Ministry of Education and Science of Ukraine
Ternopil Ivan Puluj National Technical University

Faculty Of Computer Information Systems And Software Engineering
(full name of faculty)
Computer Systems And Networks Department
(full name of department)

QUALIFYING PAPER

For the degree of

bachelor
(degree name)

topic: Smart door-Bell with real-time face recognition

Розумні двері з розпізнаванням обличчя в режимі реального часу

Submitted by: fourth year student _____, group ICI-43

specialty 123 Computer Engineering
(code and name of specialty)

	_____ (signature)	Aboulfadel Mohamed (surname and initials)
Supervisor	_____ (signature)	Andriy Lutskiv. (surname and initials)
Standards verified by	_____ (signature)	Stadnyk N.B. (surname and initials)
Head of Department	_____ (signature)	Osukhivska H.M. (surname and initials)
Reviewer	_____ (signature)	_____ (surname and initials)

Ternopil
2021

Ministry of Education and Science of Ukraine
Ternopil Ivan Puluj National Technical University

Faculty Faculty Of Computer Information Systems And Software Engineering
(full name of faculty)

Department Computer Systems And Networks Department
(full name of department)

APPROVED BY
Head of Department
_____ Osukhivska H.M.
(signature) (surname and initials)
« » 20___

ASSIGNMENT

For QUALIFYING PAPER

for the degree of *bachelor*
(degree name)

specialty *123 Computer Engineering*
(code and name of the specialty)

student *Aboulfadel Mohamed*
(surname, name, patronymic)

1. Paper topic Smart door-bell with real-time face recognition

Paper supervisor Lutskiv Andriy, PhD, Assoc. Prof.
(surname, name, patronymic, scientific degree, academic rank)

Approved by university order as of «___» _____ 20___ № _____

2. Student's paper submission deadline 29/01/2021

3. Initial data for the paper *Type of microcontroller/Raspberry Pi, , description of smart door-bell with real-time face recognition*

4. Paper contents (list of issues to be developed)
Introduction. 1. Analysis of subject area. 2. Hardware components the smart door-bell with real-time face recognition. 3. Software smart door-bell with real-time face recognition . 4. Occupational safety and health. Conclusions

5. List of graphic material (with exact number of required drawings, slides)

6. Advisors of paper chapters

Chapter	Advisor's surname, initials and position	Signature, date	
		assignment was given by	assignment was received by
Occupational safety and			
health			

7. Date of receiving the assignment _____

TIME SCHEDULE

LN	Paper stages	Paper stages deadlines	Notes
	Analysis of technical task		
	Analysis of subject area and system of the smart doorbell		
	Hardware components of system of doorbell		
	Software of the system		
	Occupational safety and health		
	Graphic materials		
	Preparation to the qualification work presentation		
	Qualification work presentation		

Student _____ Aboulfadel Mohamed.
(signature) (surname and initials)

Paper supervisor _____ Andriy Lutskiv
(signature) (surname and initials)

ABSTRACT

Raspberry Pi Real-Time Face Recognition Doorbell Qualifying paper //Aboulfadel Mohamed// Ternopil Ivan Puluj National Technical University, Fac-ulty of Computer Information Systems and Software Engineering, group ICI-43 //Ternopil, 2021 // p. – , fig. -40 figures, code snip. – 7, append. – 1, bibliogr. –.

Keywords: Raspberry Pi, OpenCV, Python, Pi camera, real-time face recognition.

CONTENT

Appendix A. Technical Task

INTRODUCTION

Being an understudy of designing, we have had insight in programming however never got an opportunity to create on equipment, for example, the Raspberry Pi. Working with genuine equipment, and realizing that this innovation is utilized worldwide in Blu-beam circles just as by site like YouTube, Vimeo and iTunes Store, spurred me further to take up this undertaking.

The human face assumes a significant part in our social communication, passing on individuals' personality. Utilizing the human face as a key to security, biometric face acknowledgment innovation has gotten huge consideration in the previous quite a long while because of its potential for a wide assortment of utilizations in both law requirement and non-law authorization. The most recent decade has been a decent one for face acknowledgment innovation. In spite of the fact that it has existed since the 1960s, any huge scope endeavors to effectively actualize it fizzled. The fundamental explanation being that it needs exactness and versatility. Everything changed in the last part of the 2000s.

When contrasted with other biometrics frameworks utilizing unique mark and iris, face acknowledgment has particular favorable circumstances on account of its non-contact measure. Facebook has revealed their face acknowledgment highlight that alarms you when untagged pictures and recordings of you are transferred in 2012. Google has utilized the organization's generous involvement with man-made intelligence and AI to assemble Google Photographs in 2015.

Face pictures can be caught from a distance without contacting the individual being distinguished, and the ID doesn't need cooperating with the individual. Also, face acknowledgment fills the wrongdoing hindrance need since face pictures that have been recorded and chronicled can later assistance distinguish an individual.

1. ANALYSIS OF SUBJECT AREA

1.1 Analysis of technical task

Home automation systems have truly struggled to find a mainstream audience, partly because they require their users to be little savvy, but the adoption from big companies and the fast proliferation of smart devices provide an easier way for even tech novies to be a part of this amazing and exiting future. Being a student of Computer Engineering gave me a chance to choose this project where im going to explain how I made a Smart Doorbell with Face recognition real-time. Using the Raspberry PI 3 model B+ and PI Camera. In The software part I used Python programing language to write my code and SQL for the Database.

