A project report on

Face Recognition Door Lock System Using Raspberry Pi

Submitted in partial fulfilment of the requirements for award of the degree of

BACHELOR OF TECHNOLOGY

In

ELECTRONICS & COMMUNICATION ENGINEERING

By

Karanam Akhil	Nandana Sai Kumar	Botta Bhanu Lokesh
(20U41A0409)	(20U41A0450)	(20U41A0425)

Under the Esteemed guidance of

Mrs. G. Siva Kumari M. Tech

Assistant Professor, Department of ECE

DEPARTMENT OF ELECTRONICS & COMMUNICATION ENGINEERING

DADI INSTITUTE OF ENGINEERING & TECHNOLOGY

(AN AUTONOMOUS INSTITUTE)

(Approved by A.I.C.T.E., New Delhi & Permanently Affiliated to JNTU GV)

Accredited by NAAC with 'A' Grade and Inclusion u/s 2(f) & 12(B) of UGC Act

An ISO 9001:2015, ISO 14001:2015 & ISO 45001:2018 Certified Institute.

NH-16, Anakapalle – 531002, Visakhapatnam, A.P.

(2020-2024)

DEPARTMENT OF

ELECTRONICS & COMMUNICATION ENGINEERING

CERTIFICATE

This is to certify that the project report entitled **"Face Recognition Door Lock System Using Raspberry Pi"** submitted by Karanam Akhil (20U41A0409), Nandana Sai Kumar Goud (20U41A0450), Botta Bhanu Lokesh (20U41A0425). In partial fulfilment of the requirements for award of the Degree of **Bachelor of Technology** in **Electronics & Communication Engineering**, from Dadi Institute of Engineering & Technology(A), Anakapalle affiliated to JNTUGV, accredited by NAAC with 'A' grade is a record of bonafide work carried out by them under my guidance and supervision.

PROJECT GUIDE **HEAD OF THE DEPARTMENT**

EXTERNAL EXAMINER

ACKNOWLEDGEMENT

We would like to express our gratitude to our project guide, **Mrs. G. Siva Kumari**, **Assistant Professor** of the Electronics & Communication Engineering, for her guidance and assistance throughout the development of this project. Without her supervision, support, and encouragement, we would not have gained awareness of many new things during our project.

We convey our heartfelt thanks to **Dr. P. Poorna Priya**, Professor & Head of the Department of Electronics & Communication Engineering, for motivating us to successfully complete the project.

We would like to thank Principal **Dr. R. Vaikunta Rao**, Dadi Institute of Engineering & Technology, for providing the necessary facilities to carry out our project work successfully.

We would like express our deep sense of gratitude to **honorable Chairman, Sri Dadi Ratnakar**, of Dadi Institute of Engineering & Technology, for providing necessary facilities to carry out our project work successfully.

We express our gratitude to the Teaching and Non-teaching Staff of the Department of Electronics & Communication Engineering, who have been directly and indirectly involved in this journey, for their encouragement in completing our project.

Endeavors over a long period can also be successful through constant effort and encouragement. We wish to take this opportunity to express our deep gratitude to all the people who have extended their cooperation in various ways during our project. It is our pleasure to acknowledge the help of all those respected individuals.

PROJECT ASSOCIATES

K.Akhil (20U41a0409)
N.SaiKumar (20U41A0450)
B.Bhanu (20U41A0425)

DECLARATION

We hereby declare that the project entitled **"Face Recognition Door Lock System Using Raspberry Pi"** is submitted in partial fulfilment of the requirements for the award of Bachelor of Technology in **Electronics & Communication Engineering** under esteemed supervision of **Mrs. G. Siva Kumari, Assistant Professor**. This is a record of work carried out by us and results embodied in this project report have not been submitted to any other university for the award of any Degree.

PROJECT ASSOCIATES

K.Akhil (20U41A0409)
N.SaiKumar (20U41A0450)
B.Bhanu (20U41A0425)

ABSTRACT

Today we are facing security issues in every aspect. So we have to resolve these issues by using updated technology. In this project, we are using the Face recognition module to capture human images and to compare with stored database images. If it matches with the authorized user then the system will unlock the door by an electromagnetic lock. The need for facial recognition system that is fast and accurate that continuously increasing which can detect intruders and restricts all unauthorized users from highly secured areas and aids in minimizing human error. Face recognition is one of the most Secured System than biometric pattern recognition technique which is used in a large spectrum of applications. The time and accuracy factor is considered about the major problem which specifies the performance of automatic face recognition in real-time environments. Various solutions have been proposed using multicore systems. By considering the present challenge, this provides the complete architectural design and proposes an analysis for a real-time face recognition system with Haar cascade classifier. In this algorithm, it converts the image from color to greyscale image and divides into pixels and it will be allocated in a matrix form and those images will be stored in the database. If an image is detected then microcontroller will send power to the motor driver unit then the electromagnetic lock will unlock the door and it will lock again when there is no power supply to that unit. If unknown person detected information is passed to authorized persons mail, so that he can unlock with android app if he wants to allow unknown person. Finally, this project concludes for the advanced implementations achieved by integrating embedded system models against the convention.

Keywords: FaceRecognition,Database,Electromagneticlock,Haarcascade,Microcontroller,Mail,App.

TABLE OF CONTENTS

LIST OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

Symbol	Definition	Unit
NC	Normally Closed	V
NO	Normally Open	V
C	Common Terminal	V

ABBREVIATIONS

CPU	Central Processing Unit
GPU	Graphics Processing Unit
DSP	Digital Signal Processing
SDRAM	Synchronous Dynamic Random Access Memory
GUI	Graphical User Interface
TCP/IP	Transmission Control Protocol/Internet Protocol
BLE	Bluetooth Low Energy
OpenCV	Open Source Computer Vision Library
LBP	Local Binary Patterns
ROCS	Recipient Operating Curves

CHAPTER 1
INTRODUCTION

1.1 BACKGROUND OF PROJECT

The face is our fundamental grouping of thought in social life accepting a basic part in passing on character and sentiments. We can see different faces adapted all through our future and perceive confronts at first notwithstanding taking after a long time of segment. This capacity is extremely solid paying little respect to tremendous assortments in visual shock due to advancing condition, developing and redirections, for instance, facial hair, glasses or changes in haircut. In picture handling, there are a considerable measure applications, for example, confront acknowledgment, question recognition, standardized identification filtering, and so on. One of the primary interesting issue in picture handling is face acknowledgment. Be that as it may, to do confront acknowledgment, the framework must have the capacity to identify the face first then it can just decide data, for example, the nearness of appearances, the pivot and stances of countenances. Confront location is critical in observation framework as it is the initial phase in the acknowledgment framework. In biometrics framework, components of confronts, for example, eyes, iris and retina can be separated from pictures to be grouped, however with the fruitful discovery of confronts first. Thus, confront discovery is the essential of picture preparing identified with appearances. Confront discovery is utilized as a part of many places these days particularly the sites facilitating pictures like Google photographs, photograph basin and face book. The subsequently naming highlight adds estimation to sharing pictures among the all inclusive community who are in the photograph and besides gives the considered who the individual is in the photo. In our wander, we have analyzed and executed a completely fundamental however amazingly intense face area figuring which considers human skin shading.

Our point, which we believe we have come to, was to develop a methodology for face affirmation that is speedy, lively, and sensibly essential and correct with respectably fundamental and clear computations and procedures.

1.2 PROBLEM STATEMENT

In the world of emerging technology, security became an essential component in day to day life. Information theft, lack of security and violation of privacy etc. are the essential components which are needed to be protected. Using smart secure systems for door lock and unlocking became popular nowadays. This is system is being adapted by many countries and first grade countries such as USA, Japan etc. already makes use of this system. This system provides either a facial recognition security feature or a keypad is provided to enter the pass code to unlock the door. Although it provides security to the doors, it also has its own drawbacks: Firstly, if the system mainly uses a facial recognition module, there might be a slight chance that sometimes the face may not be detected and hence the door cannot be unlocked. Secondly, if the system uses a keypad to enter the pass code to unlock the door, there might be a chance that the key maybe is recorded or can be observed by others without users consent. Hence, two-step verification is developed which makes use of facial recognition as the first step and pass code as its following step. But the same issues pertain in the newly developed system. Thus, a new model which rectifies all the above issues is developed.

1.3 OBJECTIVES

Based on the aforementioned problem statements, the objectives have been derived as following:

1. To detect and recognize the faces from image files from Raspberry Pi.
2. To unlock the door when face is correctly recognized.

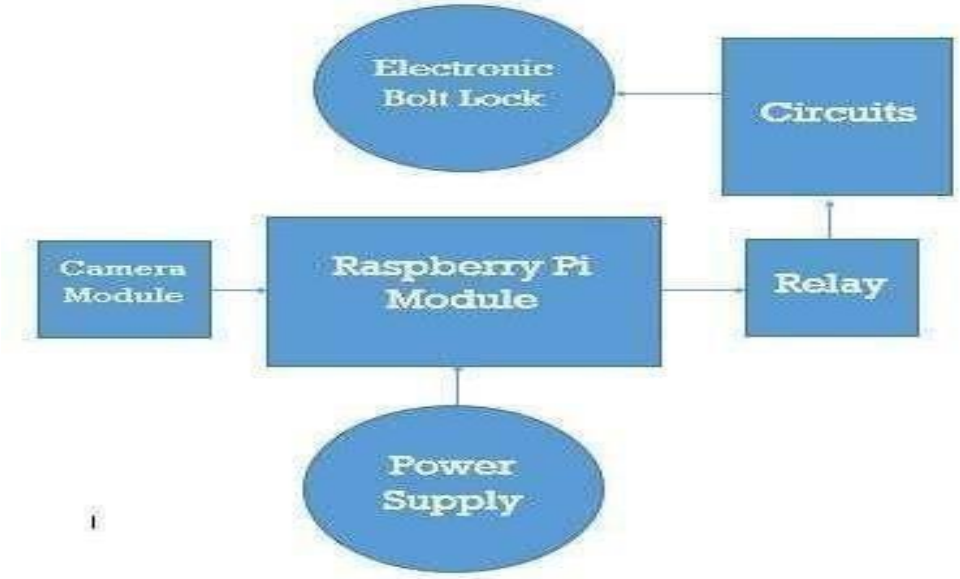


Figure 1.1: Working and Methodology

1.4 PROJECT SCOPE

This venture intends to identify faces from pictures in Raspberry Pi. In face discovery, the calculation ought to be powerful and quick. In this venture, the calculation utilized is proposed by Viola and Jones which has high identification rate, low false positive and negative rates and short computational time. Utilizing the proposed calculation, a face indicator is based on a Raspberry Pi Model B. After detection, we recognize the faces from the recognition algorithm. In the recognition part we recognize the different faces which are captured from pi cam.

CHAPTER 2

LITERATURE SURVEY

This chapter describes the past and current researches that have been carried out and the papers that have been published which are related to the project. This review investigates from various aspects of sensors, microcontrollers, indicators and the overall working system of the related projects.

2.1 REAL TIME FACE RECOGNITION USING RASBERRY PI 2

A Viji and A Pavithra have suggested the application of real time face recognition detect the faces. The overall project is discussed as below.

2.1.1 BASIC CONCEPT

There are heaps of decades in which this exploration chips away at acknowledgment and investigation have been completed for different applications identified with human machine connection. Just few methodologies have been utilized as a part of constant face acknowledgment. Confront identification is the initial phase in face acknowledgment of pictures. Viola Jones proposed a calculation that utilizations haar course classifier, course classifier. Wan-Hung Liao utilized the nearby paired example (LBP) administrator which is an effective neighborhood surface descriptor and it is utilized as a part of different applications. Tim Ahonen, 2006 utilized a novel and proficient facial picture portrayal which depends on nearby paired example and texture features. The LBP feature descriptor is extracted by dividing facial images into various regions. In 2014 kamlesh used active appearance model and local binary pattern where AAM is generic based approach and local binary pattern is hybrid approach. After the extraction of features different classifiers are used for face recognition, such as support vector model SVM, neural networks, adaboost classifier ,least mean square etc. the software testing is done by creating real time databases. Each directory contains 30 images of individual. Later hardware implementation is performed by raspberry pi along with hardware set up.

2.1.2 WORKING PRINCEPLE

The objective is to develop the real time face recognition from the facial images. it is performed using real time database which is collection of images from 10 subjects. The images of each subject is stored in specific directory with 30 images .Each image size is of about 100*130pixels. The viola Jones detection algorithm is used for face detection whereas PCA algorithm is used for feature extraction and then Adaboost classifier is used for face recognition. Raspberry pi II is the credit card sized computer which contains system on chip Broadcom BCM 2836 with CPU, GPU, DSP, SDRAM .The CPU is made up of quad core ARM cortex running at 200 MHz The architecture of proposed system is displayed in Figure b and explained as follows: The input image is captured from web camera and fed into the real time face recognition system as input. The real time face recognition is deployed with the Raspberry pi II which gives the identified image asoutput.This output is displayed in the monitor of the system.

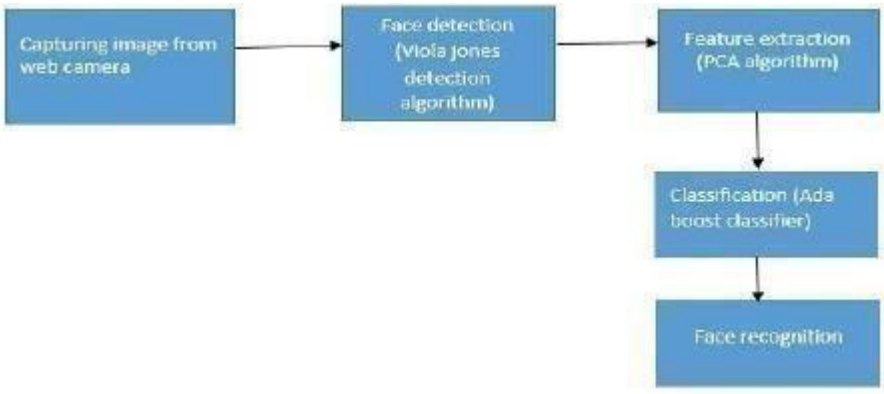


Figure 2.1. Block diagram of real time face recognition

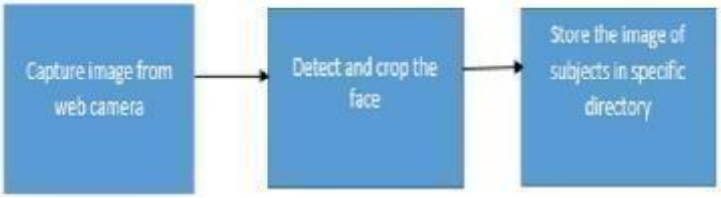


Figure 2.2. Creation of real time database

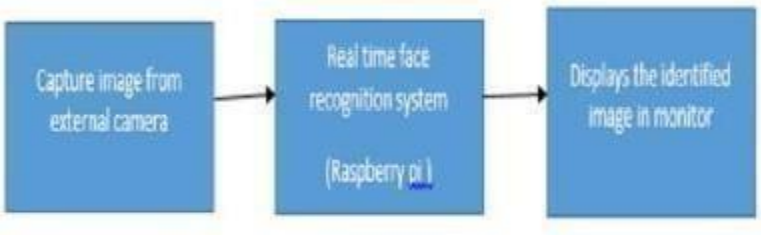


Figure 2.3. Real time face recognition using raspberry pi 4

2.2 Image-based Face Detection and Recognition: "State of the Art"

Faizan Ahmad, Aaima Najam and Zeeshan Ahmed have recommended the application Imagebased Face Detection and Recognition. The general venture is talked about as underneath.

2.2.1 BASIC CONCEPT

In current paper we built up a framework for the said strategy's assessment as a first point of referencefor video based face discovery and acknowledgment for observation. The diagram of current framework is exhibited in figure.

