2016 IEEE International WIE Conference on Electrical and Computer Engineering (WIECON-ECE)
19-21 December 2016, AISSMS, Pune, India

Face Detection and Recognition using Raspberry Pi

Ishita Gupta[1], Varsha Patil[2], Chaitali Kadam[3], Shreya Dumbre[4]
igupta95@gmail.com[1]
AISSMS IOIT, Kennedy Road, Pune, India[2] Savitribai Phule Pune University, India[1, 3, 4]

***Abstract*— in today's world, face recognition is an important part for the purpose of security and surveillance. Hence there is a need for an efficient and cost effective system. Our goal is to explore the feasibility of implementing Raspberry Pi based face recognition system using conventional face detection and recognition techniques such as Haar detection and PCA. This paper aims at taking face recognition to a level in which the system can replace the use of passwords and RF I-Cards for access to high security systems and buildings. With the use of the Raspberry Pi kit, we aim at making the system cost effective and easy to use, with high performance.**

Keywords- Face recognition, face detection, Haar Detection, PCA, Raspberry Pi.

1. INTRODUCTION

The information age is quickly revolutionizing the way transactions are completed. There is a need for a faster and accurate user identification and authentication method. Face recognition has become one of the most important user identification methods. Literature survey statistics shows that research work in face recognition system is in its booming era, and in the past forty years, the research in this field has increased exponentially [1].

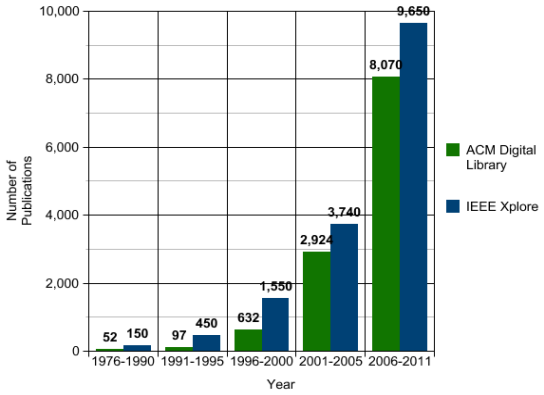


Fig1. Research Statistics of Face Recognition

Face recognition technology emulates the capabilities of human eyes to detect faces. This is done by smart computing that creates "face bunch" that consists of 70 nodal points. Features are extracted from the face and saved as templates. These templates are compared to the face detected. For this research, we interfaced an LCD, Camera and a Motor to the Raspberry Pi board. We have made a real time application, which compares the scans to records stored in the Raspberry Pi which in turn is used as a gate pass, wherein the name of the detected person is displayed on LCD and the motor will rotate indicating opening and closing of the gate.

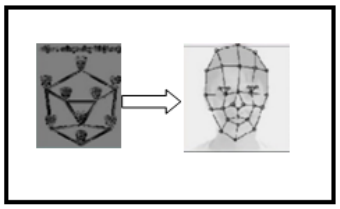


Fig 2. Face Bunch Graph

2. FACE DETECTION USING HAAR CASCADES

Object Detection using Haar feature-based cascade classifiers is an effective method which was proposed by Paul Viola and Michael Jones [2, 3]. It is an adaptive machine learning based approach in which a cascade function is trained from several positive and negative images. This is then used to detect objects in other images. Initially, this algorithm requires plenty of positive images (images of faces) and negative images (images without faces) to train the classifier. Then Features are extracted from it.

Every feature is a single value obtained by subtracting sum of pixels under white rectangle from sum of pixels under black rectangle. Then all possible sizes and locations of every kernel are used to calculate plenty of features. To calculate each feature, the sum of pixels under white and black rectangles is found [2, 3].