1.2 System components justification

1.2.1 Main components for modelling

Lately extensive advancement has been made in the territory of face acknowledgment. Through crafted by software engineering engineers, PCs would now be able to beat people in many face acknowledgment errands, especially those where huge information bases of appearances should be looked. A framework with the capacity to identify and perceive faces has numerous potential applications including group and air terminal observation, private security and improved human-PC association. A programmed face acknowledgment framework is totally fit to fix security issues and offer adaptability to shrewd house control. This undertaking plans to supplant exorbitant picture handling sheets utilizing Raspberry pi board .This task

is essentially founded on picture preparing by porting the Opencv library to the Raspberry Pi board. Calculation for face acknowledgment.

Certifiable Employments of Face Acknowledgment Development. Face affirmation advancement is used in various organizations, from a way to deal with unwind to issues of public security.

Prosperity. Over the latest couple of years, a couple of crisis facilities started to use facial affirmation systems to make understanding getting ready less difficult. The circumstance organizes the right patient to their records, in this way thwarting record duplicates or record oversight.Another zone where facial acknowledgment is utilized, is crisis circumstances.

Either on the spot of a disaster or during the transportation to a crisis facility, patients are consistently quiet or inactive, which makes it hard to secure clinical information significant to the clinical thought. Facial affirmation offers a snappy strategy to get to the clinical information and, a significant part of the time, quicken the route toward giving imperative clinical thought.

Facial affirmation can in like manner help with screening for explicit disorders. For example, experts at Duke School developed a Psychological lopsidedness and Past application that uses the iPhone\'s front camera and facial affirmation computations to screen kids for mental unevenness.

Security. The security zone is a veteran concerning using facial affirmation. Face affirmation has been used at various country borders since the time the digitized biometric visa was introduced in 2006.

As we\'ve referred to beforehand, police powers are eager customers of face affirmation development as well. Maybe the main accomplishments of this advancement was in 2016 when the \"man in the cap\", responsible for the Brussels dread attacks, was recognized appreciation to FBI facial affirmation programming.

Some high-security workplaces, for instance, government structures and nuclear plants, realize facial affirmation advancement to check the characters of laborers.

Retail, Advancing, and Publicizing. Facial affirmation is up 'til now in its beginning in retail and exhibiting, anyway there are some charming way done by enormous associations.

A way to deal with use face affirmation advancement is by setting cameras in retail sources. That way it is possible to analyze and improve the customer purchasing measure by getting to customer information from their online media profiles and offering changed offers and things.The American retail establishment Saks Fifth Road is as of now utilizing such a framework. Amazon GO stores are supposedly utilizing it too.

Banking. The financial business is utilizing facial acknowledgment to both forestall extortion and making internet banking more secure. HSBC dispatched a Face ID confirmation alternative for their corporate customers in excess of 24 nations. The Face ID login is considerably quicker than Contact ID.