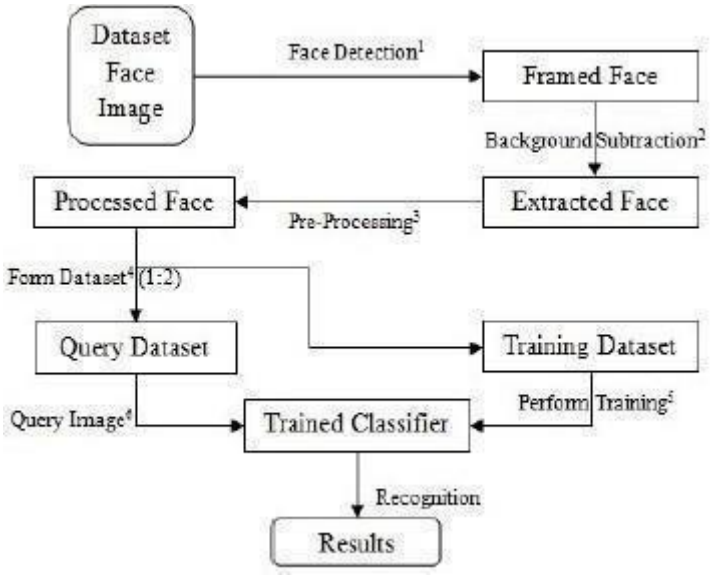


Figure 2.4. System's Overview

2.2.2 WORKING PRINCEPLE

2.2.2.1 FACE DETECTION

AdaBoost classifier is used with Haar and Local Binary Pattern highlights however Support Vector Machine classifier is used with Histogram of Oriented Gradients highlights for face revelation evaluation.Haar-like components are surveyed utilizing another photo depiction that creates an extensive plan of parts and usages the boosting computation AdaBoost to lessen degenerative tree of the helped classifiers for fiery and fast impedances simply direct rectangular Haar like components are used that gives different focal points like sort of uniquely delegated space learning is deduced and furthermore a speed increase over pixel based systems, suggestive to Haar commence limits practically identical to compel qualification readings are extremely easy to enroll. Execution of a system that used such parts would give a rundown of capacities that was outrageously broad, in this manner the rundown of abilities must be quite recently bound to couple of essential components which is refined by boosting computation, Adaboost. The primary LBP overseer names the pixels of a photo by thresholding the 3-by-3 neighbourhood of each pixel with the centre pixel regard and considering the result as a twofold number. Each face picture can be considered as a production of littler scale outlines which can be suitably perceived by the LBP chairman. To consider the shape information of goes up against, they detached face pictures into N little non-covering territories T0, T1... TN. The LBP histograms isolated from each sub-territory are then connected into a singular, spatially overhauled highlight histogram described as:Hi, j= _x,yI(fl(x,y)=i)I((x,y)_Tj) Where i = 0, ... , L-1; j = 0, ..., N-1. The removed element histogram portrays the neighborhood surface and worldwide state of face pictures.

Dataset	Detection		
	Adaboost		SVM
	Haar	LBP	HOG
[1]	99.31%	95.22%	92.68%
[2]	98.33%	98.96%	94.10%
[3]	98.31%	69.83%	87.89%
[4]	96.94%	94.16%	90.58%
[5]	90.65%	88.31%	89.19%
Mean	**96.70%**	**89.30%**	**90.88%**

Table 2.1: Face Detection Results Summery

2.2.2.2 FACE RECOGNITION

Eigen faces considered as 2-D goes up against affirmation issue, faces will be for the most part upright and frontal. That is the reason 3-D information about the face is not required that abatements multifaceted nature by a basic piece. It change over the face pictures into a course of action of introduce limits which essentially are the main fragments of the face pictures searches for headings in which it is more beneficial to address the data. This is generally significant for reduction the computational effort. Straight discriminant examination is on a very basic level used here to decrease the amount of components to a more anageable number before affirmation since face is addressed by a generous number of pixel qualities. Each of the new estimations is an immediate mix of pixel qualities, which outline an emplate. The immediate mixes got using Fisher's straight discriminant are called Fisher faces . LBP is a demand set of matched relationships of pixel strengths between within pixel and its eight incorporating pixels.

Table 2.2 : Face Recognition Results Summery

Dataset	Recognition			
	PCA	LDA	LBP	Gabor
[1]	72.10%	79.39%	85.93%	93.49%
[2]	69.87%	76.61%	80.47%	89.76%
[3]	70.95%	78.34%	84.14%	92.68%
[4]	74.79%	81.93%	86.45%	96.91%
[5]	68.04%	73.21%	77.69%	88.93%
Mean	**71.15%**	**77.90%**	**82.94%**	**92.35%**

Data Set	Sub-Division	Images	Resolution	Individuals	Image/Individual
A	Face 94	3078	180*200	153	~20
	Face 95	1440	180*200	72	20
	Face 96	3016	196*196	152	~20
	Grimace	360	180*200	18	20
B	Pain Expressions	599	720*576	23	26

A: Face Recognition Data, University of Essex
B: Psychological Image Collection at Stirling (PICS)

Table 2.3: Face Database Summery

2.3 Human Face Detection and Tracking Using Raspberry PI Processor

K. Shiva Prasad and M. Shirisha have suggested the application on Human Face Detection and Tracking Using Raspberry PI Processor. The overall project is discussed as below.

2.3.1 BASIC CONCEPT

This paper depicts the strategy for ongoing human face identification and following utilizing an altered variant of the calculation recommended by Paul viola and Michael Jones. The paper begins with the prologue to human face location and following, trailed by fear of the Vila Jones calculation and after that examining about the execution in genuine video applications. Viola Jones calculation depended on question discovery by extricating some particular elements from the picture. We utilized a similar approach for continuous human face location and following. Recreation after-effects of this created calculation demonstrate the Real time human face recognition and following supporting up to 50 human appearances. This calculation figures information and create brings about only a negligible portion of seconds. Here we are utilizing Raspberry Pi board as our stage. It has an ARM-11 SOC with incorporated peripherals like USB, Ethernet and serial and so on. On this board we are introducing Linux working framework with vital drivers for every single fringe gadget and client level programming stack which incorporates a light weight GUI in view of Server, XOrg centre product for associating with show gadgets like screens and show drivers, TCP/IP stack to communicate with system gadgets and some standard framework libraries for framework level general IO operations. By utilizing USB sort camera that is interfaced to the implanted board we can catch the live video of the specific area. The camera will ceaselessly catch the pictures and send it to ARM board. If an image is detected then microcontroller will send power to the motor driver unit then the electromagnetic lock will unlock the door and it will lock again when there is no power supply to that unit. If unknown person detected information is passed to authorized persons mail, so that he can unlock with android app if he wants to allow unknown person. Finally, this project concludes for the advanced implementations achieved by integrating embedded system models against the convention.

2.3.2 WORKING PRINCEPLE

With the fast advancement of human–machine interaction, full of feeling figuring is as of now picking up ubiquity in research and thriving in the business space. It expects to outfit figuring gadgets with easy and normal correspondence. The capacity to perceive human full of feeling state will enable the smart PC to translate, comprehend, and react to human. This is like the way that people depend on their faculties to evaluate each other's full of feeling state. Numerous potential applications, for example, clever car frameworks, diversion and media outlets, intelligent video, ordering and recovery of picture or video databases, can profit by this ability. Our framework is composed by utilizing ARM 32-bit miniaturized scale controller which underpins distinctive components and calculations for improvement of first facial acknowledgment. The webcam consolidates video detecting, video handling and communication inside a solitary gadget. It catches a video stream registers the data and exchanges the video stream to the ARM miniaturized scale controller. The picture it got is handled by utilizing picture preparing calculations and processed picture is grouped. In grouping Human is detected by utilizing Haar calculation and identified people are shown in plain view unit in particular format. Our framework is outlined by utilizing BSC2836 smaller scale processor created by BROADCOM which was called as Raspberry Pi.

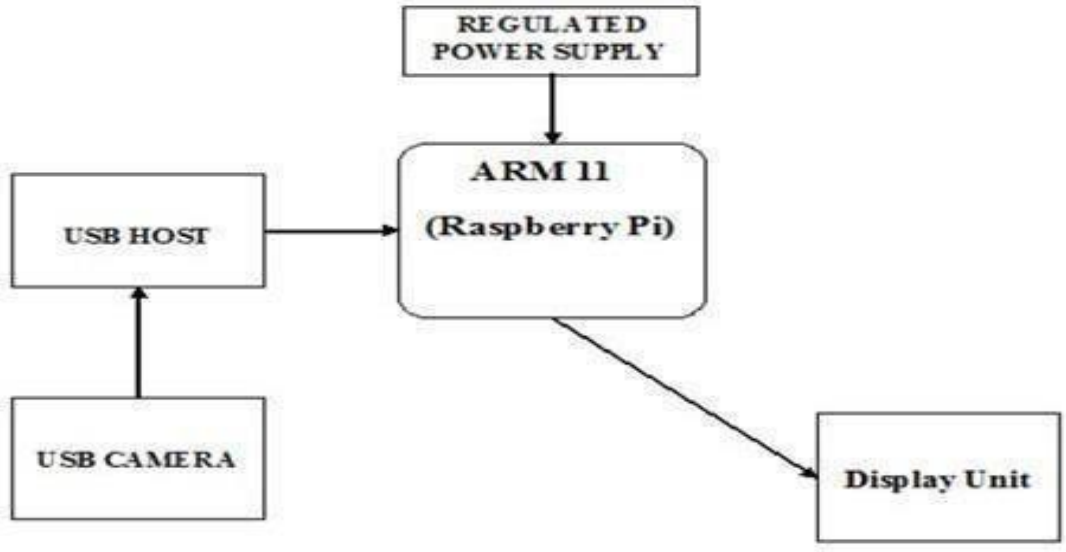


Figure 2.5. Block Diagram

CHAPTER 3

SYSTEM DEVELOPMENT

This section talks about the general venture advancement and combination of various modules. It comprises of four sections, to be specific the physical improvement, extend review, circuit outline and programming advancement.

3.1 GENERAL CONSTRUCTION

This venture means to outline a face locator on Raspberry Pi and its execution is assessed in view of identification rates and speed of recognitions the essential time of the wander is to grasp the count to do go up against recognizable proof. There are heaps of counts that can be used to do stand up to acknowledgment and this wander uses the falling Haar classifier proposed by Viola and Jones in 2001 . To get the venture working the entire calculations must be caught on. The second period of the venture is acknowledgment of the distinguished picture which is caught by pi cam.

3.2 HARDWARE TOOLS

3.2.1 RASPBERRY PI 4

The Raspberry Pi 4 is the third time Raspberry Pi. It has a 1.2GHz 64-bit quad-focus ARMv8 CPU, 802.11n Wireless LAN, Bluetooth 4.1, Bluetooth Low Energy (BLE), 4GB RAM, 4 USB ports, 40 GPIO pins, Full HDMI port, Ethernet port, united 3.5mm sound jack and composite video, Camera interface (CSI), Display interface (DSI), Micro SD card opening (now drive pull rather than push-push), Video Core IV 3D outlines focus. It is endorsed for some generally valuable use and in many wander based utilization.

Figure 3.1 – Raspberry 4

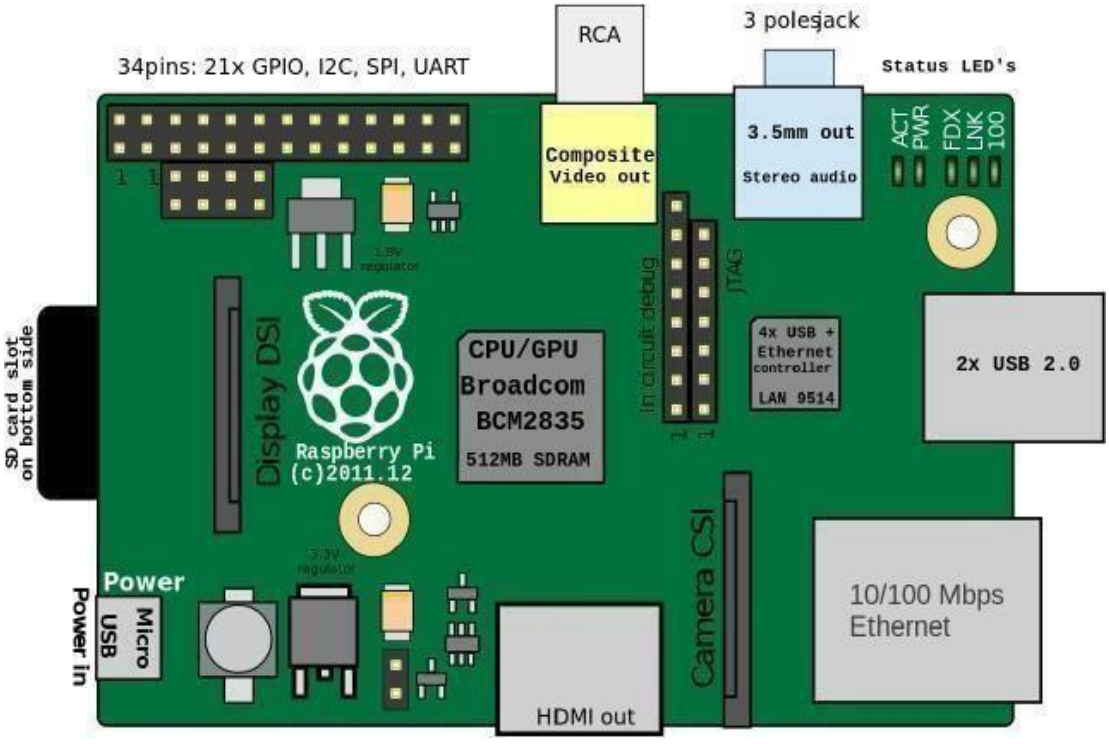


Figure 3.2. Layout of Raspberry Pi Model B

3.2.2 CAMERA MODULE USB

The usb camera module has a zebian imx219 ten-megapixel sensor. The advanced camera module might be utilized to take superior quality video, and stills pix. It's smooth to use for fledglings, however has masses to offer propelled clients in the event that you're hoping to grow your know-how. There are masses of cases online of individuals the utilization of it for time-slip by, steady development, and diverse video astuteness. You may moreover utilize the libraries we bundle manage the computerized camera to make comes about. The digital cam works with all designs of raspberry pi 1, 2,3 and 4. It can be gotten to by means of the mmal and v4l APIs, and there are different 0.33-festival libraries worked for it, alongside the picamera python library.

Figure 3.3– Camera Module USB

3.2.3 SD Card

A SD card is utilized to store the working arrangement of the Raspberry Pi. It likewise fills inas the capacity for all the bolster documents and programming for the face indicator and in addition stockpiling for info pictures to be tried.

Figure 3.4: SD Card

3.3 SOFTWARE DEVELOPMENT

3.3.1 SETTING UP RASPBIAN

Downloaded the most recent adaptation of Raspbian. Required a picture author to compose the downloaded OS into the little scale SD card. So downloaded the "win32 circle imager". Embedded the SD card into the tablet/pc and run the picture essayist. When open, peruse and chose the downloaded Raspbian picture document. Chosen the right gadget that is the drive speaking to the SD card. On the off chance that the drive (or gadget) chose is unique in relation to the SD card, then the other chose drive will end up noticeably adulterated. So be watchful. Once the compose is finished, launch the SD card and insert it into the Raspberry Pi and turn it on. It should start booting up. in the wake of booting the Pi, there might be conditions when the customer capabilities like the "username" and mystery key will be asked. Raspberry Pi goes with a default customer name and mystery key therefore constantly use it at whatever point it is being asked. The accreditations are;

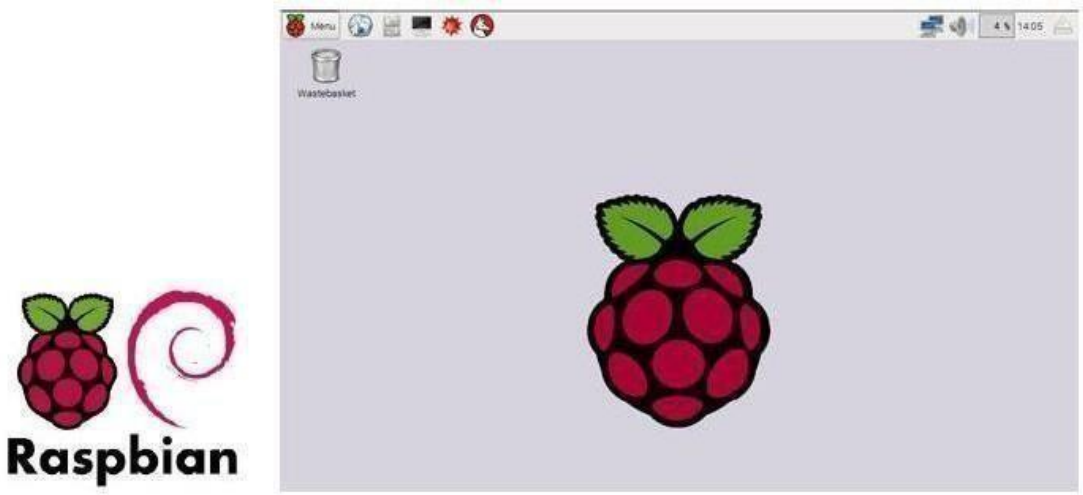


Figure 3.5. Logo and Desktop interface of Raspbian

login: pi

Password: raspberry

At the point when the Pi has been booted surprisingly, an arrangement screen called the "Setup Options" ought to show up and it will resemble the picture underneath. In the event that you have missed the "Setup Options" screen, it's not an issue, you can simply get it by writing the accompanying summon in the terminal.

$ sudo raspi-config

Since the Setup Options window is up, we set a couple of things. The main thing we did was to choose the principal choice in the rundown of the setup alternatives window, that is select the "Grow File system" choice and hit the enter key. We do this to make utilization of all the space introduce on the SD card as a full segment. This does is, grow the OS to fit the entire space on the SD card which can then be utilized as the capacity memory for the Pi. The second thing we did was to choose the third choice in the rundown of the setup alternatives window, that is select the "Empower Boot to Desktop/Scratch" choice and hit the enter key. It will take you to another window called the "pick boot alternative" window that resembles the picture beneath. In the "pick boot elective window", select the second decision, that is, "Desktop Log in as customer "pi" at the graphical desktop" and hit the enter get. e it at whatever point it is being asked. Once done you will be reclaimed to the "Setup Options" page, if not choose the "alright" catch at the base of this window and you will be reclaimed to the past window. We do this since we need to boot into the desktop condition which we know about.