3. FACE RECOGNITION USING EIGEN FACE APPROACH

The face recognition algorithms used here is Principal Component Analysis (PCA). It involves a mathematical procedure that transforms a number of possibly correlated variables into a number of uncorrelated variables called principal components, related to the original variables by an orthogonal transformation [3]. The Eigen face approach helps reducing the size of the database required for recognition of a test image. The Eigen values calculated from the Eigen Vector covariance matrix are rejected or stored depending upon the threshold thus creating a face space [3, 4, and 5]. Calculating the weights and the Euclidean distance a comparison is held and match is found [5]. This conventional Eigen face Approach is incorporated in the ARM Cortex of Raspberry Pi for face recognition using face recognition modules in python code.

4. METHODOLOGY

The Functionality of this system is mainly categorized in following steps.

a. To enrol and detect faces using camera connected to the ARM Cortex of Raspberry Pi board.

b. To display the match status on the LCD as well as the terminal running on the VGA (Video graphics array) monitor.

c. To program for the same using python language. The code imports certain modules that enable functions such as face recognition, GPIO modules.

d. To drive the motor in clockwise and anticlockwise direction upon getting a present status.

The identification and authentication technology operate using the following four stages: [6]

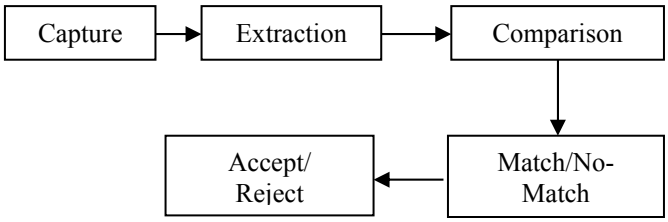


Fig3. Flow of operation

a. Capture: A physical or behavioural sample is captured by the system during enrolment and also in identification or verification process

b. Extraction: Unique data is extracted from the sample and a template is created.

c. Comparison: The template is then compared with an existing sample.

d. Match/non match: The system decides if the features extracted from the new samples are a match or a non match and accordingly accept/reject.

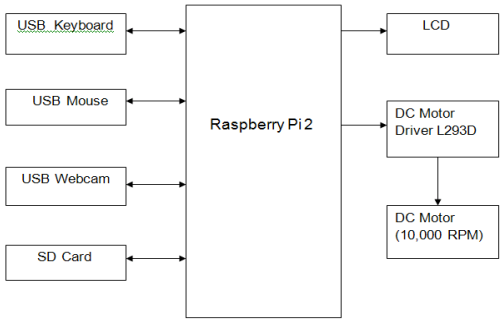


Fig4. Simplified Block Diagram

The input/output modules are interfaced to the Raspberry Pi 2. The input part, the keyboard and mouse are used to pass commands to the Raspberry pi and Webcam to capture image which are stored in Secure Digital Card and is also used to compare an image from those stored in database. On the other hand the output part consist of LCD, DC motor followed by its driver L293D.The motor used here has a range up to 10,000 RPM.

The necessary steps to initialize the system include creating a database and then using that database as a reference for comparing the live test points. As we boot the raspberry pi following things needs to be taken into consideration like connecting the LCD monitor in order to verify the image captured by the webcam which will be used for detecting the entity.

Linux based commands are provided as an input to the Raspberry Pi2 in order to save images in the external SD card. Two separate directories are maintained for the image which are captured as a reference i.e. Positive and Negative, Positive directory holds the reference image which has been captured with no error. While those with minimal to maximum defective images are stored in the negative directory.

5. SYSTEM DESIGN

Hardware Design*:*

It includes Raspberry Pi2development kit, connecting cables, LCD, DC Motor driver IC, DC motor, Power Supply, USB keyboard, USB mouse and USB webcam.

1. Raspberry Pi 2

Raspberry Pi is a small scale based computer, it performs efficiently when it comes to running games, image files, and documents. They can be manufactured using different configurations as per the user's needs.

It runs at 700MHZ with nearly 256MB RAM, with on board graphic card capabilities. RaspberryPi2 are considered one of the few low power consumption CPU's [7].

