website: www.sciencepubco.com/index.php/IJET

Research paper

Implementation of Facial Recognition for Home Security Systems

Arnab Pushilal[1], Sulakshana Chakraborty[2], Raunak Singhania[3], P. Mahalakshmi[4]*

[1,2,3, 4]School of Electrical Engineering, Vellore Institute of Technology, Vellore, India, 632 014
**Corresponding author E-mail: pmahalakshmi@vit.ac.in*

Abstract

In this paper, the design and development of a home security system has been detailed which uses facial recognition to conform the identity of the visitor and taking various security measures when an unauthorized personnel tries accessing the door. It demonstrates the implementation of one of the most popular algorithm for face recognition i.e. principal component analysis for the purpose of security door access. Since PCA converts the images into a lower dimension without losing on the important features, a huge set of training data can be taken. If the face is recognized as known then the door will open otherwise it will be categorized as unknown and the microcontroller (Arduino Uno) will command the buzzer to start ringing.

Keywords: *Arduino Uno; Covariance, Eigenvalue; Eigenvector; Eigen face; Eigen vector; Euclidean distance; Manhattan distance; PCA*

1. Introduction

With the advancement in the areas of personal identification in access control and technologies such as biometric identification seem to be gaining more popularity over the use of cards, pattern or password. Such system also requires the individual to touch the hardware for identification which is not the case in face recognition systems and hence has an added advantage making it a much quicker and efficient process [1]. In this system facial recognition has being implemented by the means of Principal component analysis. This approach is preferred because of its simplicity, speed and learning capacity. In PCA this the image are reduced to a lower dimensional representation consisting of only the salient features [2]. These salient features are called Eigen faces because these are the eigen vectors or the principal components of the training set images. These do not necessarily correspond to nose, eyes or other such part of the face. The images are categorized by finding out the weighted sum of eigen faces so for recognizing a face all we have to do is compare these weights with the faces of the known individuals [3].

In this paper we have proposed a system which comprises of dimensional reduction of the images, facial recognition, classification and a door access security system [4]. To access control to the door, an image will be captured and will be compared with the stored images using PCA after which the distance of the captured image from the stored images will be calculated to decide whether the person is authorized to gain access or not [5]. In this paper we have used two methods for calculating such distances namely Manhattan and Euclidian distance. Comparison of the two distances has been done which has been further discussed in the paper [6]. When the person has been classified as known the servo motor would rotate a certain angle signifying access to the door where as if the person is categorized as unknown, the system takes security measures such as raining the alarm with the blinking of the red led [7].

In this system, PCA and face classification has been implemented with the help of MATLAB installed on PC. An Arduino Uno board has been used which is interfaced with the PC via the USB port. The rest of the hardware connections are shown below.

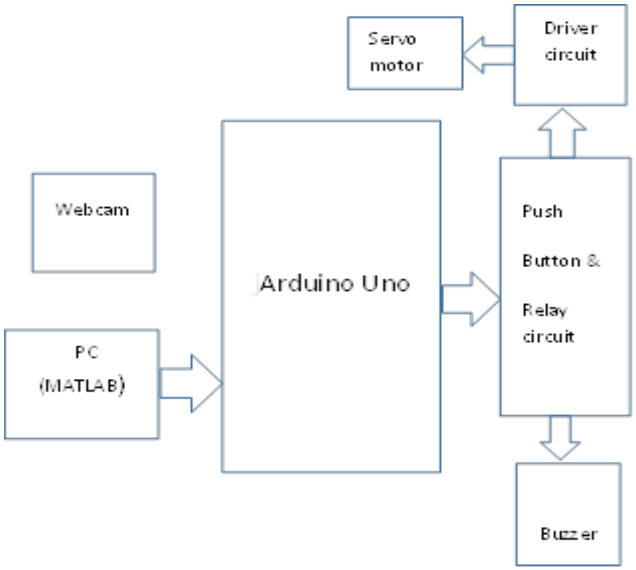


Fig.1. Block diagram of the facial recognition system for home security systems

2. Methodology

2.1. Principal Component Analysis

The Principal component analysis (PCA) is a statistical approach that performs orthogonal transformation on a set of possibly correlated variables and converts them into a smaller set of linearly uncorrelated variables called the principal components which is done by extracting the most important features of the data set and

Copyright © 2018 Authors. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

neglecting the redundancies of the data and hence in return reducing the size and compressing the data. PCA is one of the most efficient and reliable techniques and is popularly used for face recognition. Here Eigen faces would be used which are the principal components of the distributed data set of faces or equivalently the Eigen vectors of the covariance matrix of the training set images. Where a N by N pixel image is converted into a N^2 row vector and hence considered to be a point in the dimensional space [8].