On the off chance that we don't do this progression, then the Raspberry Pi boots into a terminal each time with no GUI alternatives.

Once, both the means are done, select the "complete" catch at the base of the page and it ought to reboot consequently. On the off chance that it doesn't, then utilize the accompanying summon in the terminal to reboot. $ sudo reboot

After the reboot from the past walk, if everything went right, then you will end up on the desktop. When you are on the desktop, open a terminal and enter the going with charge to revive the firmwareof the Pi.

$ sudo reboot

After the reboot from the past walk, if everything went right, then you will end up on the desktop. When you are on the desktop, open a terminal and enter the going with charge to revive the firmware of the Pi.**$ sudo rpi-update**

$ sudo apt-get update

$ sudo apt-get upgrade

$ sudo reboot

The most recent firmware may have the settle to those bugs, in this way it's essential to refresh it in the first place itself.

3.3.2 INSTALLING OPENCV 3 ON A RASPBERRY PI

OPENCV

OpenCV is released under a BSD allow and thus it's free for both insightful and business use.

It has C++, C, Python and Java interfaces and support Windows, Linux, Mac OS, iOS and Android. OpenCV was proposed for computational capability and with a strong focus on continuous applications. Written in upgraded C/C++, the library can abuse multi-focus taking care of. Enabled with OpenCL, it can abuse the hardware accelerating of the fundamental heterogeneous process organize. Gotten all around the world, OpenCV has more than 47 thousand people of customer gathering and surveyed number of downloads outperforming 9 million. Usage ranges from instinctive craftsmanship, to mines examination, sewing maps on the web or through forefront robotics.

OpenCV Library

The open source PC vision library, OpenCV, started as an examination extends at Intel in 1998. It has been accessible since 2000 under the BSD open source permit. OpenCV is gone for giving the devices expected to tackle PC vision issues. It contains a blend of low-level picture handling capacities and abnormal state calculations, for example, confront location, person on foot identification, include coordinating, and following. OpenCV's GPU module incorporates countless, and numerous of them have been actualized in various forms, for example, the picture sorts (scorch, short, drift), number of channels, and outskirt extrapolation modes. This makes it trying to report correct execution numbers. An additional wellspring of trouble in refining the execution numbers down is the overhead of synchronizing and exchanging information. This implies best execution is acquired for extensive pictures where a great deal of preparing should be possible while the information lives on the GPU. To help the engineer make sense of the exchange offs; OpenCV incorporates an execution benchmarking suite that runs GPU capacities with various parameters and on various datasets. This gives a nitty gritty benchmark of how entirely different datasets are quickened on the client's equipment.

The main thing to do is to extend your file system to incorporate all accessible space on your small scale SD card.

```
$ sudo raspi-config
```

Once provoked, we chose the primary choice, "1. Extend File System", hit Enter on our console, bolt down to the "<Finish>" catch, and afterward reboot your Pi:

In the wake of rebooting, your record structure should have been stretched out to join all available space on your littler scale SD card. You can watch that the plate has been reached out by executing df - h and taking a gander at the yield:

```
$ df -h
```

OpenCV, alongside every one of its conditions, will require a couple of gigabytes amid the gather, so you ought to erase the Wolfram motor to free up some space on your Pi.

$ sudo reboot

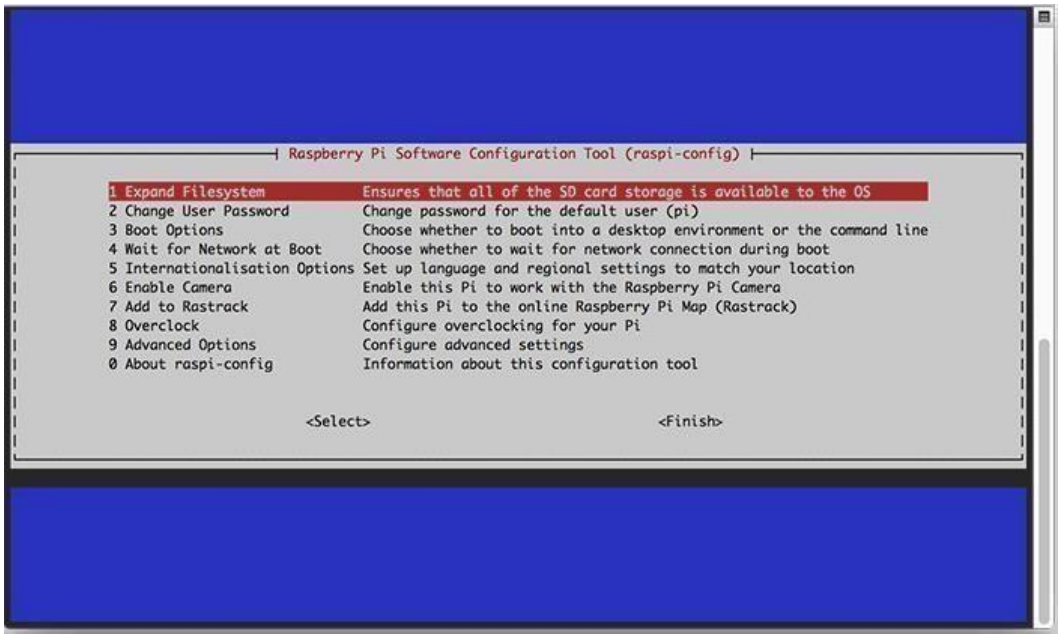


Figure 3.6- Expanding the file system on your Raspberry Pi 3

In the wake of rebooting, your record structure should have been stretched out to join all available space on your littler scale SD card. You can watch that the plate has been reached out by executing df - h and taking a gander at the yield:

$ df -h

OpenCV, alongside every one of its conditions, will require a couple of gigabytes amid the gather, so you ought to erase the Wolfram motor to free up some space on your Pi

$ sudo apt-get purge wolfram-engine

The initial step is to refresh and update any current bundles:

$ sudo apt-get update

$ sudo apt-get upgrade

We then need to introduce some designer devices, including CMake, which helps us arrange the OpenCV assemble handle:

$ sudo apt-get install build-essential cmake pkg-config

Next, we need to present some photo I/O packages that empower us to stack diverse picture archive outlines from circle. Instances of such record bunches fuse JPEG, PNG, TIFF, et cetera.

$ sudo apt-get install libjpeg-dev libtiff5-dev libjasper-dev libpng12-dev

Essentially as we need picture I/O groups, we similarly require video I/O packs. These libraries empower us to scrutinize diverse video record bunches from hover and likewise work particularly with video streams:

$ sudo apt-get install libavcodec-dev libavformat-dev libswscale-dev libv4l-dev

$ sudo apt-get install libxvidcore-dev libx264-dev

The OpenCV library goes with a sub-module named high-up which is used to show pictures to our screen and make fundamental GUIs. Remembering the ultimate objective to arrange the highgui module, we need to present the GTK change library:

$ sudo apt-get install libgtk2.0-dev

Numerous operations within OpenCV (to be specific grid operations) can be enhanced further by introducing a couple of additional conditions:

$ sudo apt-get install libatlas-base-dev gfortran

These upgrade libraries are especially crucial for resource obliged contraptions, for instance, the Raspberry Pi.

Eventually, we should present both the Python 2.7 and Python 3 header records so we can fuse OpenCV with Python ties:

$ sudo apt-get install python2.7-dev python3-dev

On the off chance that you skirt this progression, you may see a mistake identified with the Python.h header record not being found when running make to accumulate OpenCV.

Since we have our conditions introduced, how about we snatch the 3.1.0 file of OpenCV from the authority OpenCV archive. (Note: As future renditions of openCV are discharged, you can supplant 3.1.0 with the most recent adaptation number):

$ cd ~

$ wget -O opencv.zip https://github.com/Itseez/opencv/archive/3.1.0.zip

$ unzip opencv.zip

We'll need the full introduce of OpenCV 3 (to have entry to components, for example, SIFT and SURF, for example), so we likewise need to snatch the opencv_contribution storehouse too:

$ wget -O opencv_contrib.zip https://github.com/Itseez/opencv_contrib/archive/3.1.0.zip

$ unzip opencv_contrib.zip

You may need to broaden the summon above using the "<=>" get in the midst of your copy and paste. The .speed in the 3.1.0.zip may have every one of the reserves of being cutoff in a couple programs. The full URL of the OpenCV 3.1.0 report is:

$ https://github.com/Itseez/opencv_contrib/archive/3.1.0.zip

Note: Make beyond any doubt your OpenCV and opencv_contrib renditions are the same (for this situation, 3.1.0). On the off chance that the renditions numbers don't coordinate, then you'll likely keep running into either order time or runtime.

Before we can begin accumulating OpenCV on our Raspberry Pi 3, we initially need to introduce pip, a Python bundle chief:

$ wget https://bootstrap.pypa.io/get-pip.py

$ sudo python get-pip.py

At first, it's basic to appreciate that a virtual circumstance is a one of a kind gadget used to keep the conditions required by different exercises in parceled puts by making isolated, free Python circumstances for each of them.

Basically, it disentangles the "Amplify X depends on upon adjustment 1.x, yet Project Y needs 4.x" circumstance. It moreover keeps your overall site-groups flawless, clean, and free from chaos.

If you may need a full illumination on why Python virtual circumstances are awesome practice, absolutely give this shocking web journal section on Real Python a read.

It's standard practice in the Python society to use virtual circumstances or something to that effect, so I extremely endorse that you do in like manner:

$ sudo pip install virtualenv virtualenvwrapper

$ sudo rm -rf ~/.cache/pip

Since both virtualenv andvirtualenvwrapper have been presented, we need to invigorate our ~/.profile report to fuse the going with lines at the base of # virtualenv and virtualenvwrapper

export WORKON_HOME=$HOME/.virtualenvs
source /usr/local/bin/virtualenvwrapper.sh

In past instructional exercises, I've prescribed utilizing your most loved terminal-based content tool, for example, vim,emacs, or nano to refresh the ~/.profile document. In case you're alright with these editors, simply ahead and refresh the document to mirror the progressions specified previously. Else, you ought to just utilize feline and yield redirection to deal with refreshing.

$ echo -e "\n# virtualenv and virtualenvwrapper" >> ~/.profile

$ echo "export WORKON_HOME=$HOME/.virtualenvs" >> ~/.profile

$ echo "source /usr/local/bin/virtualenvwrapper.sh" >> ~/.profile

Since we have our ~/.profile refreshed, we have to reload it to ensure the progressions produce results.

You can compel a reload of your ~/.profile document by:

Logging out and afterward logging back in.

Shutting a terminal occasion and opening up another one

Or, on the other hand my undisputed top choice, simply utilize the source order

$ Source ~/.profile

Next, we should make the Python virtual condition that we'll use for PC vision advancement:

$ mkvirtualenv cv -p python2

This order will make another Python virtual condition named cv utilizing Python 2.7.

In the event that you rather need to utilize Python 3, you'll need to utilize this charge:

$ mkvirtualenv cv -p python3

If you ever reboot your Raspberry Pi; log out and log back in; or open up another terminal, you'll need to use the workon request to re-get to the cv virtual condition. In past blog passages, I've seen perusers use the mkvirtualenv summon — this is totally unneeded! The mkvirtualenv request is planned to be executed only once: to truly make the virtual condition.

Starting there ahead, you can use deal with and you'll be dropped down into your virtual condition:

e of the record:

$ Source ~/.profile

$ workon cv

To approve and guarantee you are in the cv virtual condition, look at your summon line — in the event that you see the content (cv) going before your incite, then you are in the cv virtual condition:

$ workon cv

When you have promised you are in the cv virtual condition, we can setup our produce using CMake:

$ cd ~/opencv-3.1.0/

Figure 3.7- The "(cv)" message on your provoke, showing that you are in the cv virtual

Our exclusive Python reliance is NumPy, a Python bundle utilized for numerical handling:

$ pip install numpy

We are as of now arranged to arrange and present OpenCV! Twofold watch that you are in thecv virtual condition by taking a gander at your incite (you should see the (cv) content going before it), and if not, simply execute take a shot at:

$ mkdir build

$ cd build

**$ cmake -D CMAKE_BUILD_TYPE=RELEASE **

-D CMAKE_INSTALL_PREFIX=/usr/local \

**-D INSTALL_PYTHON_EXAMPLES=ON **

-D OPENCV_EXTRA_MODULES_PATH=~/opencv_contrib-3.1.0/modules \

-D BUILD_EXAMPLES=ON

In the event that you are incorporating OpenCV 3 for Python 2.7, then ensure your Python 2 area incorporates substantial ways to the Interpreter, Libraries, numpy and bundles way, like my screenshot underneath:

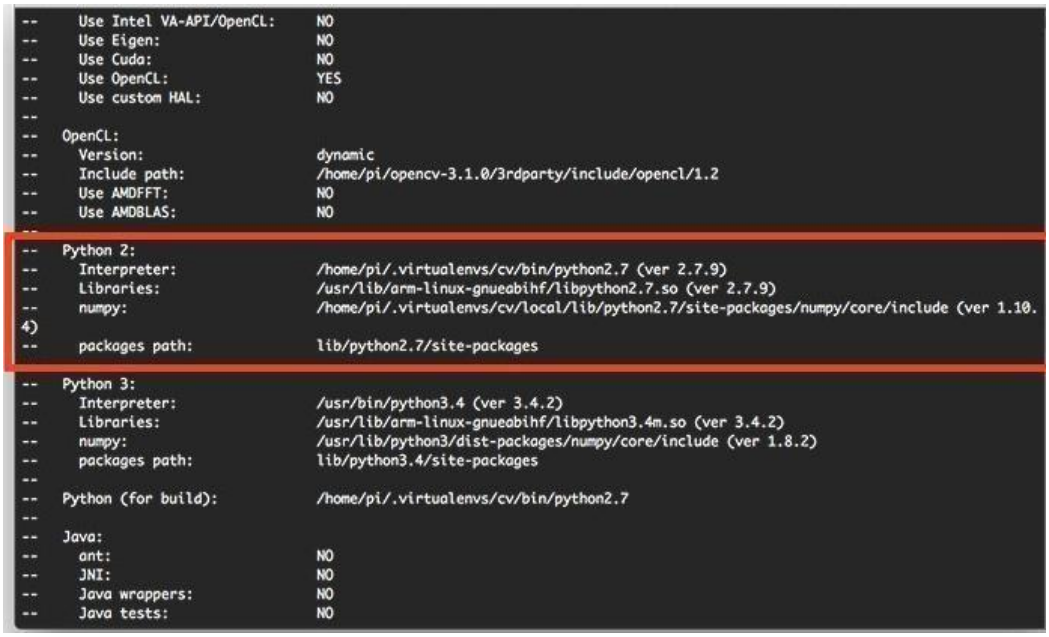


Figure 3.8 - Ensuring that Python 2.7 will be utilized when gathering OpenCV 3 for Raspbian Jessieon the Raspberry Pi 3.