The basic layout of the Raspberry pi2 consist of

1. Input/Output.
2. RAM
3. CPU/GPU
4. USB hub
5. Ethernet
6. HDMI Port

2. LCD

The LCD used here is 16 x 2 alphanumeric LCD. It is a very basic module which overcomes the disadvantage of the seven segment display as well as multi-segment LCD. It consists of command register to perform the user defined instructions and data register to display the data on LCD.

3. DC Motor Driver

The L293D Motor drivers provide voltages ranging from 5V TO 36 V with output current of 1 A per channel. L293D provides bi directional currents and can be operated at temperature up to 70 degree Celsius.

4. DC Motor

The speed of the motor can be changed by varying the voltage levels. Here the motor rotates in clockwise and anticlockwise direction at a speed of 10,000RPM, representing opening and closing of gate.

Software design:

To code for the Hardware setup, python language is used [8]. To access the editor Linux terminal is required. The face detection and recognition part is carried out by certain Linux commands which are as follows:

- **sudo su** # To switch to root user.
- **python capture-positive.py** # To create database of images.
- **ctrl+c** # To give keyboard interrupt and stop the image capturing after desired number of images are taken for a face.
- **python train.py** # To train the captured images and extract features from it, by separating the face (positive images) from the background or non-face image (negative images) and thus creating a .xml file.
- **python project2.py** # To detect and recognise which face is present in front of the web cam. It calculates the confidence value and matches the flag status. If a face corresponds to certain flag 10 times then it recognises the respective face and indicates a present status on the terminal as well as the LCD and the motor rotates in the clockwise direction for 15 seconds and anticlockwise direction for 5 seconds, indicating opening and closing of gate

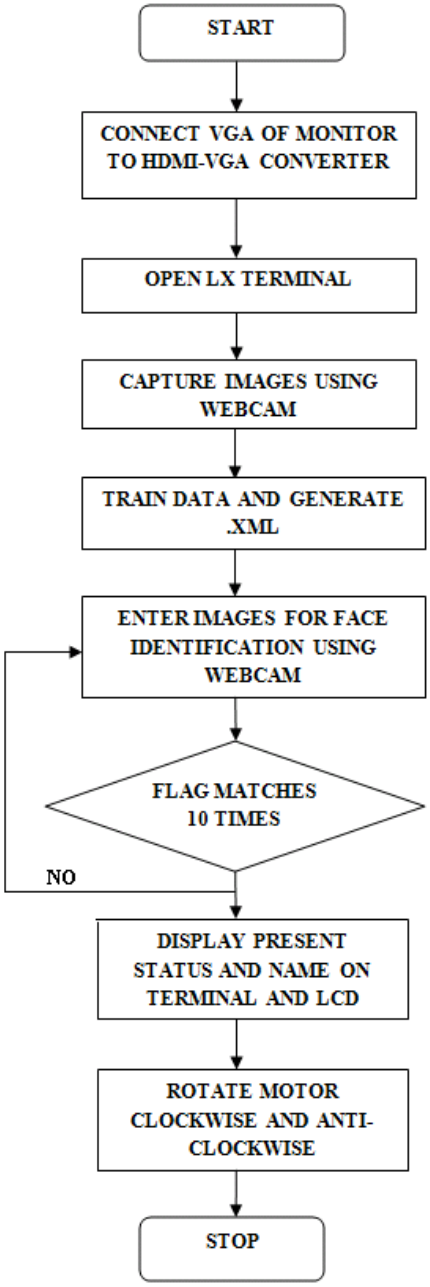


Fig5. Flow Chart of Software functioning

6. OUTPUT RESULT

To recognise correct face it must match at least 10 times with an existing flag.