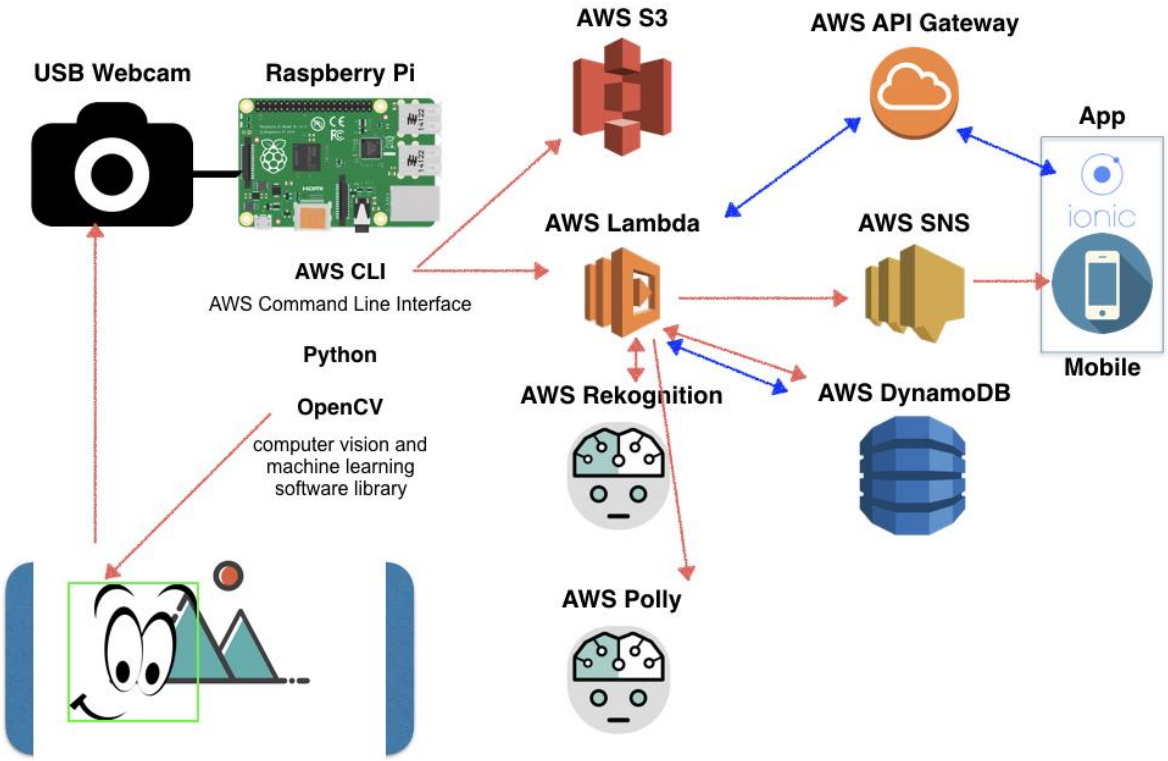

Fig.1.1 - Chart for facial-recognition using RPi

Facial Recognition using Raspberry Pi, Raspberry Pi camera

What Face Recognition Framework A facial acknowledgment framework is a PC application equipped for distinguishing or checking an individual from a computerized picture or video outline. It is commonly utilized in security frameworks and can be contrasted with other biometrics, for example, unique mark or eye iris acknowledgment frameworks. As of late, it has additionally gotten famous as a business ID and advertising apparatus

Its points of interest are:

- Least intrusive.
- More secure.

An important difference with other biometric solutions is that faces can be capture from some distance away, with for example surveillance cameras. Therefore face recognition can be applied without the subject knowing that he is being observed for Security Purpose.

Fig.1.2 - Photo for face detection

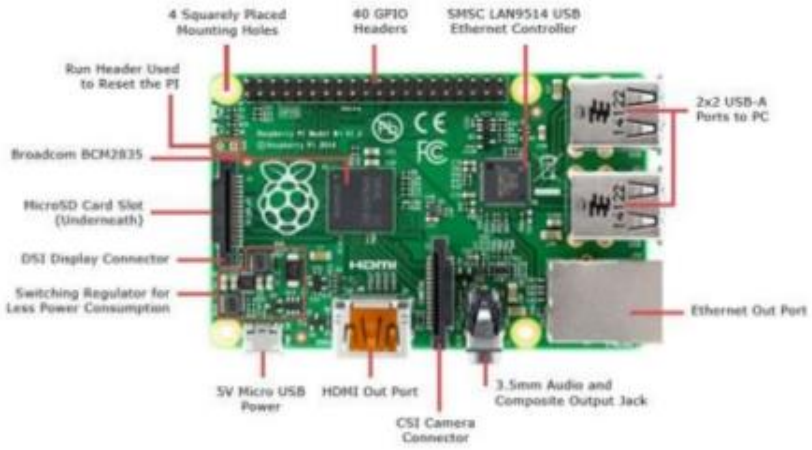

Fig.1.3 - Picture for hardware components of the raspberry Pi

A Raspberry Pi is a general-purpose computer, usually with a Linux operating system, and the ability to run programs. It is also low cost (($25- RS 1550) for model B/B+.

Raspberry Pi is Lighter, Smaller and Efficient.

It is known that it consume low power (5W) even less.

Supports Full HD video (1080p), Multiple USB Ports etc.

Smartcard swapping, alcohol detection and agriculture humidity sensing etc.

1.3. Software justification

Face Recognition technology has improved drastically in the past decade and now it is primarily used for surveillance and security purpose. I will build the Face Recognition Door Lock System using Raspberry Pi. This project consists of three phases:

- Data Gathering
- Training the Recognizer
- Face Recognition

In the first phase, i will collect the face samples that are authorized to open the lock. In the second phase, i will train the Recognizer for these face samples,

and in the last phase, trainer data will be used to recognize the faces. If raspberry pi recognizes a face, it will open the door lock.

- How Raspberry Pi face recognition doorbell works
  - Process each video frame at 1/4 resolution
  - Detect faces in every other frame of video.
  - Load sample pictures of known faces in memory and learn how to recognize.
  - See if the face is a match for the known face(s)
  - Call the “espeak” speech synthesis to say the name of person in video.

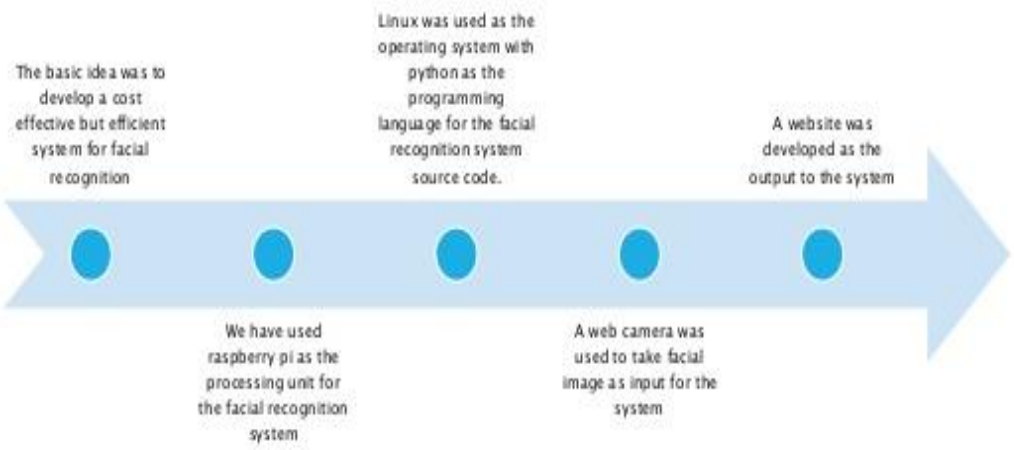

Fig.1.4 - Chart for the project description

Face Detection . Face location is a more extensive and less difficult term than face acknowledgment. Face identification is worried about distinguishing if a human face is available in a picture or video, not whom it has a place with.