Provided that this is true, get to the cv virtual condition using workon cv and re-run the cmake arrange plot above.Finally, we are directly arranged to accumulate OpenCV:

$ make -j4

The - j4 charge controls the amount of focuses to utilize when organizing OpenCV 3. The Raspberry Pi 3 has four focuses; along these lines we supply an estimation of 4 to empower OpenCV to total speedier. Notwithstanding, due to race conditions, there are times when make mistakes out when utilizing different centres. In the event that this transpires, I recommend beginning the aggregation once again and utilizing just a single centre::

$ make clean

$ make

From that point, you should simply introduce OpenCV 3 on your Raspberry Pi 3

$ sudo make install

$ sudo ldconfig

OpenCV ought to now be introduced in/usr/neighborhood/lib/python2.7/site-pacakges. You can confirm this utilizing the ls charge:

$ ls -l /usr/local/lib/python2.7/site-packages/

Total 1852

Our last stride is to sym-interface the OpenCV ties into our cv virtual condition for Python 2.7:

$ cd ~/. virtualenvs/cv/lib/python2.7/site-packages/

$ ln -s /usr/local/lib/python2.7/site-packages/cv2.so cv2.so

How about we initially confirm that your OpenCV establishment is working appropriately. Open up another terminal; execute the source andwork on charges, and after that at last endeavour to import the Python + OpenCV ties:

$ source ~/.profile

$ workon cv

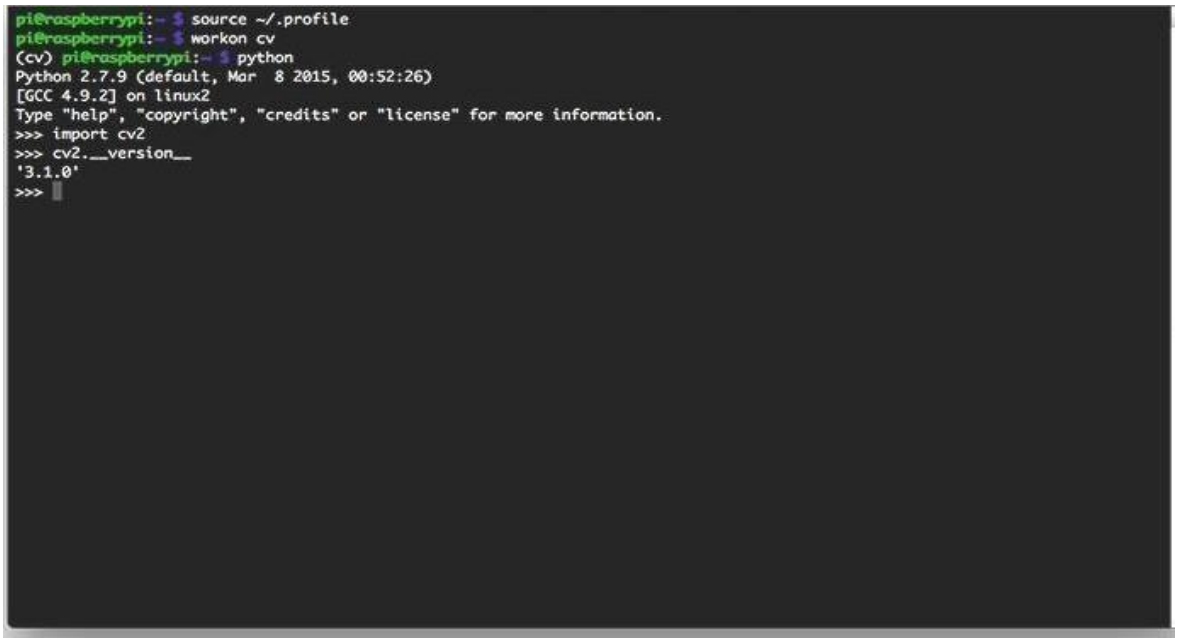


Figure 3.9- Affirming OpenCV 3 has been effectively introduced on my Raspberry Pi 3 runningRaspbian

3.3.3 FACE DETECTION ALGORITHM

A Haar-like component considers neighboring rectangular areas at a specific region in an area window, adds up to up the pixel controls in each locale and figures the differentiation between these totals. This refinement is then used to request subsections of a picture. An instance of this would be the revelation of human appearances. For the most part, the districts around the eyes are darker than the extents on the cheeks. One instance of a Haar-like component for face acknowledgment is thus a plan of two neighboring rectangular locales over the eye and cheek ranges.

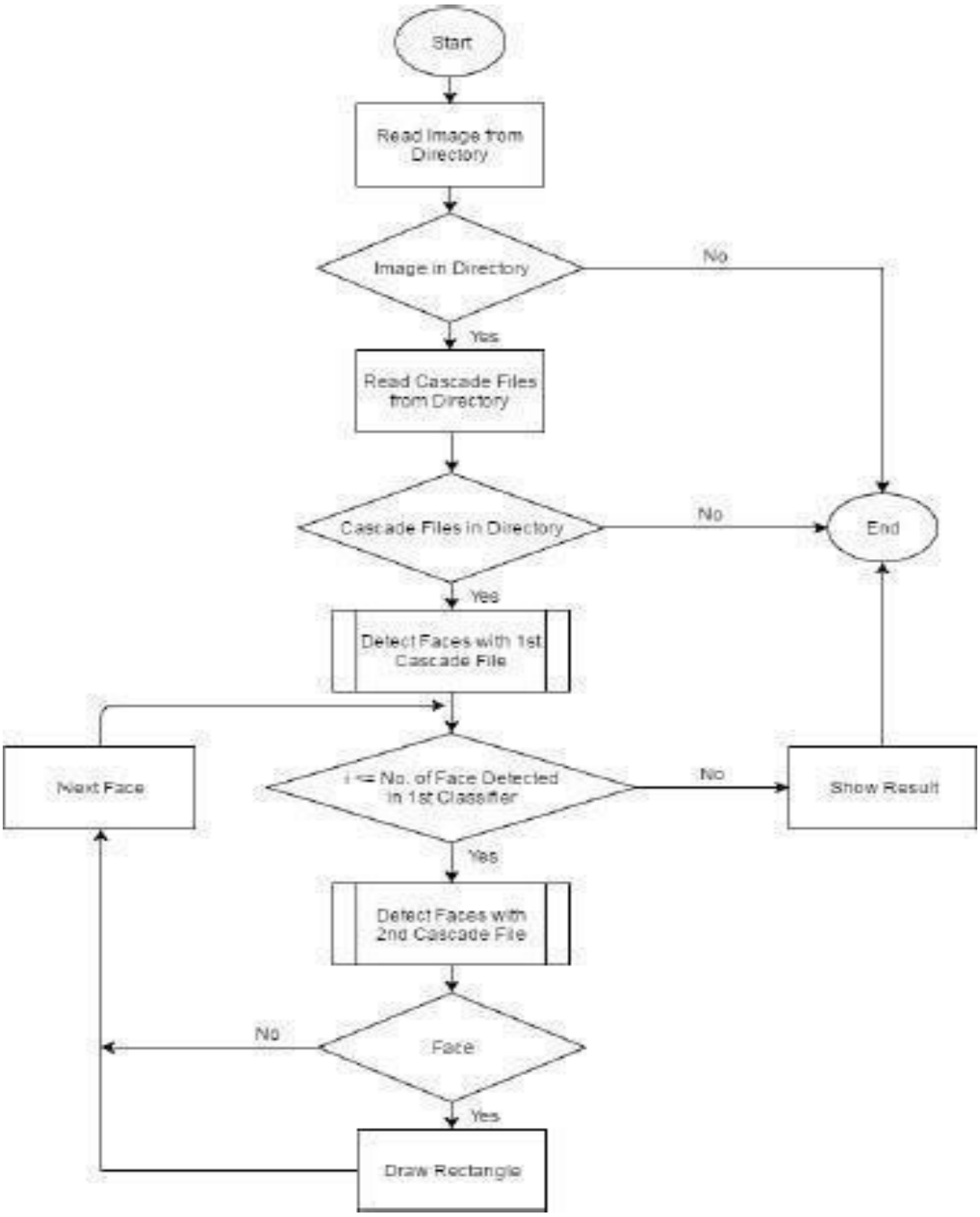


Figure 3.10. Flowchart for face detection for image files

Cascade classifier

The course classifier comprises of a rundown of stages, where each stage comprises of a rundown of feeble learners. The framework identifies protests being referred to by moving a window over the picture. Each phase of the classifier marks the particular district characterized by the present area of the window as either positive or negative – positive implying that a question was found or negative implies that the predetermined protest was not found in the picture. On the off chance that the naming yields a negative outcome, then the order of this particular locale is therefore total and the area of the window is moved to the following area. On the off chance that the naming gives a positive outcome, then the locale moves of to the following phase of arrangement. The classifier yields a last decision of positive, when every one of the stages, including the last one, yield an outcome, saying that the question is found in the picture.

A genuine positive implies that the protest being referred to is to be sure in the picture and the classifier marks it all things considered – a positive outcome. A false positive implies that the marking procedure dishonestly establishes that the protest is situated in the picture, in spite of the fact that it is most certainly not. A false negative happens when the classifier can't distinguish the real protest from the picture and a genuine negative implies that a non-protest was accurately classifier as not being the protest being referred to.

Keeping in mind the end goal to function admirably, each phase of the course should have a low false negative rate, in light of the fact that if the real protest is named a non-question, then the characterization of those branch stops, with no real way to amend the oversight made. Be that as it may, each stage can have a generally high false positive rate, in light of the fact that regardless of the possibility that the n-th arrange orders the non-protest as really being the question, then this oversight can be settled in n+1-th and resulting phases of the classifier.

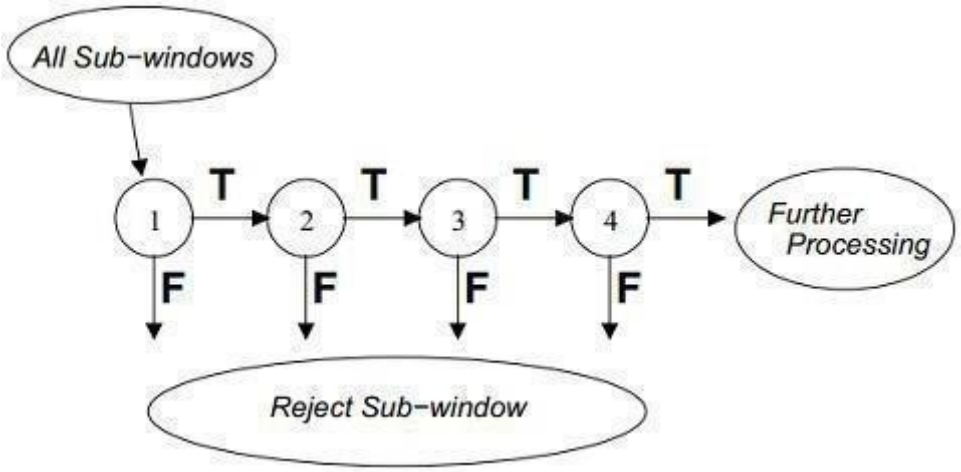


Figure 3.11 - Stages of the cascade classifier

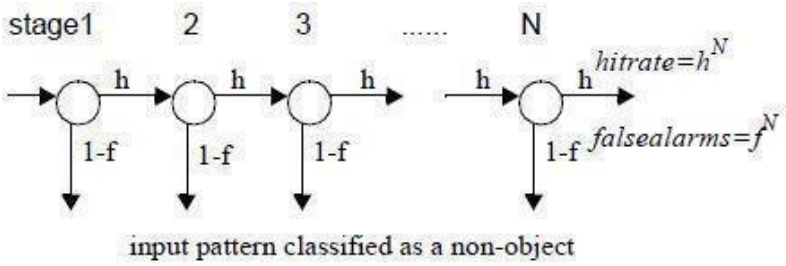


Figure 3.12- Cascade of classifiers with N stages. At each stage a classifier is trained to achievea hit rate of h and a false alarm rate of f.

The human face acts extensively a bigger number of issues than various things since the human face is a dynamic challenge that comes in many structures and shades. In any case, facial area and taking after gives many points of interest. Facial affirmation is unrealistic if the face is not segregated from the establishment. Human Computer Interaction (HCI) could altogether be improved by using feeling, stance, and movement affirmation, all of which require face and facial component acknowledgment and taking after. Yet an extensive variety of figuring's exist to perform stand up to distinguishing proof, each has its own specific inadequacies and qualities. Some use tissue tones, some usage shapes, and other are fundamentally more eccentric including formats, neural frameworks, or channels. These estimations encounter the

evil impacts of a comparable issue; they are computationally exorbitant. Aphoto is only a social affair of shading or possibly light power values. Looking at these pixels for face revelation is monotonous and difficult to accomplish because of the wide assortments of shape and pigmentation inside a human face. Pixels as often as possible require reanalysis for scaling and precision. Viola and Jones imagined a figuring, called Haar Classifiers, to rapidly recognize any challenge, including human goes up against, using AdaBoost classifier falls that rely on upon Haar-like

Haar Cascade Classifiers

The center clarification behind Haar classifier disagree revelation is the Haar-like segments. These portions, as opposed to utilizing the power estimations of a pixel, utilize the change of course values between adjoining rectangular social events of pixels. The qualification changes between the pixel parties are utilized to pick relative light and lessen ranges. Several coterminous social events with a relative multifaceted nature change shape a Haar-like part.Haar-like parts, as appeared in Figure 3 are utilized to perceive a photograph. Haar parts can without a considerable measure of an augment be scaled by developing or diminishing the measure of the pixel get-together being inspected. This engages fragments to be utilized to see objects of different sizes.

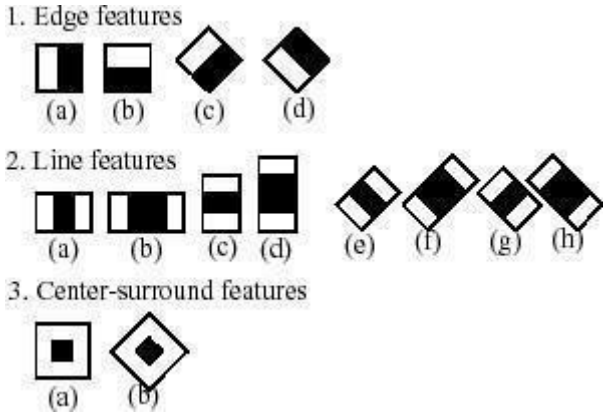


Figure 3.13 - Common Haar Features

Integral Image

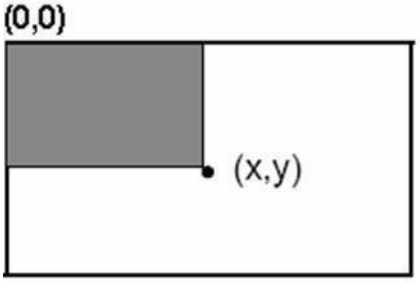
The straightforward rectangular components of a picture are computed utilizing a transitional portrayal of a picture, called the fundamental picture. The basic picture is a cluster containing the aggregates of the pixels' power values found straightforwardly to one side of a pixel and specifically over the pixel at area (x, y) comprehensive. So if A [x, y] is the first picture and AI

[x, y] is the indispensable picture then the basic picture is registered as appeared in condition 1 and represented in Figure 3.4.

$$AI[x, y] = \sum_{x_F \le x,\, y_F \le y,} A(x', y') \qquad (1)$$

The components pivoted by forty-five degrees, similar to the line include appeared in Figure 3 2(e), as presented by Lienhart and Maydt, require another transitional portrayal called the turned basic picture or pivoted whole assistant picture. The pivoted essential picture is ascertained by finding the total of the pixels' force values that are situated at a forty-five- degree point to one side or more for the x esteem and beneath for the y esteem. So if A [x, y] is the first picture and AR [x, y] is the pivoted fundamental picture then the indispensable picture is processed as appeared in condition 2 a delineated in Figure 3.5.

$$AR[, y] = \sum_{x_F \le x,\, x_F \le x - |y - y_F|,} A(x'y)$$



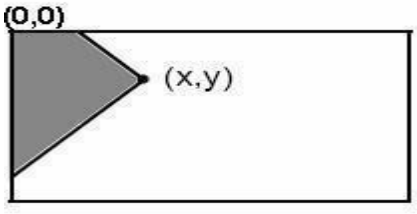


Fig 3.14 - Summed area of integral image Fig 3.15 - Summed area of rotated integral image

It just takes two goes to enroll both essential picture shows, one for each group. Using the reasonable crucial picture and taking the differentiation between six to eight bunch segments encircling a couple related rectangles, a component of any scale can be figured. Along these lines learning a component is to an incredible degree speedy and gainful. It moreover suggests finding out components of various sizes requires an unclear effort from a component of only a couple of pixels. The revelation of various sizes of a comparative challenge requires a vague measure of effort and time from objects of tantamount sizes since scaling requires no additional effort.

Training Classifiers for Facial Features

Recognizing human facial components, for instance, the mouth, eyes, and nose require that Haar classifier falls at first be readied. Remembering the ultimate objective to set up the

classifiers, this fragile AdaBoost count and Haar incorporate estimations must be executed. Fortunately, Intel developed an open source library focused on encouraging the use of PC vision related activities called Open Computer Vision Library (OpenCV). The OpenCV library is proposed to be used as a piece of conjunction with applications that identify with the field of HCI, mechanical innovation, biometrics, picture dealing with, and diverse locales where portrayal is basic and joins an execution of Haar classifier acknowledgment and planning. To set up the classifiers, two game plans of pictures are required. One set contains a photo or scene that does not contain the question, for this circumstance a facial component, which will be recognized. This game plan of pictures is implied as the negative pictures. The other course of action of pictures, the positive pictures, contains no less than one events of the challenge.

The region of the things inside the positive pictures is demonstrated by: picture name, the left name, the upper left pixel and the height, and width of the challenge. For get ready facial components 5,000 negative pictures with no not as much as a super pixel assurance were used for get ready. These photos included common things, like paperclips, and of trademark scene, like photographs of woodlands and mountains. Remembering the ultimate objective to convey the most energetic facial part revelation possible, the principal valuable plan of pictures ought to be illustrative of the change between different people, including, race, sexual introduction, and age. An OK hotspot for these photos is National Institute of Standards and Technology's (NIST) Facial Recognition Technology (FERET) database. This database contains more than 10,000 pictures of more than 1,000 people under different lighting Conditions, stances, and focuses . In setting up each facial component, 1,500 pictures were used. These photos were taken at focuses going from zero to forty-five degrees from a frontal view. This gives the required contrast required to allow distinguishing proof if the head is turned possibly. Three separate classifiers were readied, one for the eyes, one for the nose, and one for the mouth. Once the classifiers were readied, they were used to perceive the facial components inside another game plan of pictures from the FERET database. The exactness ofthe classifier was then handled as showed up in Table 1. Aside from the mouth classifier, the classifiers have a high rate of area. Nevertheless, as proposed by, the false positive rate is furthermore high.