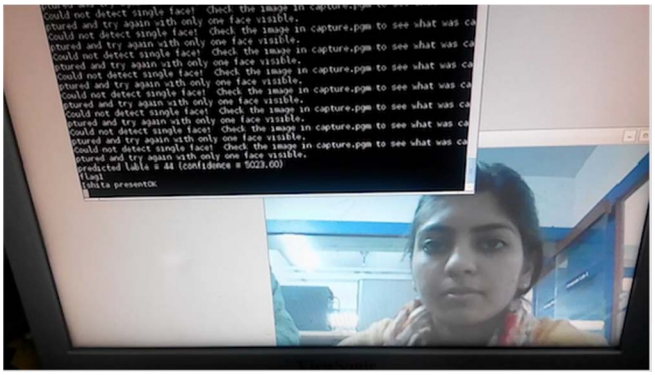
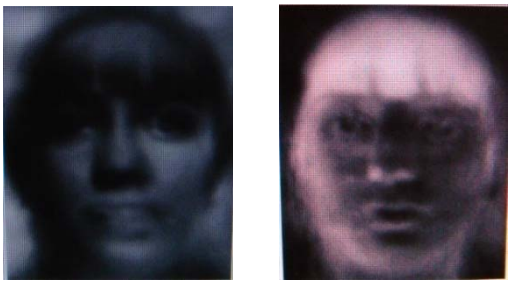
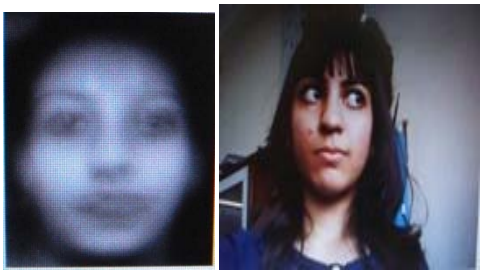


Fig6. Identification of Face

When face images are captured and trained several positive and negative images are created. Capture.pgm, positive.png, negative.png and mean.png files are created in the first instance after obtaining the confidence value.



a. Negative Image b. Positive Image



a. Mean Image d. Capture.pgm

Fig7. Various images created

If the face is recognised correctly then the LCD glows and reads the name of the person detected and a “Present” status. Similarly at the terminal on the Personal computer the name and a “presentOk” status are obtained.

Fig8. Output on LCD

7. ACKNOWLEDGEMENT

We would like to express our sincere gratitude to the staff of Department of Electronics and Telecommunication, AISSMS IOIT, Pune for their encouragement and support. We would also like to thank the faculty of ARDE (DRDO), Pune who helped us in this research work.

8. REFERENCES

[1] W. Zhao, R. Chellappa, P. J. Phillips, A. Rosenfeld, "Face Recognition: A Literature Survey", *ACM ComputingSurveys, Vol. 35, No. 4, December 2003, pp. 399–458.*

[2] R. Lienhart and J. Maydt, "An extended set of Haar-like features for rapid object detection," *Proceedings. International Conference on Image Processing*, 2002, pp. I-900-I-903 vol.1.

[3] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, 2001, pp. I-511-I-518 vol.

[4] A. M. Patil, Dr. Satish R. Kolhe, Dr. Pradeep M. Patil (2009), *“*Face Recognition by PCA Technique”*, Second International Conference on Emerging Trends in Engineering and Technology, ICETET-09*

[5] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces" *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Maui, HI, 1991, pp. 586-591

[6] U. Gupta, P. Sinha and P. Chharia, "Reduction of genuine and imposter score overlapping based on intra-class variations and deviation of scores in a multimodal biometric system," *2015 International Conference on Signal Processing and Communication Engineering Systems*, Guntur, 2015, pp. 126-131.

[7] Ningthoujam Sunita Devi and K Hemachandran, “Automatic Face Recognition System using Pattern Recognition Techniques: A Survey” *Proceedings International Journal of Computer Applications, IJCA-13*

[8] Pritish Sachdeva and Shrutik Katchii, “A Review paper on Raspberry Pi” *Proceedings International Journal of Current Engineering and Technology, IJCET-14*

[9] Y. C. Huei, "Benefits and introduction to python programming for freshmore students using inexpensive robots," *2014 IEEE International Conference on Teaching, Assessment and Learning for Engineering (TALE)*, Wellington, 2014, pp. 12-17