Face discovery has a few applications. Camera auto component utilizes face discovery to figure out what parts of the image are generally significant. Face recognition can likewise be utilized to check the number of individuals have entered a specific region. It is additionally utilized without help from anyone else driving vehicles to decide whether a human is going across the street.

A paltry utilization of face location that you've presumably experienced yourself is Snapchat's filters. The channel searches for a presence of a human face and afterward maps the facial tourist spots (eyes, nose, lips, and so forth) to realize where to put, for instance, a bloom crown or feline ears.

Face Check. Face confirmation, likewise regularly alluded to as face verification, is tied in with approving a character dependent on the picture of a face by checking against a current information base.

Face validation includes contrasting an info picture just and the picture that has a place with the character the individual professes to be. All in all, the framework will just contrast your face with one picture, not the entire information base.

Face Acknowledgment. Face acknowledgment, likewise called distinguishing proof includes contrasting one info picture with all pictures in a picture library to figure out who the information picture has a place with. Or then again on the off chance that it doesn't have a place with the information base by any stretch of the imagination.

An illustration of a face acknowledgment framework would present an image of a suspect to the police information base to create a match, for example, they did at Panama's Tocumen air terminal.

Conclusions

So after the installation of the hardware components the phase of software comes next.

Data Gathering (the camera will gather the photos and videos)

Training the Recognize(training the recognize to detect my face and compare with other inputs (other peoples face))

Face recognition (my face is in the database so the system will recognize me)

These are the main steps for my system.

2. HARDWARE COMPONENTS OF THE SYSTEM

2.1. Hardware component justification

2.1.1. Raspberry pi3 B+

Raspberry Pi is the third top rated PC brand on the planet. The Raspberry Pi is a charge card measured PC that connects to your television or show, and a console and mouse. You can utilize it to pick up coding and to assemble hardware projects, and for a large number of the things that your work area PC does, similar to bookkeeping pages, word preparing, perusing the web, and messing around. It likewise plays superior quality video. The Raspberry Pi is being utilized by grown-ups and youngsters everywhere on the world to get the hang of programming and computerized making.

The Raspberry Pi 3 Model B is the most recent adaptation of the $35 Raspberry Pi PC. The Pi isn\'t care for commonplace machine, in its least expensive structure it doesn\'t have a case, and is just a charge card estimated electronic board - of the sort you may discover inside a PC or PC however a lot more modest. The Pi can be run as a spending work area, giving you have the tolerance.i can browse the net using my raspberry pi.

The Pi can run the official Raspbian OS, Ubuntu Mate, Snappy Ubuntu Core, the Kodi-based media centers OSMC and LibreElec, the non-Linux based Risc OS (one for fans of 1990s Acorn computers). It can also run Windows 10 IoT Core, which is very different to the desktop version of Windows.

Fig.2.5 - Picture of Raspberry pi

Beside that the cost of the RASPBERRY PI is not expensive, it depends on which model the person wants. I bought RASPBERRY PI 3 B+ for 35$.

Its consist of many part as I explained bellow

- Processor
- Broadcom BCM2387 chipset.
- 1.2GHz Quad-Core ARM Cortex-A53 (64Bit)
- 802.11 b/g/n Wireless LAN and Bluetooth 4.1 (Bluetooth Classic and LE)
- IEEE 802.11 b / g / n Wi-Fi. Protocol: WEP, WPA WPA2, algorithms AES-CCMP (maximum key
- length of 256 bits), the maximum range of 100 meters.
- IEEE 802.15 Bluetooth, symmetric encryption algorithm Advanced Encryption Standard (AES)
- with 128-bit key, the maximum range of 50 meters.
- GPU
- • Dual Core Video Core IV® Multimedia Co-Processor. Provides Open GL ES 2.0, hardware-accelerated
- Open VG, and 1080p30 H.264 high-profile decode.

– • Capable of 1Gpixel/s, 1.5Gtexel/s or 24GFLOPs with texture filtering and DMA infrastructure

– Memory 1GB LPDDR2

– Operating System

– Boots from Micro SD card, running a version of the Linux operating system or Windows 10 IoT

– Dimensions :85 x 56 x 17mm

– PowerMicro USB socket 5V1, 2.5A

– Connectors Ethernet10/100 BaseT Ethernet socket

– Video OutputHDMI (rev 1.3 & 1.4)Composite RCA (PAL and NTSC)

– Audio Output :Audio Output 3.5mm jack HDMIUSB 4 x USB 2.0 Connector

– GPIO Connector 40-pin 2.54 mm (100 mil) expansion header: 2x20 strip

– Providing 27 GPIO pins as well as +3.3 V, +5 V and GND supply lines

– Camera Connector

– 15-pin MIPI Camera Serial Interface (CSI-2)

– Display Connector

– Display Serial Interface (DSI) 15 way flat flex cable connector with two data lanes and a clock lane

– Memory Card Slot Push/pull Micro SDIO

The GPU provides Open GL ES 2.0, hardware-accelerated Open VG, and 1080p30 H.264

high-profile decode and is capable of 1Gpixel/s, 1.5Gtexel/s or 24 GFLOPs of general purpose compute. It means that if you plug the Raspberry Pi 3 into your HDTV, you could watch BluRay quality video, using H.264 at 40MBits/s