Facial Feature	Positive Hit Rate	Negative Hit Rate
Eyes	93%	23%
Nose	100%	29%
Mouth	67%	28%

Table 3.1 - Accuracy of Classifiers

The underlying stage in facial component disclosure is distinguishing the face. This requires looking at the entire picture. The second step is using the detached face(s) to perceive every part. The result is showed up in Figure 4. Since each the piece of the photo used to distinguish a component is considerably more diminutive than that of the whole picture, acknowledgment of each one of the three facial components takes less time things being what they are than perceiving the face itself. Using a 1.2GHz AMD processor to dismember a 320 by 240 picture, a packaging rate of 3 edges for each second was refined. Since an edge rate of 5 edges for each second was proficient in facial revelation just by using a considerably speedier processor, regionalization gives a colossal augmentation in efficiency in facial component ID. Regionalization also staggeringly extended the precision of the area. Every single false positive were wiped out, giving an area rate of around 95% for the eyes and nose. The mouth revelation has a lower rate due to the base size required for recognizable proof. By changing the height and width parameter to more definitely address the estimations of the mouth and retraining the classifier the precision should manufacture the precision to that of interchange components.

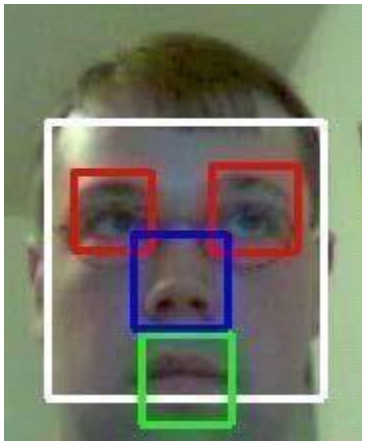


Figure 3.16 - Detected Objects: Face (white), Eyes (red), Nose (blue), and Mouth (green)

NUMPY ARRAYS

In the Python world, Numpy shows are the standard depiction for numerical data. Here, we show how these bunches enable gainful utilization of numerical counts in an anomalous state lingo. By and large, three frameworks are associated with improve execution: vector sing estimations, refraining from copying data in memory, and constraining operation checks. The Python programming lingo gives a rich course of action of strange state data structures: records for posting a collection of articles, vocabularies to produce hash tables, et cetera. In any case, these structures are not ideally suited to prevalent numerical figuring. In the mid- 90s, an all inclusive gathering of volunteers started to develop a data structure for beneficial display estimation. This structure progressed into what is by and by known as the N dimensional NumPy display. The NumPy package, which contains the NumPy display and furthermore a game plan of running with logical limits, has found extensive allocation in the academic world, national research focuses, and industry, with applications going from gamingto space examination. A NumPy display is a multidimensional, uniform social event of segments. A display is depicted by the kind of parts it contains and by its shape. For example, a system may be addressed as an assortment of shape (M×N) that contains numbers, e.g., floating point or complex numbers. Not in the least like systems, can NumPy bunches have any dimensionality. Moreover, they may contain distinctive sorts of segments (or even mixes of parts, for instance, Booleans or dates. Underneath the hood, a NumPy show is really just a beneficial technique for depicting no less than one bits of PC memory, so that the numbers addressed may be easily controlled.

Computer Vision

Consider a n × 3 cluster of three dimensional focuses and a 3 × 3 camera grid:

```
Points = np.random.random((100000, 3)) camera =
np.array([[500., 0., 320.], [ 0., 500., 240.], [ 0., 0., 1.]])
```

Regularly, we need to change the 3D arranges into their 2D pixel areas on the picture, as seen by the camera. This operation includes taking the network speck result of each point with the camera lattice, and after that partitioning the subsequent vector by its third segment. With NumPy, it is composed as:

```
# Perform the matrix product on the coordinates vecs =
camera. dot (points).T
```

 # Divide resulting coordinates by their z-value pixel_coords

= vecs/vecs [:, 2, np.newaxis]

The spot work executes the lattice item, rather than the component shrewd item. It can be connected to maybe a couple dimensional clusters. This code executes in 9 milliseconds–a 70x speedup over a Python for-circle form. Beside the enhanced NumPy dab item, we make utilization of NumPy's exhibit operations with component by-component division and the telecom apparatus. The code new_vecs/new_vecs [:, 2, np.newaxis] separates every section of new_vecs by its third segment (at the end of the day, each column is isolated by its third component). The np.newaxis file is utilized to change new_vecs [:, 2] into a column-vector so that broadcasting may take place.

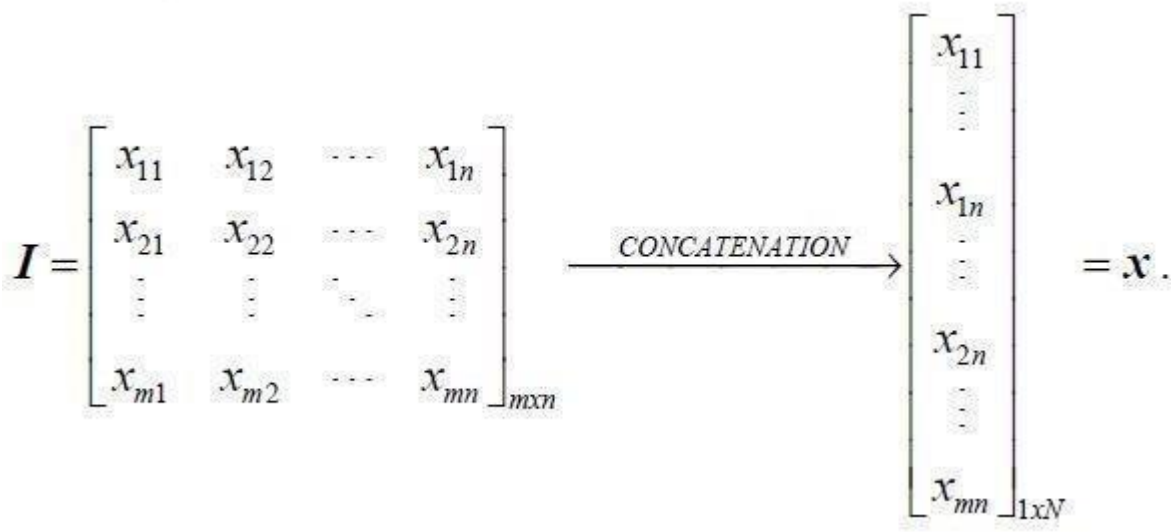
3.3.4 FACE RECOGNITION

One of the least complex and best PCA approaches utilized as a part of face acknowledgment frameworks is the supposed eigenface approach. This approach changes faces into a little arrangement of basic attributes, eigenfaces, which are the fundamental parts of the underlying arrangement of learning pictures (preparing set). Acknowledgment is finished by anticipating another picture in the eigenface subspace, after which the individual is grouped by contrasting its position in eigenface space and the position of known people The upside of this approach over other face acknowledgment frameworks is in its Effortlessness, speed and harshness to little or progressive changes on the face. The issue is constrained to records that can be utilized to perceive the face. In particular, the pictures must be vertical frontal perspectives of human appearances.

PCA Approach to Face Recognition [5]

Essential part investigation changes an arrangement of information gotten from perhaps related factors into an arrangement of estimations of uncorrelated factors called vital segments. The quantity of parts can be not exactly or equivalent to the quantity of unique factors. The primary important segment has the most astounding conceivable fluctuation, and each of the succeeding part has the most elevated conceivable difference under the limitation that it must be orthogonal to the past segment. We need to discover the foremost parts, for this situation eigenvectors of the covariance network of facial pictures. The main thing we have to do is to frame a preparation informational collection.

2D picture Ii can be spoken to as a 1D vector by connecting columns. Picture is changed into a vector of length N = mn.



Let M such vectors xi (i = 1, 2... M) of length N frame a framework of learning pictures, X. To guarantee that the primary central part portrays the heading of most extreme fluctuation, it is important to focus the framework. To begin with we decide the vector of mean qualities Ψ, and afterward subtract that vector from each picture vector.

$$\Psi = \frac{1}{M}\sum_{i=1}^{M} x_i ,$$

$$\phi_i = x_i - \Psi .$$

Averaged vectors are arranged to form a new training matrix (size N×M);

A = ($\varphi_1, \varphi_2, \ldots, \varphi_M$) .

The next step is to calculate the covariance matrix C, and find its eigenvectors ei and eigenvalues λi:

$$C = \frac{1}{M}\sum_{n=1}^{M}\varphi_n\varphi_n{}^T = AA^T,$$

$$Ce_i = \lambda_i e_i .$$

Covariance network C has measurements N×N. From that we get N eigenvalues and eigenvectors. For a picture size of 128×128, we would need to compute the network of measurements 16.384×16.384 and find 16.384 eigenvectors. It is not exceptionally successful since we needn't bother with a large portion of these vectors. Rank of covariance framework is restricted by the quantity of pictures in learning set — on the off chance that we have M pictures, we will have M–1 eigenvectors relating too non-zero eigenvalues. One of the hypotheses in direct polynomial math expresses that the eigenvectors ei and eigenvalues λi can be acquired by discovering eigenvectors and eigenvalues of grid C1 = ATA
(measurements M×M) [3]. On the off chance that νi and μi are eigenvectors and eigenvalues of lattice ATA, then:

$$A^T Av_i = \mu_i v_i$$

Multiplying both sides of this equation with **A** form the left, we get:

$$AA^T Av_i = A\mu_i v_i ,$$

$$AA^T (Av_i) = \mu_i (Av_i) ,$$

$$C(Av_i) = \mu_i (Av_i) .$$

Differentiating these conditions, we can reason that the essential M–1 eigenvectors ei and eigenvalues λi of system C are given by Aνi and μi, independently. Eigenvector related with the most essential eigenvalue reflects the most raised change, and the one related with the slightest eigenvalue, the smallest distinction. Eigenvalues decrease exponentially so that around 90% of the total contrast is contained in the underlying 5% to 10% eigenvectors. In this way, the vectors should be sorted by eigenvalues so that the foremost vector looks at to the most dumbfounding eigenvalue. These vectors are then institutionalized. They shape the new system E so that each vector ei is a segment vector. The estimations of this grid are N×D, where

D addresses the desired number of eigenvectors. It is used for projection of data system An and calculatation of yivectors of matrix.

$$Y = (y_1, \ldots, y_M):$$

$$Y = E^T A.$$

Every unique picture can be reproduced by adding mean picture Ψ to the weighted summation of all vectors ei.The last stride is the acknowledgment of appearances. Picture of the individual we need to find in preparing set is changed into a vector P, decreased by the mean esteem Ψ and anticipated with a network of eigenvectors (eigenfaces):

$$\omega = E^T (P - \Psi)$$

Arrangement is finished by deciding the separation, εi, amongst ω and each vector yi of network Y. The most well-known is the Euclidean separation, however other measures might be utilized. This paper introduces the outcomes for the Euclidean and Manhattan separate. On the off chance that An and B are two vectors of length D, the distance between them is resolved as takes after:

1. Manhattan distance:

$$d(A,B) = \sum_{i=1}^{D} |a_i - b_i|;$$

2. Euclidean distance:

$$d(A,B) = \sqrt{\sum_{i=1}^{D} (a_i - b_i)^2} = \|A - B\|.$$

If the base partition between test confront and get ready appearances is higher than an edge θ, the test face is thought to be dark, else it is known and has a place with the individual argmin. The program requires a base separation between the test picture and pictures from the readiness base. Notwithstanding the likelihood that the individual is not in the database, the face would be seen. It is accordingly important to set a breaking point that will empower us to choose if a

man is in the database. There is no condition for choosing the edge. The most widely recognized way is to first process the base partition of each photo from the preparation base from substitute pictures and place that division in a vector rast.Threshold is taken as 0.8 times of the best estimation of vector rast.

Local Binary Patterns for Face Recognition [6]

The LBP executive is one of the best performing surface descriptors and it has been by and large used as a piece of various applications. It has wound up being exceedingly discriminative and its key favorable circumstances, to be specific, its invariance to monotonic dim level changes and computational efficiency, make it sensible for asking for picture investigation assignments. For a book reference of LBP-related research, seehttp://www.ee.oulu.fi/ask about/imag/surface/.using LBP for face depiction is influenced by the way that faces can be seen as a union of littler scale plans which are all around portrayed by such manager. The LBP chairman was at first planned for surface depiction. The executive delegates a name to every pixel of a photo by thresholding the 3 x 3- neighborhood of each pixel with the middle pixel regard and considering the result as a matched number. By then, The histogram of the marks can be utilized as a surface descriptor. See Fig. 6 for a representation of the essential LBP operator. To have the capacity to manage surfaces at various scales, the LBPoperator was later stretched out to utilize neighbourhoods of various sizes. Characterizing the nearby neighbourhoods an arrangement of inspecting points evenly separated on a hover focused at the pixel to be marked allows any span and number of testing focuses. Bilinear interpolations utilized when a testing point does not fall in the focal point of a pixel. In the accompanying, the documentation ðP; Rþ will be utilized for pixel.

Neighborhoods which infers P testing centers around a float of range of R. See Fig. 7 for an instance of round neighborhoods. Another expansion to the main overseer is the meaning of purported uniform cases. An area twofold case is called uniform if the combined case contains at most two bitwise moves from 0 to 1 or the other route around when the bit configuration is viewed as circuitous. For example, the cases 00000000 (0 transitions), 01110000 (2 moves) and 11001111 (2 moves) are uniform while the examples 11001001 (4 moves) and01010011 (6 moves) are most certainly not. In the calculation of the LBP histogram, uniform examples are utilized so that the histogram has separate canister for each uniform example and all no

uniform patterns are doled out to a solitary container. Ojala et al. seen that in their experiments with surface pictures, uniform examples represent abit under 90 percent of all examples when utilizing the neighbourhood and for around 70 percent in the neighbourhood. We have found that 90.6 percent of the examples in the neighbourhood and

85.2 percent of the examples in the neighbourhood are uniform if there should arise an occurrence of pre-handled FERET facial images. We utilize the accompanying documentation for the LBP chairman: LBPu2P;R.The subscript addresses using the executive in a ðP;RÞ neighbourhood. Superscript u2 stays for using simply uniform cases.

Face Description with LBP

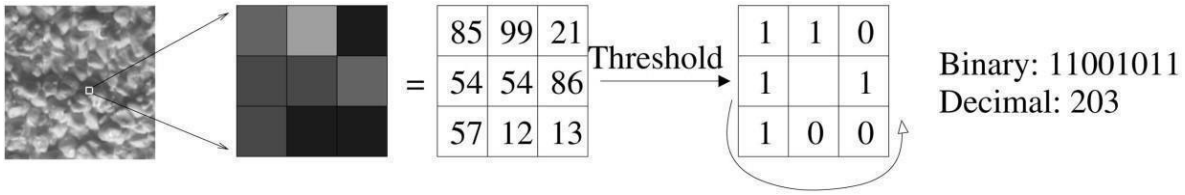


Figure 3.17 - The basic LBP operator

In this work, the LBP technique introduced in the past segment issued for face depiction. The system comprises of utilizing the texture descriptor to manufacture a few nearby depictions of the face and combining them into a worldwide portrayal. Rather than taking a stab at an all encompassing portrayal, this approach was moved by two reasons: the area incorporate based or cross breed approaches to manage confront acknowledgment have been getting interest recently, which is justifiable given the requirements of the far reaching depictions. These nearby incorporate based and cream methodologies give off an impression of being more vigorous against assortments in position or edification than complete strategies. Another clarification behind picking the close-by segment based approach is that endeavoring to make a widely inclusive depiction of a face using surface strategies is not sensible since surface descriptors tend to normal over the photo run. This is an appealing property for common surfaces, since surface depiction should as a rule be invariant to understanding or even upset of the surface and, particularly, for minimal dull surfaces, the little scale connections decide the nearness of the surface and, in this way, the largescale relations don't contain profitable information. For appearances, be that as it may, the condition is particular: holding the information about spatial relations is vital.

This reasoning prompts the crucial procedure of thiswork. The facial picture is secluded into close-by areas and surface descriptors are removed from each zone self-rulingly. The descriptors are then connected to shape an overall delineationof the face. See Fig. 7 for a case of a facial picture parceled into rectangular areas.

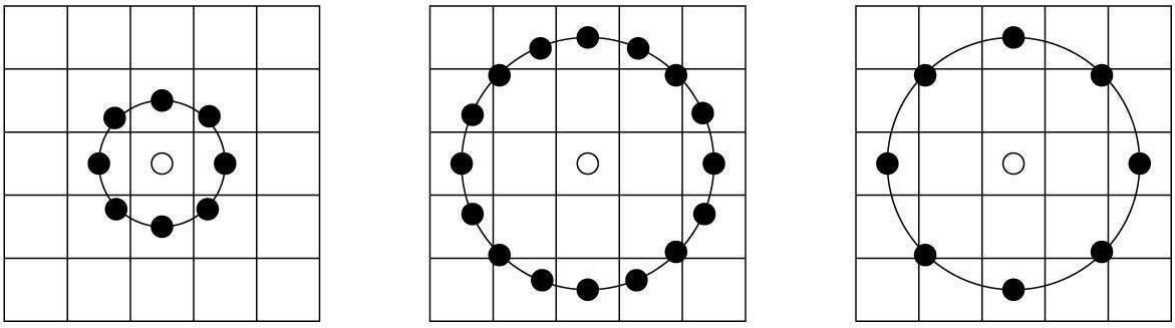


Figure 3.18 - The circular (8, 1), (16, 2), and (8, 2) neighbourhoods. The pixel values are bilinearinterpolated whenever the sampling point is not in the centre of a pixel.