Algorithm for implementing PCA is as follows
1) the image matrix of (N x N) pixels is converted into a column vector Γ of size (S x1) where S = (N x N). Hence
The **Training Set** is: Γ = [Γ1Γ2 … Γ]
2) The average of the face images is calculated by
$\Psi = 1/M \sum_{i=1}^{M} \Gamma i$
3) Each image is then normalized by calculating
$\Phi i = \Gamma i - \Psi$
Hence the difference Matrix equals to:
A = [1Φ2 … ΦK]
4) Next the covariance matrix is found by :
C = $A\,A^T$, where C = (S×S).
5) For reducing the dimensionality of the covariance matrix with such huge dimensions we perform the following step:
L = , where L is of the size (K×K).
6) To calculate the Eigen vectors of the covariance matrix:
$A^T A\,Xi = \lambda i\,Xi_k$ - Y_{ik}
Now, multiplying both sides with A in the above equation,
$AA^T A\,Xi = A\,\lambda i\,Xi\,AA^T\,(A\,Xi) = \lambda i(A\,Xi)$
Taking $Xi = Ui$
Where Ui is the Eigenvector with eigenvalues λi
7) For projecting an image in the face space we use the equation given below
$\Omega k = U_k\,T\Phi i$
So for projecting the test image we first subtract the image vector with the mean which is shown below:
$\Phi t = \Gamma t - \Psi$
Then for projection in the face space is achieved by [9].
$\Omega = U_k \Phi_t$

To find out the optimal number of Eigen vectors to be taken for further computations, we observe the graph of Eigen vectors Vs variance. We see that the variance tends to increase proportionally to a certain extent and then there is a decrease in the amount of variance with any additional Eigen vectors. From Figure 2, we find out that 30 Eigen vectors is the optimal amount.

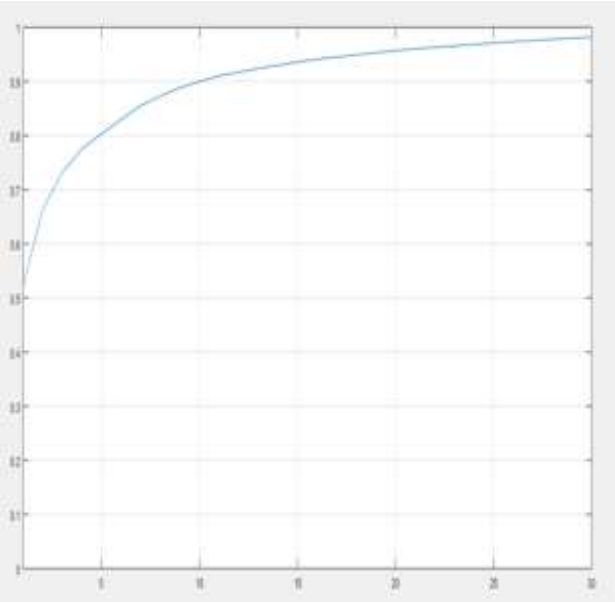


Fig.2 Eigen Vectors Vs variance

2.2. Face Recognition

After the computation of Eigen faces, next come the recognition of faces which is a broad term which is inclusive of the following steps:

- Identification of the images, where the labels of each images must be obtained
- Recognition of the test image, where it would be decided whether the image matches the ones from the data base.
- Classification of the image where the image is assigned to a certain class [10].

The test image are classified into known and unknown which is done by calculating the distance of the test images from the Eigen faces projected in the face space. We have used Euclidean method and Manhattan method to find out this distance and later have done a comparative analysis between these two methods.
Euclidian distance is defined as:

$$d(\mathbf{q},\mathbf{p}) = \sqrt{(q_1 - p_1)^2 + (q_2 - p_2)^2 + \cdots + (q_n - p_n)^2}$$

Manhattan distance is defined as:

$$d(i,j) = |x_{i1} - x_{j1}| + |x_{i2} - x_{j2}| + \ldots\ldots + |x_{in} - x_{jn}|$$

Manhattan and Euclidean are popularly used distance metrics to determine the similarities between the data items of a cluster[11]. A threshold value Θ is chosen to categorize the images as known and unknown. The threshold value being the maximum acceptable distance more than the witch the image will be classified as unknown compared with minimum.

If $\varepsilon_k(\min) \geq \Theta$, then the test image is categorized as unknown .

If $\varepsilon_k(\min) < \Theta$, then the test image is categorized as known.