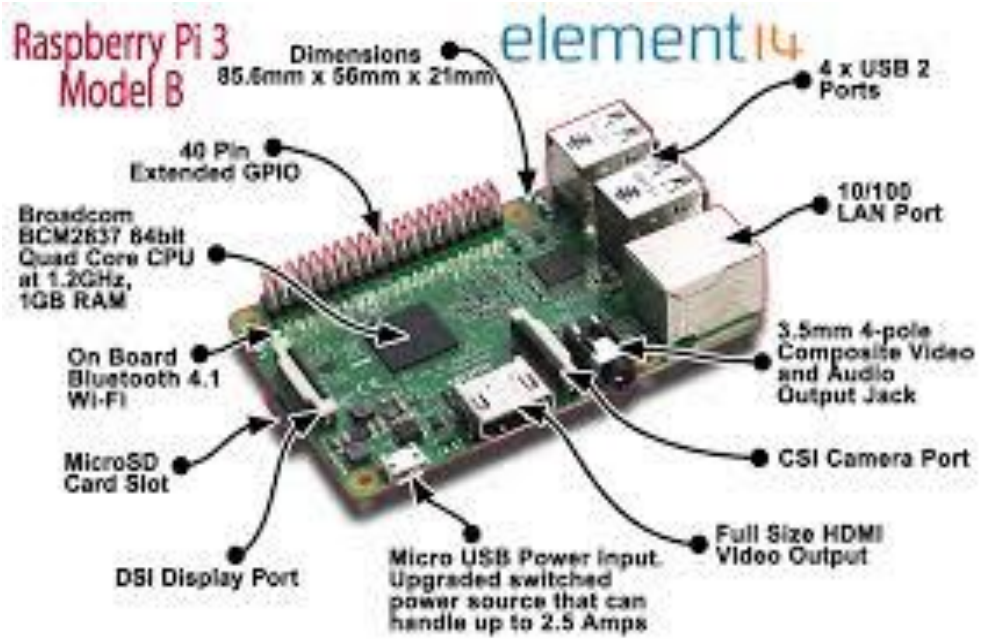

Fig.2.6 – Shematic of raspberry Pi3 B+

The greatest change that has been ordered with the Raspberry Pi 3 is a move up to a cutting edge principle processor and improved availability with Bluetooth Low Energy (BLE) also, BCM43143 Wi-Fi ready. Furthermore, the Raspberry Pi 3 has improved force the board, with an updated exchanged force source up to 2.5 Amps, to help more ground-breaking outside USB gadgets

Fig.2.7 – shematic of Raspberry pi3 B+

The Raspberry Pi 3's four underlying USB ports give enough availability to a mouse, console, or whatever else that you feel the RPi needs, however on the off chance that you need to add considerably more you can at present utilize a USB center point. Remember, it is suggested that you utilize a fueled center point so as

not to strain the on-board voltage controller. Controlling the Raspberry Pi 3 is simple, simply plug any USB power supply into the miniature USB port. There's no force button so the Pi will start to boot when force is applied, to turn it off basically eliminate power. The four inherent USB ports can even yield up to 1.2A empowering you to associate more power hungry USB gadgets (This requires a 2Amp miniature USB Force Supply).

On top of all that, the low-level peripherals on the Pi make it incredible for equipment hacking. The 0.1\" separated 40-nail GPIO header to the Pi gives you admittance to 27 GPIO, UART, I 2C, SPI just as 3.3 and 5V sources. Each pin on the GPIO header is indistinguishable to its archetype the Model B+.