This approach was moved by two reasons: the area incorporate based or cross breed approachesto manage confront acknowledgment have been getting interest recently, which is justifiable given the requirements of the far reaching depictions. These nearby incorporate based and cream methodologies give off an impression of being more vigorous against assortments in position or edification than complete strategies. Another clarification behind picking the closeby segment based approach is that endeavoring to make a widely inclusive depiction of a face using surface strategies is not sensible since surface descriptors tend to normal over the photo run. This is an appealing property for common surfaces, since surface depiction should as a rule be invariant to understanding or even upset of the surface and, particularly, for minimaldull surfaces, the little scale connections decide the nearness of the surface and, in this way, the largescalerelations don't contain profitable information. For appearances, be that as it may, the condition is particular: holding the information about spatial relations is vital. This reasoning prompts the crucial procedure of this work. The facial picture is secluded into closeby areas and surface descriptors are removed from each zone self-rulingly. The descriptorsare then connected to shape an overall delineation of the face. See Fig. 3.7 for a case of a facial picture parceled into rectangular areas.

$$\chi_w^2(\mathbf{x},\boldsymbol{\xi}) = \sum_{j,i} w_j \frac{(x_{i,j} - \xi_{i,j})^2}{x_{i,j} + \xi_{i,j}},$$

In which x and m are the normalized enhanced histograms to be compared, indices i and j refer to ith bin in histogram corresponding to the jth local region and wj is the weight for region j.

Figure 3.19 -A facial image divided into 7 x 7, 5 x 5, and 3 x 3 rectangular regions.

3.4 RELAY

The **SRD-05VDC-SL-C** relay has three high voltage terminals (NC, C, and NO) which connect to the device you want to control. The other side has three low voltage pins (Ground, Vcc, and Signal) which connect to the Arduino.

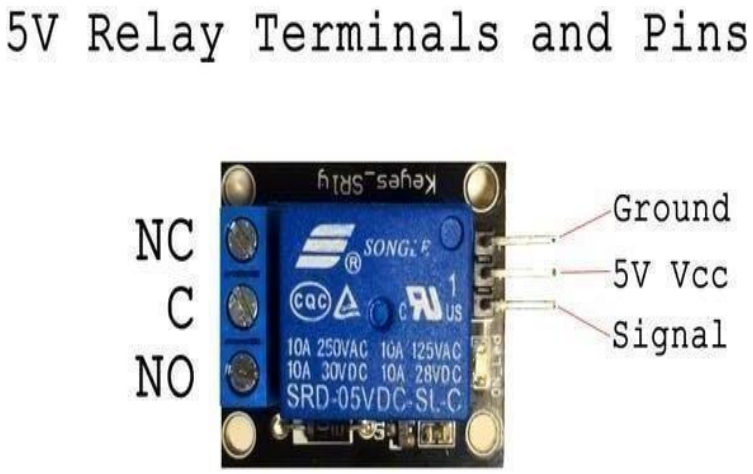


Figure 3.20 relay pin description

- NC: Normally closed 120-240V terminal
- NO: Normally open 120-240V terminal
- C: Common terminal
- Ground: Connects to the ground pin on the Arduino
- 5V Vcc: Connects the Arduino's 5V pin
- Signal: Carries the trigger signal from the Arduino that activates the relay.

Inside the relay is a 120-240V switch that's connected to an electromagnet. When the relay receives a HIGH signal at the signal pin, the electromagnet becomes charged and moves the contacts of the switch open or closed.

3.4.1 NORMALLY OPEN VS. NORMALLY CLOSED

The relay has two different types of electrical contacts inside – normally open (NO) and normally closed (NC). The one you use will depend on whether you want the 5V signal to turn the switch on or turn the switch off. The 120-240V supply current enters the relay at the common (C) terminal in both configurations. To use the normally open contacts, use the NO terminal. To use the normally closed contacts, use the NC terminal.

NORMALLY OPEN

In the normally open configuration, when the relay receives a HIGH signal the 120-240V switch closes and allows current to flow from the C terminal to the NO terminal. A LOW signal deactivates the relay and stops the current. So if you want the HIGH signal to turn ON the relay, use the normally open terminal:

NORMALLY CLOSED

In the normally closed configuration, a HIGH signal opens the switch and interrupts the 120-240V current. A LOW signal closes the switch and allows current to flow from the C terminal to the NC terminal. Therefore, if you want the HIGH signal to turn OFF the 120-240V current, use the normally closed terminal:

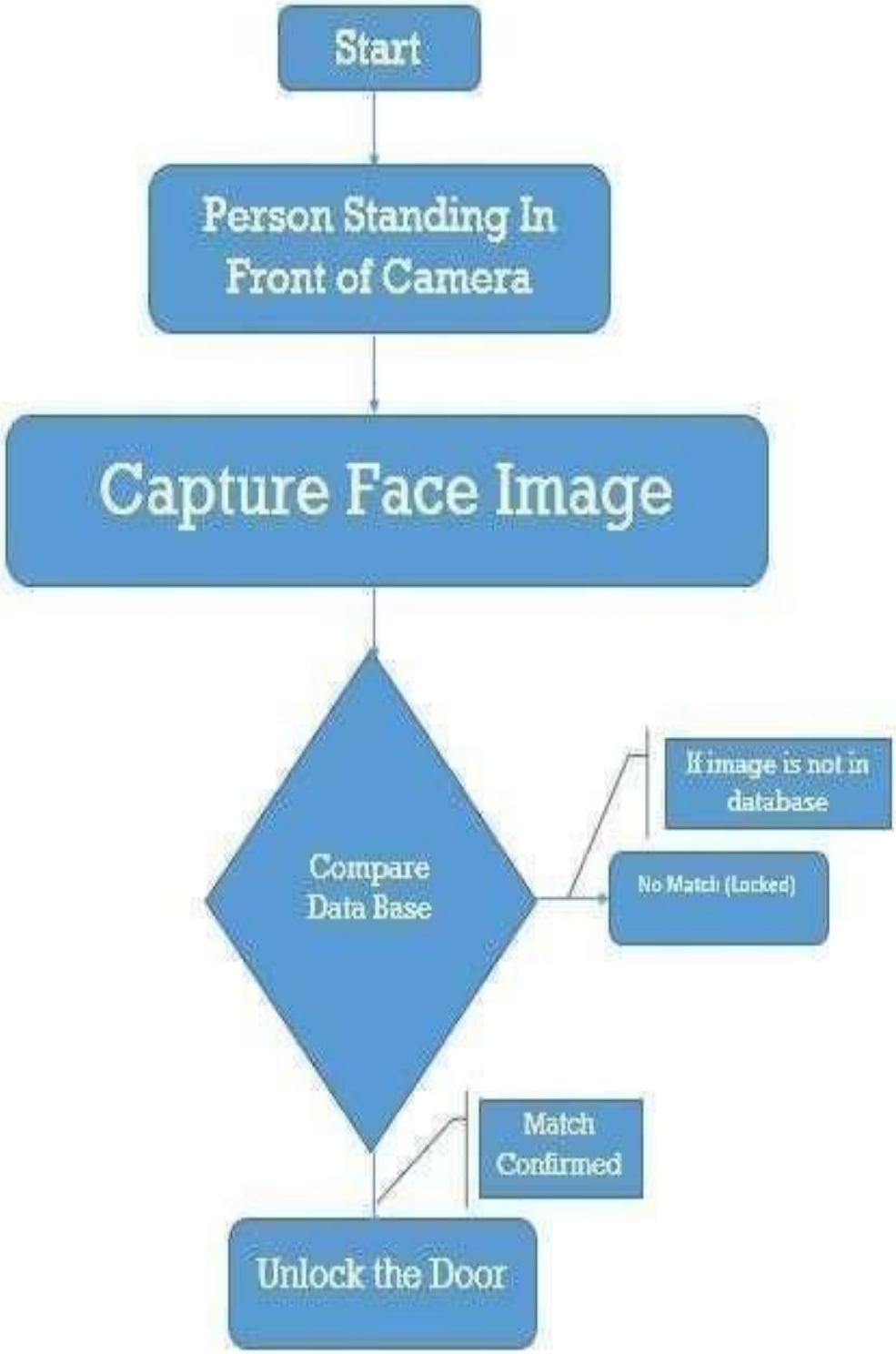
Figure 3.21: relay normally open

NORMALLY CLOSED

In the normally closed configuration, a HIGH signal opens the switch and interrupts the 120-240V current. A LOW signal closes the switch and allows current to flow from the C terminal to the NC terminal. Therefore, if you want the HIGH signal to turn OFF the 120-240V current, use the normally closed terminal:

Figure 3.22:relay normally closed

WORKING FLOW CHART



CHAPTER 4

PERFORMANCE ANALYSIS

4.1 PERFORMANCE COMPARISON BETWEEN IDENTICALLY TRAINED CASCADES WITH THREE DIFFERENT BOOSTING ALGORITHMS

All examinations were execution on the entire CMU Frontal Face Test Set of 130 greyscale pictures with 510 frontal countenances [11]. Ahit was pronounced if and just if the Euclidian separation between the focal point of a distinguished and genuine face was under 30% of the width of the real face also asthe width (i.e., size) of the recognized face was inside ±50% of the real face width. Each recognized face, which was not a hit, was considered a false alert. Hit rates are accounted for in percent, while the false cautions are determined by their supreme numbers keeping in mind the end goal to make the outcomes practically identical with related work on the CMU Frontal Face Test set. But generally noted 5000 positive frontal face designs and 3000negative examples separated by stage 0 to n-1 were utilized to prepare phase of the course classifier. The 5000 positive frontal face examples were gotten from 1000 unique face designs by arbitrary pivot about ±10 degree, irregular scaling about ±10%, irregular reflecting and irregular moving up to ±1 pixel. Each stage was prepared to dismiss about portion of the negative examples, while effectively tolerating 99.9%of the face designs. A completely prepared course comprised of 20 phases. Amid recognition, a sliding window was moved pixel by pixel over the photo at each scale. Beginning with the first scale,the elements were developed by 10% and 20%, individually (i.e., speaking to a rescale variableof 1.1 and 1.2, separately) until surpassing the span of the photo in no less than one measurement. Regularly various appearances are recognizing at close by area and scale at a genuine face area. Hence, various adjacent discovery results were consolidated. Recipient Operating Curves (Rocs) were developed by changing the required number of identified countenances per genuine face before converging into a solitary location result. Amid experimentation just a single parameter was changed at once. The best method of a parameter found in an examination was utilized for the resulting tests.

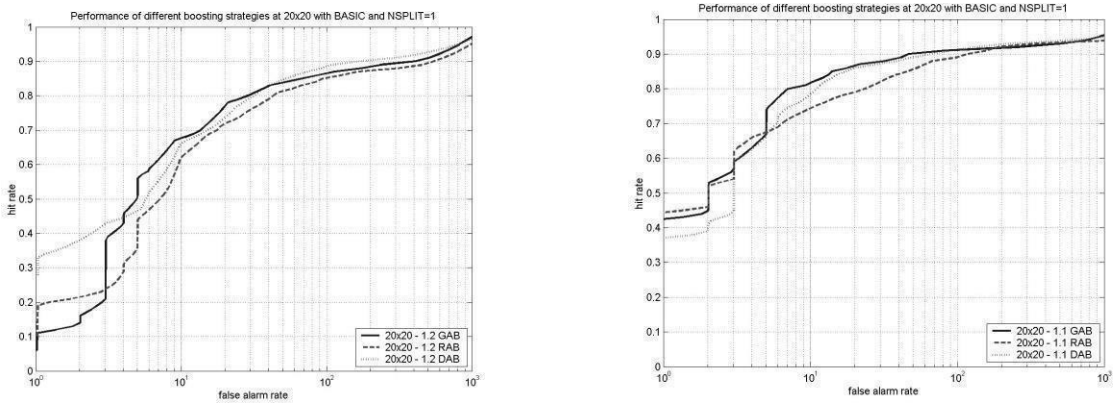


Figure 4.1 - Performance comparison between identically trained cascades with three different boosting algorithms. Only thebaic feature set and stumps as weak classifiers (nsplit=1) were used.