2.3. Hardware Design

The hardware configuration of this system shown in Figure 3, is composed of Arduino Uno, 2 port relay, push buttons, Servo motor and buzzer. USB cable is used to interface PC to Arduino. The results from the MATLAB program is used as the input to the Arduino Uno board and the rest of the components are used as the outputs.

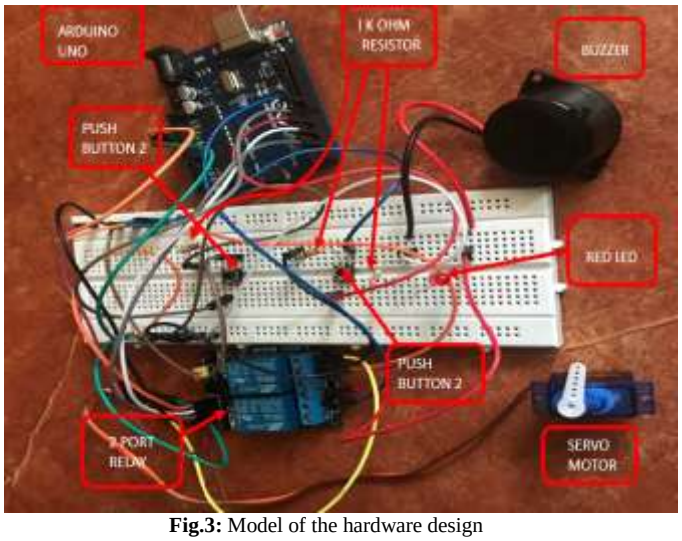


Fig.3: Model of the hardware design

The hardware is modeled according the figure above. The specifications are as follows:

2.4. ARDUINO UNO

Arduino is one of the most popular microcontrollers and is an open source hardware/ programming platform and is based around Atmel microcontroller open source of the browser electronics of hardware and software components. The objective of the design platform provided by Arduino is to suit the needs of artists and designers of the amateurs, any person interested in creating an interactive environment. The Arduino can receive inputs from the sensors and it can therefore be the use of indicators and the hinge assemblies, sensors and other components [12].
In this project Arduino Uno (Figure 4) is being used, the system configuration is : ATmega is used on the basis of 328 it has 14-pin Digital Input, 16 Mhz crystal oscillator, USB port, power socket, ICSP head and a reset button. To connect it with the PC one simply needs to connect it with the USB cable [13].

2.5. SERVO

A 3 pole motor plastic 9g servo was chosen. This would serve as a model for the door that needs to be unlocked. Whenever the program successfully executes the servo rotates a fixed degree.
A 2 channel level with TTL control is used rated at 7A/240V.

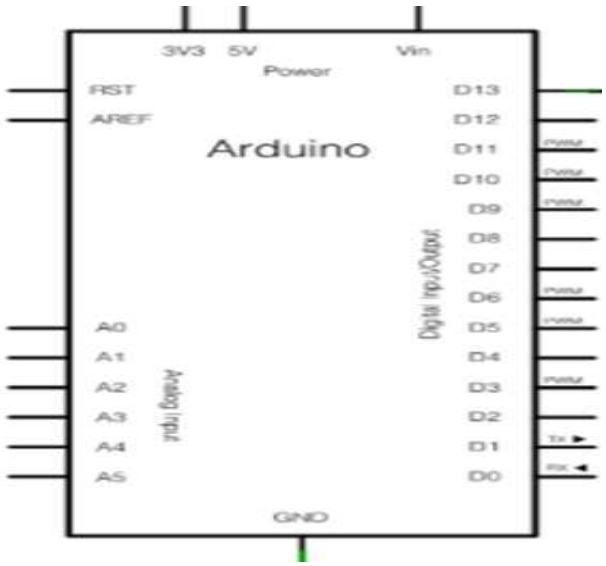


Fig. 4: Pin diagram of Arduino Uno

3. Implementation

50 ordinary test images of two known faces were taken and labeled accordingly in a file. MATLAB was started on an infinite loop and breaks off when a push button is pressed similar to pressing your home doorbell. Then it asks for registration which would be password protected. This involves labeling a person once their picture is taken. Then a test image is taken. The images are taken either using a webcam or a phone which is connected to the system with Wi-Fi. This represents an IP camera connected outside your door. The train images go through PCA analysis and principle components are taken out. Now a face space is formed with the reduced amount of data.
The distance from the testing data is then calculated with Manhattan and Euclidian distances. With trial and error thresholds for recognizing the test images are found. The Arduino board is connected to the program using a USB com port or a Bluetooth module. 2 pushbuttons are configured through 2 digital I/P pins and grounded through a resistor. One of these pushbuttons will act as the doorbell and the other will be the prompt for clicking the testing image. A relay is connected to the buzzer and a servo motor, simulating the alert if the image is unrecognized and success by running the motor. The relay, buzzer and the servomotor are all connected to the Arduino via digital pins. A success triggers the relay channel connected to the servo, thereby unlocking the door. A failure in turn activates the relay channel connected to the buzzer.
The PCA process in MATLAB could be achieved in the following steps:

- Store in Notepad the image names with the labels, indicating who is who.
- Input File into a cell array of 1*2 cell each containing 50*1 entries of the Image Index and the label
- Concatenate image path in another cell array, create a structure, to find the number of training images.
- Find the number of training images.
- Then each Image is stored in a column vector
- Mean of each ‘characteristic’ or pixel is taken across the row, and subtracted with all the images to normalize the data. Normalization of images leads to removal of the average brightness of the data points, i.e. the illumination. As we are not interested in the illumination condition of the images in this case but on the data that they represent hence we remove the average pixel value per data point. [13]
- Take principal components in row vectors
- C will be the eigenvectors of the covariance matrix (AAT), S will be the ‘score’ of size (50*49) (from 300*200 dimensions, it has been reduced to 49). L will be eigenvalues. S columns would be orthogonal to each other and forms the components for ‘separating the data’.
- Take different Range of eigenvectors, for testing
- Normalize the testing data
- Project into the ‘face space’ or the reduced dimensioned space by multiplying it with the eigenvectors
- Calculate the distance from every image.
- Test out both Euclidian and Manhattan Distance, to check accuracy
- Find the minimum distance and its position, across the row to determine which image it is closest to.
- Display the output in another file

Apart from this the registration process would add to the train Images and Labels will be added. And the testing Image would be added to the testing folder. In Figure 5, ‘a’ &‘s’ refers to the identity of the authors. Figure 6 shows some of the Eigen faces.

4. Results and Observations

The Eigen faces were found to be as follows. We can see from the set of images below that as we go further down, there is an increase in the variance or noise in the images, hence rather than considering all the Eigen faces an optimal number of them is chosen. These faces represent a weighted average of the training set data. Each image in the testing data set can be retrieved by multiplying the coefficients to the Eigen faces. The training data can also be projected into the face space by similar multiplication.

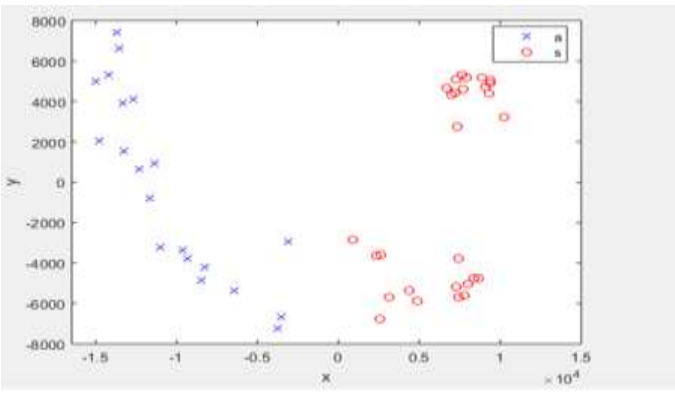


Fig. 5: Scatterplot for 2 principal components

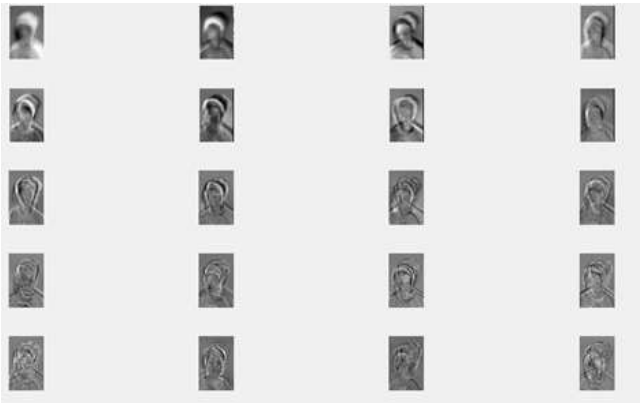


Fig. 6: Some of the Eigen faces displayed

After several trial and errors, the threshold for Euclidian distance to recognize an unknown person was determined to be at 19000. For Manhattan Distance this was determined to be at 40000.

In our tests, Euclidian Distance was determined to be more accurate than Manhattan Distance. Euclidian distance had a 90% accuracy in determining the unknown person, with the same background as the training data, whereas Manhattan had an accuracy of 80%.