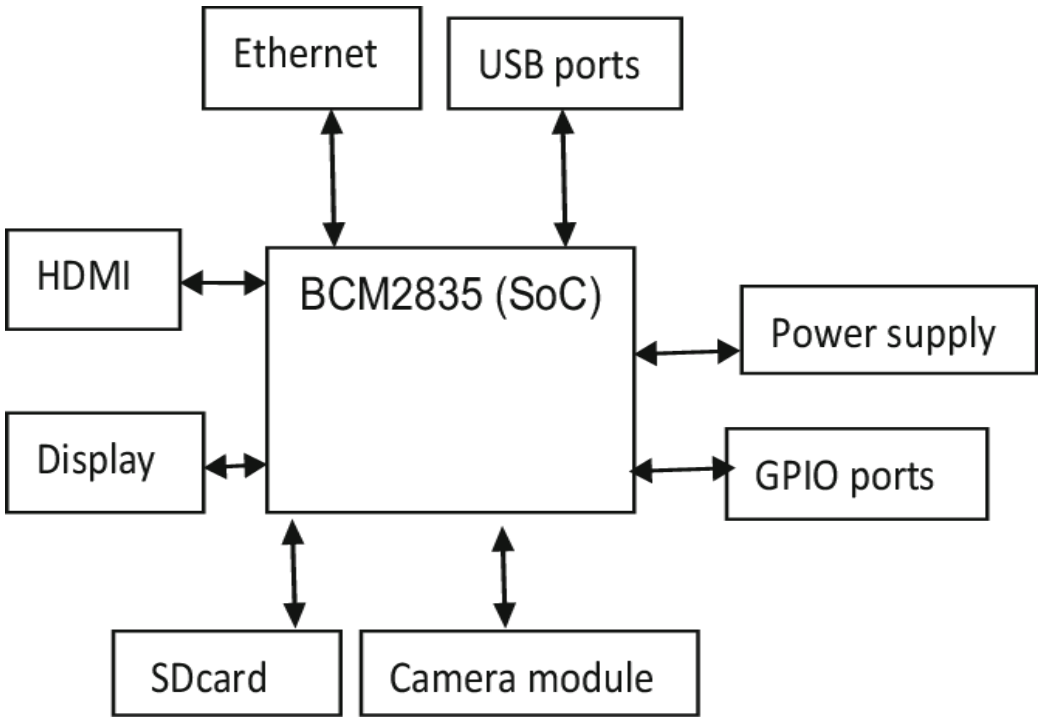

Fig.2.8 – Chart for the Raspberry pi3 B+

2.1.2. Raspberry pi Camera

The Pi camera module is a convenient light weight camera that upholds Raspberry Pi. It speaks with Pi utilizing the MIPI camera sequential interface convention. It is ordinarily utilized in picture handling, AI or in observation projects.The camera also doesn’t cost a lot, I bought mine for 20$.

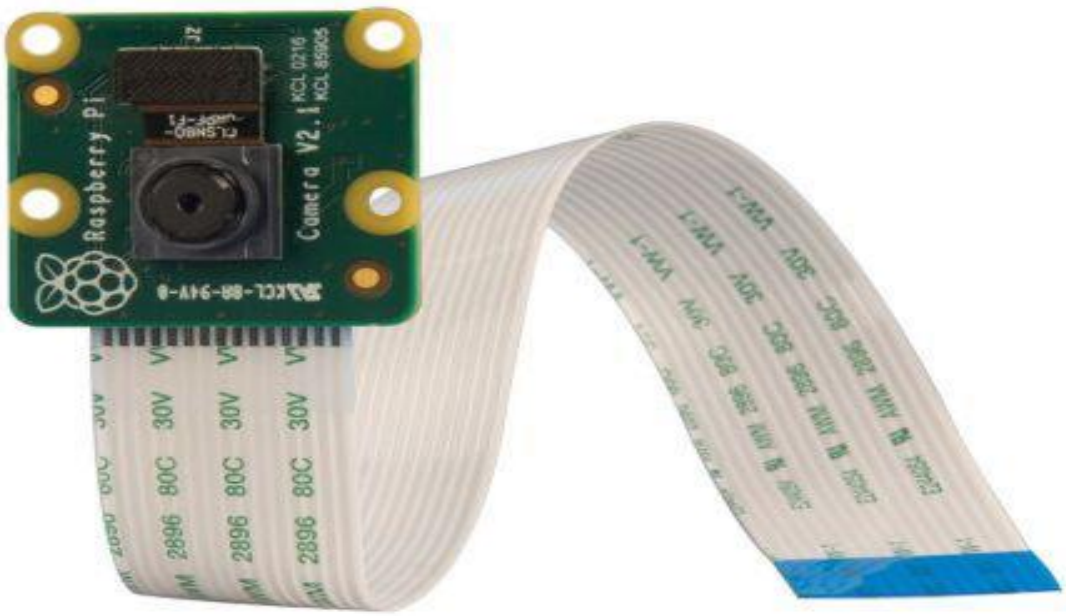

Fig.2.9 - Picture ofRaspberry pi camera

For existing Raspberry Pi clients the B+ will be that tad better for a wide scope of assignments: from a media focus ready to play a more extensive scope of video, because of the B+\'s better presentation and late help for equipment helped deciphering of HDCP-encoded 1080p H.

2.1.3. Pam 8403 Amplifier (Or any speaker)

PAM8403 is an Amplifier Board which can be powered using simple 5V input and could drive two 3W + 3W stereo speakers. ... This Amplifier allows the user to achieve high-quality audio reproduction from a stereo input. Additionally, it has a special feature that is, it can drive speakers directly from its output.

Fig.2.10 - Picure of Pam 8403 Amplifier

A RPI power adapter: usually it comes with the purchase of the raspberry pi. The power supply requirements differ by Raspberry Pi model. All models require a 5.1V supply, but the current supplied generally increases according to model. All models up to the Raspberry Pi 3 require a micro USB power connector, whilst the Raspberry Pi 4 uses a USB-C connector.