4.2 PRTOTYPE FOR FACE RECOGNITION AND DOOR UNLOCK

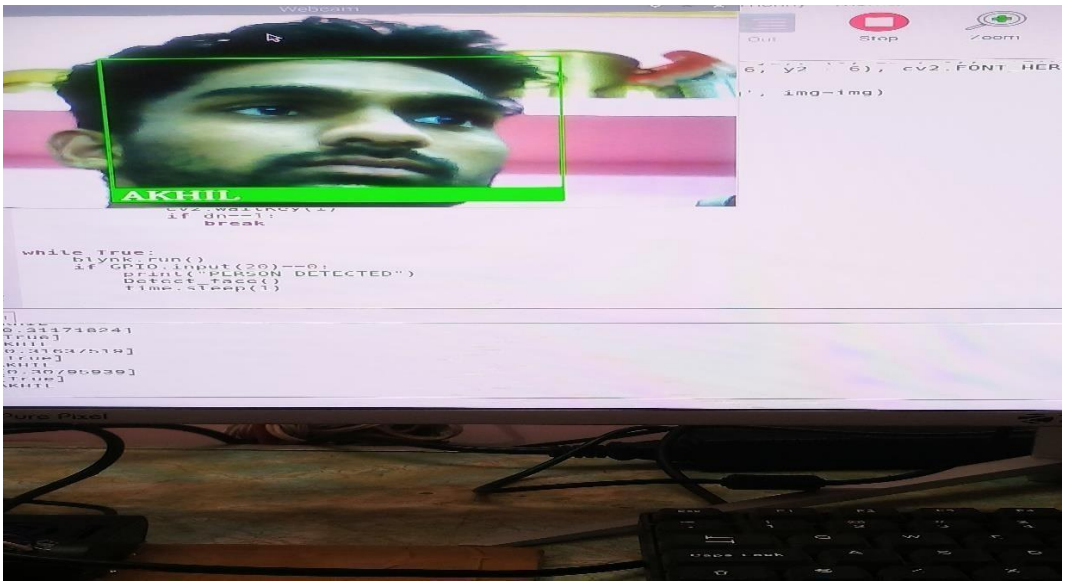


Figure 4.2 showing the prototype model for facial recognition

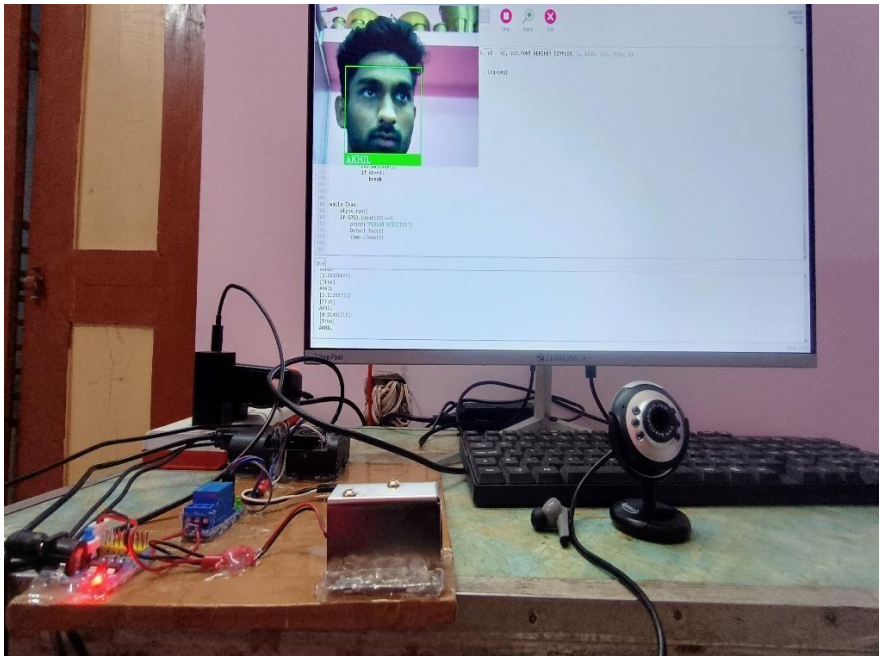


Figure 4.3 figure showing the detection of known person

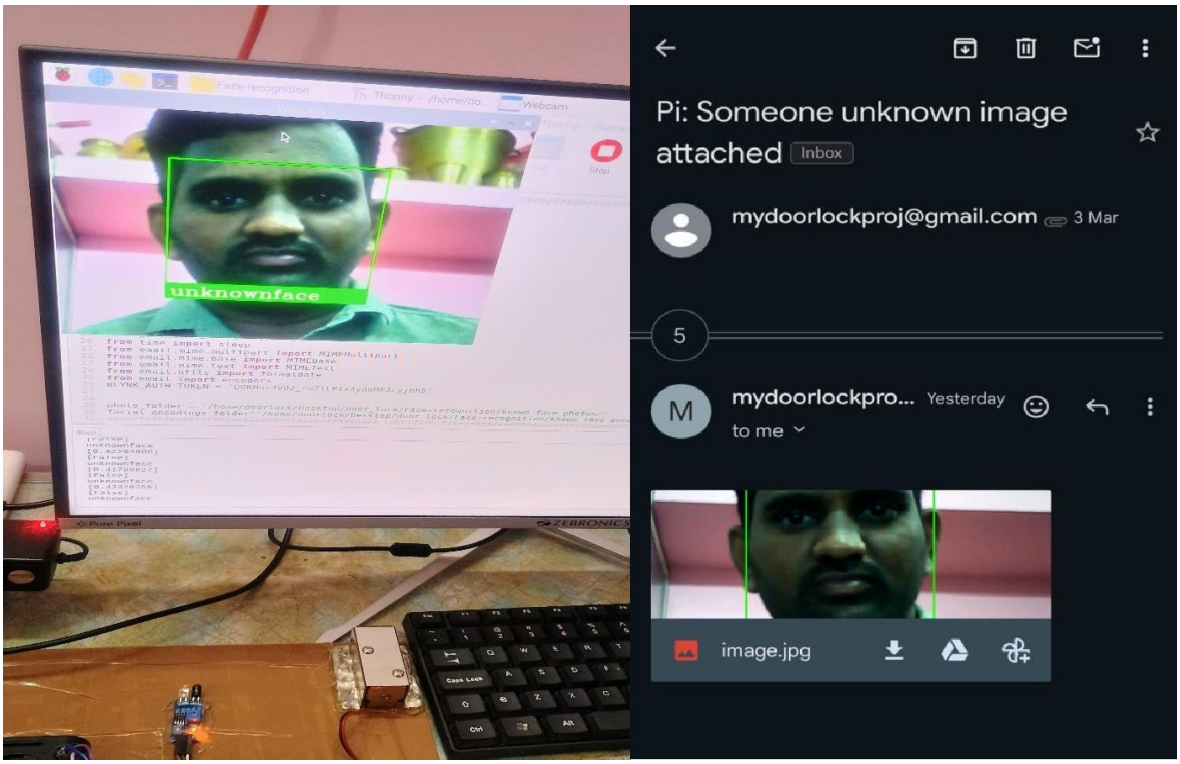


Figure 4.4 figure showing the unknown person and email sent

CHAPTER 5

CONCLUSION

PCA based facial identification framework utilizes the Raspberry Pi 3 Model Development stage construct around a BCM2837 System-with respect to Chip wearing an ARM Cortex- A53 processor. Picture catch gadget utilized here is Pi Camera Module v2.

The recovery of pictures containing human countenances requires location of human faces in such pictures. We executed the Haar Cascade Classifiers technique that outputs the entire picture base on Haar Classifiers and distinguishes if a face is available in the picture. We took around 30 pictures from our Picamera and got around 83% of precision. The misses as a rule were because of revolution of picture or awful lighting. Our present execution is restricted to the location of frontal human appearances. A conceivable and intriguing augmentation is extend the format coordinating procedure to incorporate sided-see confronts too.

This system is used door lock access for Residential and Commercial Purposes. Here we have designed a highly secured door locking system by using Raspberry pi.

REFERENCES

[1] Bradski, Gary, and Adrian Kaehler. Learning OpenCV: Computer vision with the OpenCVlibrary. " O'Reilly Media, Inc.", 2008: pp. 1 – 8

[2] Pulli, Kari, et al. "Real-time computer vision with OpenCV." Communications of the ACM 55.6 (2012): pp. 63-66.

[3] Wilson, Phillip Ian, and John Fernandez. "Facial feature detection using Haar classifiers." Journal of Computing Sciences in Colleges 21.4 (2006): pp. 127-133.

[4] Van Der Walt, Stefan, S. Chris Colbert, and Gael Varoquaux. "The NumPy array: a structure for efficient numerical computation." Computing in Science & Engineering 13.2 (2011): pp. 22-30.

[5] Slavković, Marijeta, and Dubravka Jevtić. "Face recognition using eigenface approach." Serbian Journal of Electrical Engineering 9.1 (2012): pp. 121-124.

[6] Ahonen, Timo, Abdenour Hadid, and Matti Pietikainen. "Face description with local binary patterns: Application to face recognition." IEEE transactions on pattern analysis and machine intelligence 28.12 (2006): pp. 2037-2038.

[7] https://www.howtoforge.com/tutorial/howto-install-raspbian-on-raspberry-pi/

[8] http://www.pyimagesearch.com/2016/04/18/install-guide-raspberry-pi-3-raspbian-jessie-opencv-3/

[9] Soo, Sander. "Object detection using Haar-cascade Classifier." (2014).

[10] Lienhart, Rainer, Alexander Kuranov, and Vadim Pisarevsky. "Empirical analysis of detection cascades of boosted classifiers for rapid object detection." Joint Pattern Recognition Symposium. Springer Berlin Heidelberg, 2003: pp. 4-5

[11] H. Rowley, S. Baluja, and T. Kanade. Neural network-basedface detection. In IEEE Patt. Anal. Mach. Intell., Vol. 20, 1998: pp.22-38.

APPENDIX

PYTHON SOURCE CODES

Code for enrollment

```
import tkinter as tk
from tkinter import ttk
from tkinter import messagebox as
mess import tkinter.simpledialog as tsd
import cv2,os import csv import numpy
as np from PIL import Image import
pandas as pd import datetime import
RPi.GPIO as GPIO
import time
import face_recognition,cv2,pickle
import time
photo_folder ='/home/doorlock/Desktop/door_lock/Face-recognition/known face photos/'
facial_encodings_folder='/home/doorlock/Desktop/door_lock/Face-recognition/known
faceencodings/'
cap = cv2.VideoCapture(0)
known_face_encodings=[]
known_face_names=[]
TOLERANCE = 0.4 def
load_facial_encodings_and_names_from_memory():
    for filename in os.listdir(facial_encodings_folder):
        known_face_names.append(filename[:-4])
        with open(facial_encodings_folder+filename, 'rb') as fp:
                known_face_encodings.append(pickle.load(fp)[0])
def encoding_of_enrolled_person(name,image):
        enroll_encoding=[]
 enroll_encoding.append(face_recognition.face_encodings(face_recognition.load_imag
e_file(image))[0])
```

```
        with open(facial_encodings_folder+name+'.txt','wb') as fp:
                pickle.dump(enroll_encoding,fp)
        f.close

def enroll_via_camera(name):
        while True:
                ret,frame=cap.read()
                cv2.imshow('Enrolling new attendee',frame)
        k=cv2.waitKey(1)

                if k & 0xFF==ord('y'):
        cv2.imwrite(photo_folder+name+'.jpg',frame)
                        time.sleep(2)
                        encoding_of_enrolled_person(name,photo_folder+name+'.jpg')
                cv2.destroyAllWindows()
                        break
        if k&0xFF==ord('q'):
        print('quitting')
        cv2.destroyAllWindows()
                        break
        cap.release()


def enroll():
   name=txt2.get()
   enroll_via_camera(name)


def tick():
   time_string =
time.strftime('%H:%M:%S')
clock.config(text=time_string)
clock.after(200,tick)
```

```
    txt.delete(0, 'end')
    res = "1)Take Images >>> 2)Save
Profile" message1.configure(text=res)
def refresh():
with open('Attendance.csv') as f:
        reader = csv.DictReader(f, delimiter=',')

        for item in tv.get_children():
            tv.delete(item)
for row in reader:
            Name = row['Name']
            Date = row['Date']
            Time = row['Time']
            tv.insert("", 0, values=(Name, Date,Time))
            window = tk.Tk()
            window.geometry("1280x720")
            window.resizable(True,False)
window.title("doorlocl System")
window.configure(background='#2d420a')

frame1 = tk.Frame(window, bg="#c79cff")
frame1.place(relx=0.11, rely=0.17, relwidth=0.39, relheight=0.80)
frame2 = tk.Frame(window, bg="#c79cff")
frame2.place(relx=0.51, rely=0.17, relwidth=0.38, relheight=0.80)
message3 = tk.Label(window, text="Face Recognition Based Door lock system"
,fg="white",bg="#2d420a" ,width=55 ,height=1,font=('comic', 26, ' bold '))
message3.place(x=0, y=10)
```

```
head2 = tk.Label(frame2, text="              For New Registrations                  ",
fg="black",bg="#00fcca" ,font=('comic', 17, ' bold ') )
head2.grid(row=0,column=0)

lbl2 = tk.Label(frame2, text="Enter Name",width=20 ,fg="black" ,bg="#c79cff" ,font=('comic',
17, ' bold '))

lbl2.place(x=80, y=140)

txt2 = tk.Entry(frame2,width=32 ,fg="black",font=('comic', 15, ' bold ') )
txt2.place(x=30, y=173)
message1 = tk.Label(frame2, text="Take Images" ,bg="#c79cff" ,fg="black"  ,width=39
,height=1, activebackground = "#3ffc00" ,font=('comic', 15, ' bold '))
message1.place(x=6, y=230)
message = tk.Label(frame2, text="" ,bg="#c79cff" ,fg="black"

,width=39,height=1, activebackground = "#3ffc00" ,font=('comic', 16, ' bold '))
message.place(x=7, y=450) res=0
#exists = os.path.isfile("StudentDetails\StudentDetails.csv")

takeImg = tk.Button(frame2, text="Enroll", command=enroll ,fg="white"  ,bg="#6d00fc"
,width=25  ,height=1, activebackground = "white" ,font=('comic', 15, ' bold '))
takeImg.place(x=50, y=300)
quitWindow = tk.Button(frame1, text="Quit",
command=window.destroy ,fg="black"

,bg="#eb4600" ,width=25 ,height=1,

activebackground = "white" ,font=('comic', 15, ' bold')) quitWindow.place(x=50, y=350)

window.mainloop()
```

Main code:

```
import cv2,os import
csv import numpy as np
from PIL import Image
import pandas as pd
import datetime import
RPi.GPIO as GPIO
import time
import face_recognition,cv2,pickle
import time import smtplib,ssl
import BlynkLib import RPi.GPIO
asGPIO from BlynkTimer import
BlynkTimer
from picamera import PiCamera
from time import sleep
from email.mime.multipart import
MIMEMultipart  from email.mime.base import
MIMEBase from email.mime.text import
MIMEText from email.utils import formatdate
from email import encoders
BLYNK_AUTH_TOKEN = 'DORHur4V0J_roZlFPix4ydGMFJcyjbh5'

photo_folder = '/home/uasir/Desktop/door_lock/Face-recognition/known face photos/'
facial_encodings_folder='/home/uasir/Desktop/door_lock/Face-recognition/known face
encodings/' RELAY = 21
GPIO.setwarnings(False)
GPIO.setmode(GPIO.BCM)
GPIO.setup(20, GPIO.IN, pull_up_down=GPIO.PUD_UP)
```

```
GPIO.setup(RELAY, GPIO.OUT)
GPIO.output(RELAY,GPIO.HIGH)
ad=0
cd=0
dn=0
cap  =cv2VideoCapture(0)
known_face_encodings=[]
known_face_names=[]
TOLERANCE = 0.4
blynk = BlynkLib.Blynk(BLYNK_AUTH_TOKEN)

# Led control through V0 virtual
pin@blynk.on("V0") def
v0_write_handler(value):
 #global led_switch if
if int(value[0]) != 0:
      GPIO.output(RELAY, GPIO.HIGH)
print('RELAY HIGH')         else:
      GPIO.output(RELAY,
GPIO.LOW)print('RELAY LOW')

#function to sync the data from virtual pins
@blynk.on("connected")
def blynk_connected():
   print("Raspberry Pi Connected to New Blynk")

def load_facial_encodings_and_names_from_memory():
        for filename in os.listdir(facial_encodings_folder):
        known_face_names.append(filename[:-4])

        with open(facial_encodings_folder+filename, 'rb') as fp:
                known_face_encodings.append(pickle.load(fp)[0])
```

```
def send_an_email():
toaddr = 'karanamakhil9666@gmail.com' # To id
me = 'mydoorlockproj@gmail.com'          # your id
   subject = "Pi: Someone unknown image attached"              # Subject

msg                          =
MIMEMultipart()
msg['Subject'] = subject
msg['From'] = me
msg['To'] = toaddr
msg.preamble      =      "test      "
msg.attach(MIMEText(text))

part = MIMEBase('application', "octet-stream")
part.set_payload(open("saved_img.jpg", "rb").read())
encoders.encode_base64(part)
part.add_header('Content-Disposition', 'attachment; filename="image.jpg"') # File nameand
format name
   msg.attach(part)

try:
      s = smtplib.SMTP('smtp.gmail.com', 587)  # Protocol
       s.ehlo()
      s.starttls()
      s.ehlo()
      s.login(user = 'mydoorlockproj@gmail.com', password = 'igak bjig wdei fhbh') # User id &
password
      #s.send_message(msg)
      s.sendmail(me, toaddr, msg.as_string())
      s.quit()
      # except:
       #  print ("Error: unable to send email")
```

```
except smtplib.SMTPException as error:
        print ("Error")             #
        Exception
def encoding_of_enrolled_person(name,image):
         enroll_encoding=[]
 enroll_encoding.append(face_recognition.face_encodings(face_recognition.load_imag
e_file(image))[0])
                f=open(facial_encodings_folder+name+'.txt','w+')
         with open(facial_encodings_folder+name+'.txt','wb') as fp:
                  pickle.dump(enroll_encoding,fp)
         f.close
def Detect_face():
 load_facial_encodings_and_names_from_memory()
 ad=0
 cd=0
dn=0
 while True:
      success, img =cap.read()
 # img = captureScreen()
 if img is None:
        print('IMG not read')
 else:
         imgS = cv2.resize(img, (0, 0), None, 0.25, 0.25)
 imgS = cv2.cvtColor(imgS, cv2.COLOR_BGR2RGB)
 facesCurFrame = face_recognition.face_locations(imgS)
 encodesCurFrame = face_recognition.face_encodings(imgS, facesCurFrame)
 for encodeFace, faceLoc in zip(encodesCurFrame, facesCurFrame):
            matches = face_recognition.compare_faces(known_face_encodings,
encodeFace,TOLERANCE)
```

```
            faceDis = face_recognition.face_distance(known_face_encodings,
encodeFace)
                    print(faceDis)
                     print(matches)
                    matchIndex = np.argmin(faceDis)
 if matches[matchIndex]:
 name =known_face_names[matchIndex].upper()
print(name)
y1, x2, y2, x1 = faceLoc
 y1, x2, y2, x1 = y1 * 4, x2 * 4, y2 * 4, x1 * 4
cv2.rectangle(img, (x1, y1), (x2, y2), (0, 255, 0), 2)
 cv2.rectangle(img, (x1, y2 - 35), (x2, y2), (0, 255, 0), cv2.FILLED)
  cv2.putText(img, name, (x1 + 6, y2 - 6), cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255,
  255), 2)
 ad=ad+1
if ad==5:
GPIO.output(RELAY,GPIO.LOW)
prevTime = time.time()
doorUnlock = True
time.sleep(5)
 GPIO.output(RELAY,GPIO.HIGH)
print("door unlock")
time.sleep(2)

 print('quitting')
cv2.destroyAllWindows(
)dn=1
 break
 #break
```

```
else:
 print("unknownface")
y1, x2, y2, x1 =
faceLoc
 y1, x2, y2, x1 = y1 * 4, x2 * 4, y2 * 4, x1 * 4
cv2.rectangle(img, (x1, y1), (x2, y2), (0, 255, 0), 2)
 cv2.rectangle(img, (x1, y2 - 35), (x2, y2), (0, 255, 0), cv2.FILLED)
 cv2.putText(img, "unknownface", (x1 + 6, y2 - 6),
 cv2.FONT_HERSHEY_COMPLEX, 1, (255, 255, 255), 2)
 cd=cd+1
 if cd==10:
  cv2.imwrite(filename='saved_img.jpg', img=img)
time.sleep(3)
send_an_email()
time.sleep(2)
cv2.destroyAllWindows()
dn=1
 break

 k=cv2.waitKey(1)
if k&
0xFF==ord('q'):
print('quitting')
cv2.destroyAllWindows()
cap.release()
 break
cv2.imshow('Webcam', img)
cv2.waitKey(1)
 if
dn==1:
break
```

```
while True:
 blynk.run() if GPIO.input(20) ==GPIO.LOW:
 print("Button was pushed!")
Detect_face()
time.sleep(1)
```

Face Recognition Systems for Smart Door Access Using Raspberry pi

G.Siva Kumari
Assistant Professor,
Electronics and Communication Engineering,
Dadi Institute Of Engineering & Technology,
Anakapalle,India

K.Akhil
Student,
Electronics and Communication Engineering
Dadi Institute of Engineering & Technology,
Anakapalle,India

B.Bhanu Lokesh
Student,
Electronics and Communication Engineering
Dadi Institute Of Engineering & Technology,
Anakapalle,India

N.Sai Kumar Goud
Student,
Electronics and Communication Engineering,
Dadi Institute Of Engineering &Technology
Anakapalle,India

***Abstract*—This project aims to enhance security using facial recognition technology. By comparing captured facial images with stored ones, the system unlocks a door for authorized users. Facial recognition is increasingly popular for its effectiveness in identifying intruders and restricting unauthorized access, outperforming other methods like fingerprints or passwords. However, achieving quick and accurate recognition poses a challenge. This project addresses this by simplifying images, storing them, and rapidly comparing new images for matches. When a match is found, a controller unlocks a door. If an unknown face is detected, it alerts designated individuals for a decision on granting access. This project demonstrates how facial recognition tech can swiftly enhance security measures.**

Keywords - Facial Recognition, Security Systems, Biometric Authentication, Access Control

I. INTRODUCTION

In the era of expanding interest in smart home systems driven by the Internet of Things (IoT), security remains a paramount concern. Traditionally, security experts have recommended approaches like biometrics and passwords to fortify home security. However, as technology evolves, the focus has shifted markedly towards face recognition systems, seen as a pivotal advancement in safeguarding homes.