Apart from this, tests were done to see how many principle components were actually required to separate the data. As shown in figure 1, 30 principle components contained 90% of the variance. To reduce the computational time and to maintain accuracy how many principle components to be is important.

As seen in the images mentioned below, the scatter plot in MATLAB shows us that two principal components are enough to separate the 'a' and the 's'. The x axis represents the first principle component and the variance along that direction is the maximum, whereas the 2nd principle component along the y axis contains lesser variance. The 'x' represents the pictures of a & the 'o's represents the pictures of s. As we can clearly see there is a partition amongst the two along the two principle components.

Some of the matched Images are as follows:



Euclidian Distance: 9425



Manhattan Distance: 35238



Euclidian Distance 10020



Manhattan Distance 32293

After trial and error, the number of principal components was chosen to be 10, which was an ideal balance between computational time and accuracy. If we take a three dimensional scatter plot, we'll get more separation and so forth, but the variance along each dimension reduces. PCA has a flattening effect on the data set, which basically means it weeds out all the unnecessary information, so a dataset of 50 images of 300*200 was reduced down to 49, and then only 30 were chosen for testing.

5. Conclusion

In this paper we have successfully implemented a face recognition system using Principal Component Analysis and Eigen face approach. Successfully recognition of the images and categorizing the faces into known and unknown was done which was quite accurate. Hence the Eigen face method provides a practical solution to the problem of facial recognition and is well fitted as it gives fast results with relatively simple implementation. Therefore an efficient doorbell access facial recognition system was created and it was seen that it is more efficient with the ideal threshold using the Euclidian distance and the ideal number of principle components. In terms of future scope, the code could be developed to recognize faces in real time and create a boundary box around them, which would increase efficiency even more. Also alerts can be added to the code in case of an emergency to notify the home owner.

References

[1] Automatic Door Access System Using Face Recognition Hteik Htar Lwin, Aung Soe Khaing, Hla Myo Tun

[2] . Shemi P M, Ali M A, A Principal Component Analysis Method for Recognition of Human Faces: Eigenfaces Approach, International Journal of Electronics Communication and Computer Technology (IJECCT),Volume 2 Issue 3 (May 2012)

[3] M. Turk, A. Pentland: Face Recognition using Eigenfaces, Conference on Computer Vision and Pattern Recognition, 3 – 6 June 1991, Maui, HI , USA, pp. 586 – 591

[4] Ayushi Gupta, Ekta Sharma, NehaSachan and Neha Tiwari. Door Lock System through Face Recognition Using MATLAB. International Journal of Scientific Research in Computer Science and Engineering, Vol1, Issue-3, 30 June 2013.

[5] Survey Paper on Door Level Security using Face Recognition Harshada B. More1 , Anjali R. Bodkhe2 Department of Comp Science & Engg, Government College of Engg, Jalgaon Maharashtra, India1.

[6] Study of Euclidean and Manhattan Distance Metrics using Simple K-Means Clustering Deepak Sinwar #1 , Rahul Kaushik *2 #Assistant Professor, *M.Tech Scholar Department of Computer Science & Engineering BRCM College of Engineering & Technology, Bahal.

[7] MohammaAmanullah "MICROCONTROLLER BASED REPROGRAMMABLE DIGITAL DOOR LOCK SECURITY SYSTEM BY USING KEYPAD & GSM/CDMA TECHNOLOGY", IOSR Journal of Electrical and Electronics Engineering (IOSR - JEEE), Volume 4, Issue 6 (Mar. - Apr. 2013).

[8] Principal component analysis Herve Abdi and Lynne J. Williams2J. H. Davis and J. R. Cogdell

[9]] K. I. Diamantaras and S. Y. Kung, "Principal Component Neural Networks: Theory and Applications", John Wiley & Sons,Inc., 1996

[10] Face Recognition using Principle Component Analysis Kyungnam Kim Department of Computer Science University of Maryland, College Park MD 20742, USA

[11] Vadivel, A.K.Majumdar & S. Sural, " Performance comparison of distance metrics in content based Image retrieval applications", Intl. Conference on Information Technology *Motorola Semiconductor Data Manual*, Motorola Semiconductor Products Inc., Phoenix, AZ, USA, 1989.

[12] P. Jamieson,"Arduino for Teaching Embedded Systems. Are Computer Scientists and Engineering Educators Missing the Boat" International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS'11), 2011

[13] M. A. Zermani and E. Feki and A. Mami ," Temperature Acquisition and Control System based on the Arduino",International Journal of Emerging Science and Engineering , Vol.2 No.12, pp:1-6,2014.