Fig.2.11 - Picture of RPI power adapter

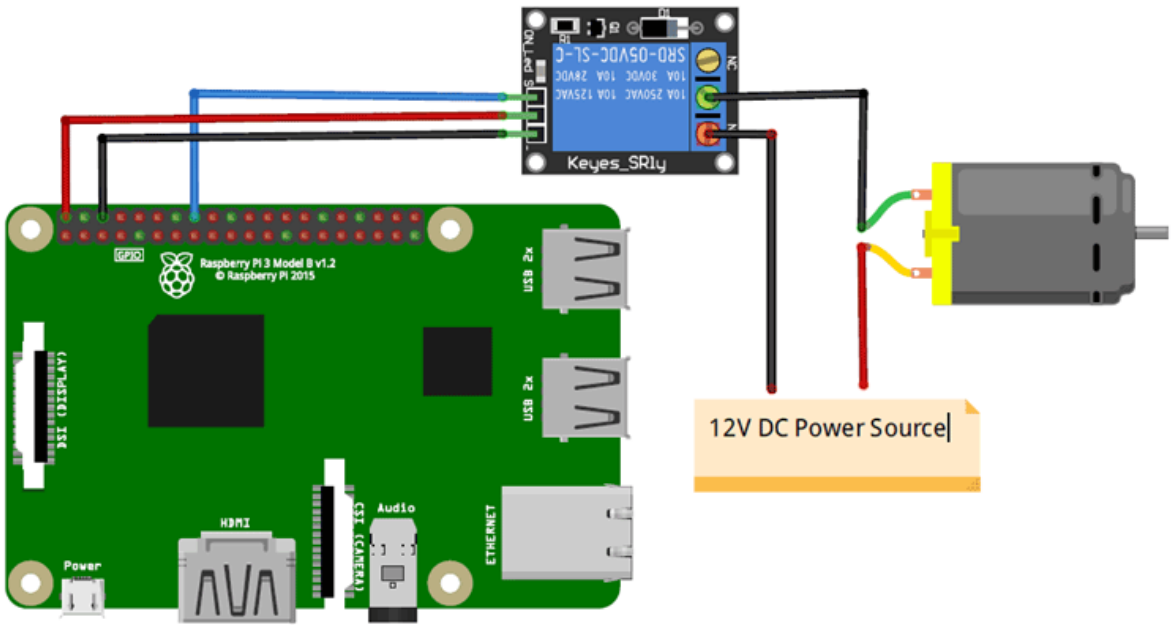

Fig.2.12 - Shematic of Circuit diagram

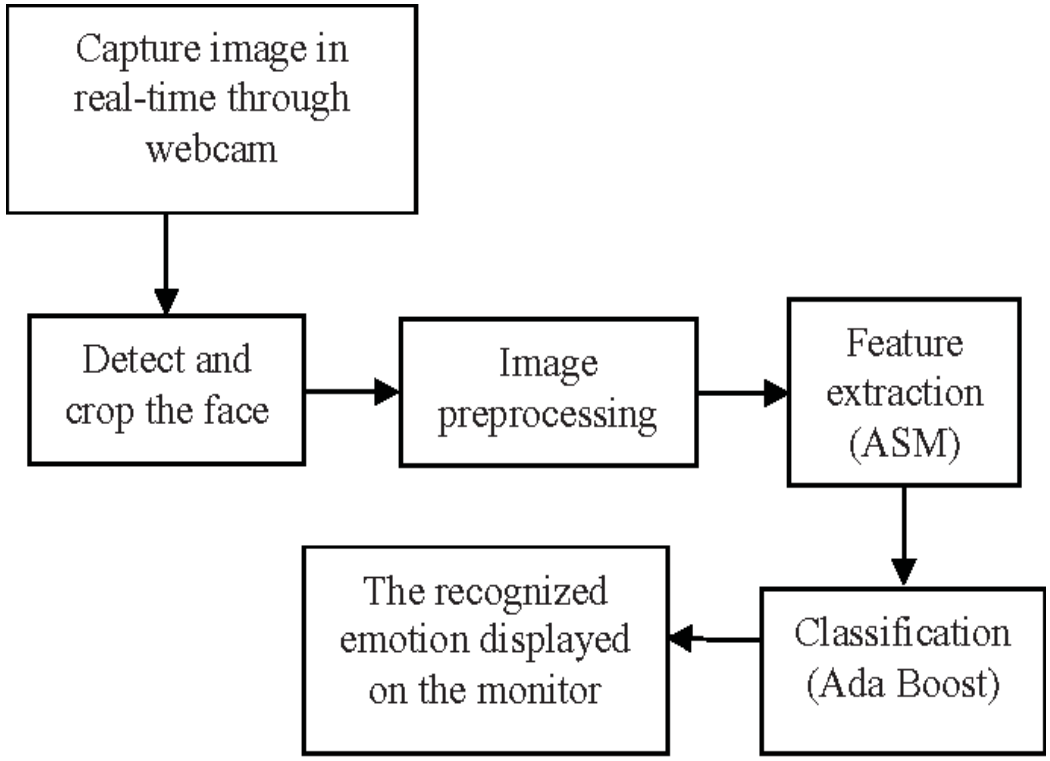

Fig.2.13 -Shematic of the system structure Diagram

2.1.4. Conclusions

The main Hardware components is (Raspberry Pi3 B+, Raspberry Camera). These two hardwares are enough to make a face recognition system. And here is some pictures of my own setup

Fig.2.14 –Photo of my Raspberry pi3 Model B+

Fig.2.15 - Photo of my Raspberry pi3 Model B+

Fig.2.16 - Photo of my Raspberry pi3 Model B+

Fig.2.17 - Photo of the camera ofRaspberry pi3 Model B+

Fig.2.18 - Photo of my setup

3. 3. SOFTWARE OF SYSTEM

3.1. Analysis of Raspberry Pi 3

The Raspberry Pi is a credit card-sized computer. The Raspberry Pi 3 Model B+ is an improved version of the Raspberry Pi 3 Model B. It is based on the BCM2837B0 system-on-chip (SoC), which includes a 1.4 GHz quad-core ARMv8 64bit processor and a powerful VideoCore IV GPU.

3.2. Python programming language justification

Python is a deciphered, object-arranged, elevated level programming language with dynamic semantics. ... Python\'s basic, simple to learn sentence structure underlines intelligibility and in this manner diminishes the expense of program upkeep. Python underpins modules and bundles, which energizes program seclusion and code reuse. Python is a dynamic, object-situated, significant level programming language that can be utilized for some sorts of programming improvement. Python programming language was made by Guido van Rossum. Python got its name from the BBC satire arrangement "Monty Python\'s Flying Bazaar". Python is a deciphered, elevated level and broadly useful programming language. Python\'s plan reasoning stresses code meaningfulness with its striking utilization of critical whitespace.