The proposed project centers on leveraging a Raspberry Pi microcontroller board, a Pi camera module for face recognition, and a programmable stepper motor to enhance home security. The foundational step involves installing a suitable Linux-based operating system onto the Raspberry Pi microcontroller board. The core mechanism for unlocking the door involves positioning a stepper motor at the door latch, programmed to respond when the system authenticates a person in front of the camera. The innovation lies in the integration of image processing technology, utilizing the Pi camera module for facial recognition. This module interfaces with the Raspberry Pi, facilitating the storage of various facial profiles within the database. When an individual approaches the home, they stand before the camera. The system employs image recognition to compare the captured face with those stored in the database, granting automatic door access upon a match. In cases of mismatch, an alert message promptly notifies the homeowner.

This project represents a convergence of IoT, image processing, and Raspberry Pi technology, offering a sophisticated yet user-friendly means to reinforce home security through facial recognition, ensuring seamless access while maintaining robust protective measures.

II. LITERATURE SURVEY

The literature survey explores several facets of face recognition systems implemented within IoT-based security setups. Kulkarni, Bagul, and Dukare (2017) introduced a system adaptable for various connections, such as cascade, parallel, or series, offering flexibility for system expansion. However, its limitation lies in the absence of a secondary unlocking method if facial recognition fails for authorized users. The system's dual functionality in online and offline modes adds versatility, utilizing internet connectivity based on operational requirements.

Vamsi, Sai, and Vijayalakshmi (2019) highlighted the efficiency of the LBPH algorithm for face recognition due to its representation of local features, especially in controlled environments. Yet, its sensitivity to scale requires preprocessing for normalization. While robust against grayscale transformations, LBPH's performance diminishes with variations in pose and illumination.

Gsponer (2018) aimed to create a cost-effective security system utilizing facial recognition, emphasizing three core elements: data gathering, machine learning, and facial recognition. Similar to the previous systems, the absence of an alternative unlocking method poses a limitation.

Deshmukh, Nakrani, Bhuyar, and Shinde (2019) utilized Haar classifiers for face detection, highlighting its advantage in calculating features rapidly due to integral image usage. However, while it offers swift detection, Haar classifiers tend to sacrifice accuracy compared to techniques like CNN.

In examining the literature and proposed systems, a notable aspect absent from both is the incorporation of email

alerts and IoT-driven responses when an unauthorized user attempts access. The reviewed systems primarily focus on facial recognition as the primary authentication method for door unlocking, but they lack provisions for immediate alerts or actions when unauthorized entry is detected.

The proposed system, in contrast, introduces a novel layer of security enhancement. Apart from facial recognition-based door unlocking, it incorporates an alert mechanism via email notification to the homeowner or administrator when an unrecognized individual approaches. Furthermore, the system harnesses IoT capabilities to enable responsive actions upon unauthorized access attempts. This could involve triggering alarms, capturing images of the intruder, or logging the event for future reference, enhancing the system's security by providing real-time alerts and potential deterrents against unauthorized entry. This addition elevates the proposed system beyond the scope of the reviewed literature, addressing a crucial aspect of security by integrating proactive measures in response to unauthorized access attempts, thereby enhancing the overall robustness of the IoT-based facial recognition security system.

III. PROPOSED SYSTEM

The proposed system shown in fig1 centers on utilizing Raspberry Pi in conjunction with a USB camera to establish a comprehensive security setup. Upon program activation, the Pi camera assumes the role of a surveillance system. When the authorized owner stands before the camera, facial recognition triggers the unlocking of the door, granting access. However, if the facial features do not match those stored in the system's database, an alert is promptly dispatched to the administrator. Simultaneously, an image of the individual attempting access is captured and stored in a designated folder within the Raspberry Pi. This feature serves as a security measure, enabling the system to document unauthorized access attempts for subsequent review and analysis.

To further enhance user convenience and remote access control, the system integrates a door unlock app. This app empowers the user to remotely unlock the door for authorized individuals even when they are not physically present at home. This functionality adds a layer of flexibility and control, allowing homeowners to manage door access conveniently from a distance, ensuring secure and controlled entry even in their absence.

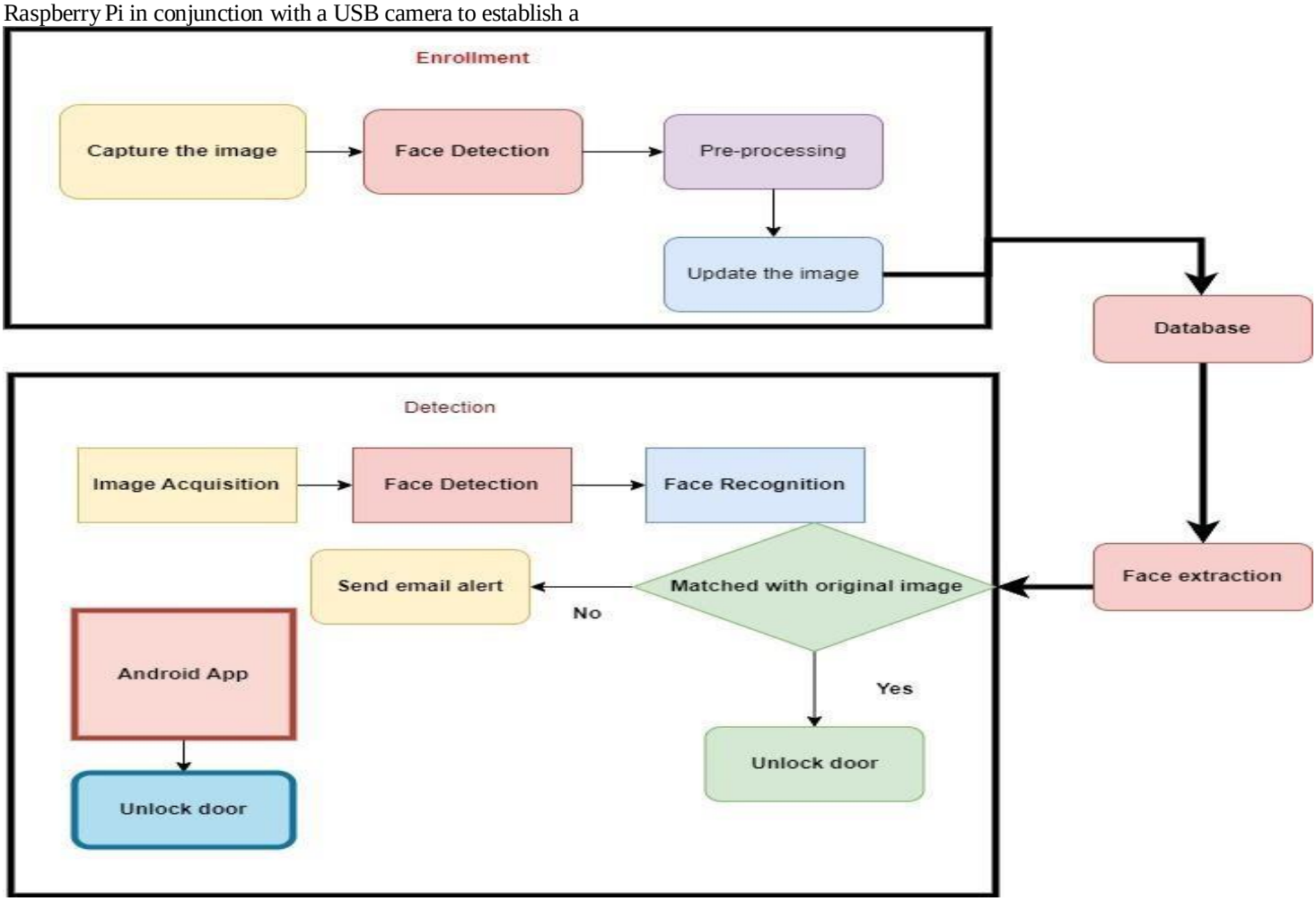


Fig. 1 Proposed system architecture

IV. SYSTEM ARCHITECTURE

The system architecture is explained in following sections.

A. Enrollment :

Capture and Pre-Process Images: The system utilizes a camera module to capture images of individuals. These images undergo pre-processing, enhancing their suitability for face detection.

Face Detection: The system employs accurate face detection algorithms, specifically a Convolutional Neural Network (CNN), to identify faces. If a face isn't detected correctly, the system restarts for accurate processing.

Pre-Processing: Prior to feature extraction using PCA (Principal Component Analysis), the face images undergo additional pre-processing to enhance recognition rates.

Update Database: Processed images of authorized users are stored in a designated folder for recognition during access attempts.

B. Face Detection and Recognition:

CNN for Detection and Recognition: CNN, known for high accuracy, detects the homeowner's face. The system then matches it against stored face data in the database. Upon a successful match, the Raspberry Pi signals the solenoid lock to unlock the door.

C. System Implementation:

Hardware Setup: The system utilizes a modified Pi Camera module connected to a Raspberry Pi 4 Model B via WLAN for real-time face recognition.

Objective: Implement a face recognition system based on CNN (Open Face) within a real-time embedded system like Raspberry Pi, showcasing practical use through Automated Door Access.

Enhancements: Future system improvements include implementing anti-spoofing measures like eye-blink detection and sending an intruder alert email containing a visitor's picture.

Convolutional Neural Networks (CNNs):

CNN Operations: CNNs perform four main operations: convolution, non-linearity (Re LU), pooling or subsampling, and classification. These operations facilitate efficient feature extraction and recognition.

The proposed system showcases a comprehensive approach to face recognition, emphasizing real-time application and security enhancements through CNN-based detection, with future considerations for anti-spoofing measures and intruder alerts.

V. REQUIREMENTS

Raspberry Pi 4 Model B:

The selection of the Raspberry Pi 4 Model B as the core hardware for this project was meticulous, considering its exceptional features and versatility. In an extensive evaluation of microcontrollers, the Raspberry Pi stood out for its robust processing capabilities, affordability, and adaptability across diverse programming environments. Running on the Linux OS, it offers access to an extensive array of compatible libraries and applications.

Fig. 2 Raspberry-pi 4

Key Features of Raspberry Pi 4 Model B:

Connectivity: Featuring an Ethernet port for network connectivity within the same subnet, it allows device access and management. With four USB ports, it facilitates connections for peripherals like keyboards, mice, cameras, and other USB-compatible devices.

Interface and GPIO Pins: Equipped with an HDMI port providing an interface to access the installed operating system, it aids initial device setup. The GPIO (General Purpose Input/Output) pins, divided into 3V and 5V groups, enable signal reception and transmission, crucial for interfacing with external components.

Software and OS Compatibility: Upon acquisition, the Raspberry Pi doesn’t come with a pre-installed operating system, which can be downloaded from the Raspberry website and transferred onto an SD card for installation. Supported by Debian and Arch Linux ARM distributions, it primarily uses Python as its main programming language, supplemented by BBC BASIC, C, and Perl.

USB Camera Module: The USB Camera module seamlessly interfaces with the Raspberry Pi through its USB ports. Offering compatibility with various USB cameras, it enhances flexibility in choosing camera models based on project requirements and supports different image and video resolutions efficiently.

Fig. 3 USB camera

Additional Components: Apart from the USB Camera module, the system integrates a Solenoid Lock, offering power-on unlocking and locking modes, ensuring secure access based on the solenoid's power status. Additionally, a Buck Converter efficiently steps down voltage from the input supply to the load, ensuring optimal power management within the system.

Fig. 4 Solenoid lock

Each hardware component, from the Raspberry Pi as the central controller to the USB camera module, solenoid lock, and buck converter, plays a vital role in ensuring functionality, connectivity, and security within the system.

VI. RESULTS

CONCLUSION

The Raspberry Pi 4 Model B serves as the linchpin of this project, meticulously chosen for its robust processing prowess, adaptability across programming environments, and extensive connectivity options. Its integration with a USB Camera module offers versatile image capture, while GPIO pins facilitate seamless interfacing with peripherals. Complemented by a Solenoid Lock and a Buck Converter, this hardware ensemble ensures security, efficient power management, and seamless functionality. Each component's synergy within the system, from the Raspberry Pi to the USB Camera, solidifies the project's foundation, enabling a robust and adaptable platform for the envisioned face recognition system.

REFERENCES

[1] S. Kulkarni, M. Bagul, and A. Dukare, " Confront acknowledgment framework utilizing IoT," IJARCET Distributions, 2017.

[2] T.K. Vamsi, K.C. Sai, and V. M, " Confront acknowledgment based entryway opening utilizing Raspberry Pi," IJARIIT Distributions, Feb. 2019.

[3] D. Gsponer, "Building a Raspberry Pi security framework utilizing facial acknowledgment," Haaga-Helie distributions, 2018.

[4] K.M. Mande and N. Bhansali, " Shrewd entryway utilizing confront acknowledgment," IRJET Distributions, May 2018.

[5] A.D. Deshmukh, M.G. Nakrani, D.L. Bhuyar, and U.B. Shinde, "Confront acknowledgment utilizing OpenCV on IoT for savvy entryway," Elsevier SSN distributions, 2019.

GIS SCIENCE JOURNAL

An UGC-CARE Approved Group II Journal

ISSN NO : 1869-9391 / Website : www.gisscience.net /
Email : editorgsjournal@gmail.com

Certificate of Publication

Paper ID : GSJ/12271

This is to certify that the paper titled

Face Recognition Systems for Smart Door Access Using Raspberry pi

Authored by

G.Siva Kumari

From

Dadi Institute of Engineering & Technology, Anakapalle, India.

Has been published in

GIS SCIENCE JOURNAL Volume 11, Issue 3, March 2024.

UGC APPROVED JOURNAL

DOI: 20.18001.GSJ

M Palaniswami
Editor-in-chief
GISSCIENCE

6.1 IMPACT FACTOR

GIS SCIENCE JOURNAL

An UGC-CARE Approved Group II Journal

ISSN NO : 1869-9391 / Website : www.gisscience.net /
Email : editorgsjournal@gmail.com

Certificate of Publication

Paper ID : GSJ/12271

This is to certify that the paper titled

Face Recognition Systems for Smart Door Access Using Raspberry pi

Authored by

K.Akhil

From

Dadi Institute of Engineering & Technology, Anakapalle, India.

Has been published in

GIS SCIENCE JOURNAL Volume 11, Issue 3, March 2024.

UGC APPROVED JOURNAL

DOI: 20.18001.GSJ

crossref member

CROSSREF.ORG THE CITATION LINKING BACKBONE

M Palaniswami
Editor-in-chief
GISSCIENCE

6.1 IMPACT FACTOR

GIS SCIENCE JOURNAL

An UGC-CARE Approved Group II Journal

ISSN NO : 1869-9391 / Website : www.gisscience.net /
Email : editorgsjournal@gmail.com

Certificate of Publication

Paper ID : GSJ/12271

This is to certify that the paper titled

Face Recognition Systems for Smart Door Access Using Raspberry pi

Authored by

B.Bhanu Lokesh

From

Dadi Institute of Engineering & Technology, Anakapalle, India.

Has been published in

GIS SCIENCE JOURNAL Volume 11, Issue 3, March 2024.

UGC APPROVED JOURNAL

DOI: 20.18001.GSJ

crossref member

CROSSREF.ORG THE CITATION LINKING BACKBONE

M Palaniswami
Editor-in-chief
GISSCIENCE

6.1 IMPACT FACTOR

GIS SCIENCE JOURNAL

An UGC-CARE Approved Group II Journal

ISSN NO : 1869-9391 / Website : www.gisscience.net /
Email : editorgsjournal@gmail.com

Certificate of Publication

Paper ID : GSJ/12271

This is to certify that the paper titled

Face Recognition Systems for Smart Door Access Using Raspberry pi

Authored by

N.Sai Kumar Goud

From

Dadi Institute of Engineering & Technology, Anakapalle, India.

Has been published in

GIS SCIENCE JOURNAL Volume 11, Issue 3, March 2024.

UGC APPROVED JOURNAL

DOI: 20.18001.GSJ

M Palaniswami
Editor-in-chief
GISSCIENCE

6.1 IMPACT FACTOR