Highlights in Python

- Simple to codePython is an elevated level programming language.
- Free and Open Source.
- Item Situated Language.
- GUI Programming Backing.
- Significant Level Language.
- Extensible component.

– Python is convenient language.

– Python is coordinated language.

Python is a broadly useful coding language—which implies that, in contrast to HTML, CSS, and JavaScript, it tends to be utilized for different sorts of programming and programming advancement other than web improvement. That incorporates back end advancement, programming improvement, information science and composing framework contents in addition to other things.

3.3. Alghoritm of system

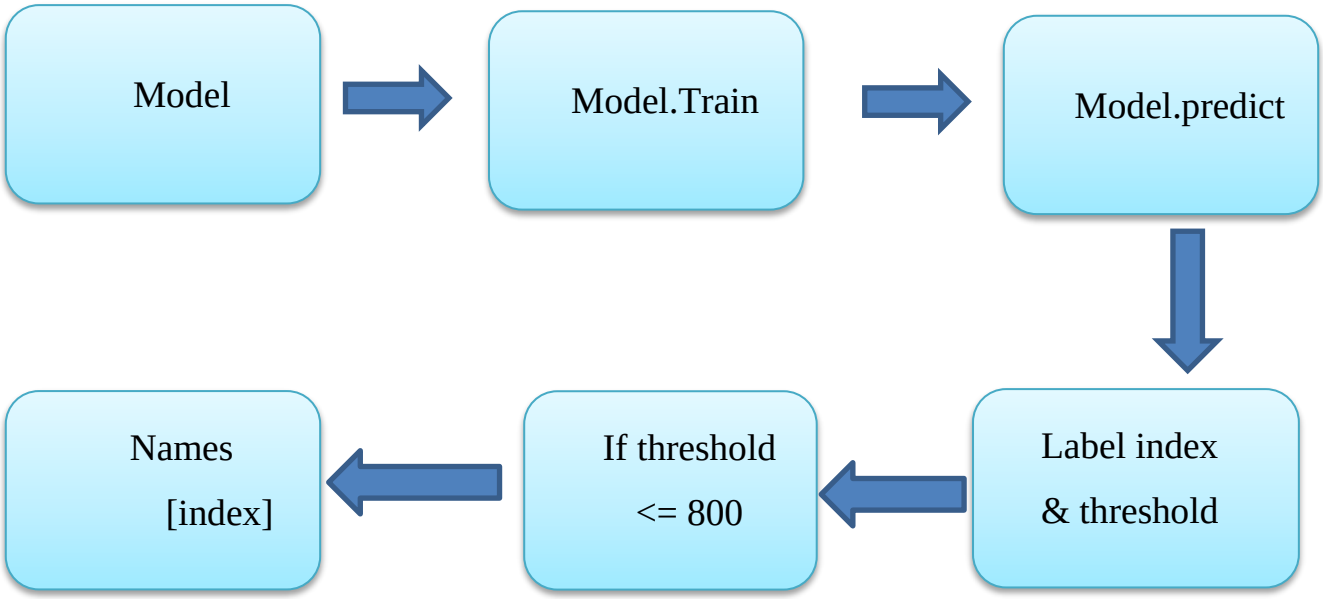

Fig.3.19 - Face Recognition Algorithm

Face Detection

For face detection, OpenCV cascade classifier will be used.

These trained classifiers include detectors of face, eyes, nose and whole body, etc.

Haar cascade classifier: The Viola- Jones object detection framework is the first object detection framework to provide competitive object detection rates in real-time.

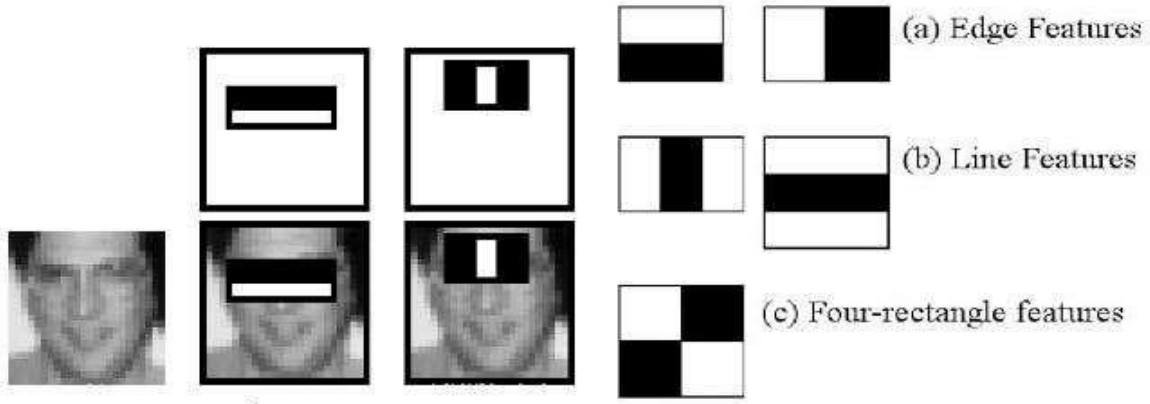

Fig.3.20 - Photo and shematic of Haarcascade

Three Popular Algorithms for Recognition:

- Eigen Faces
- Fisher faces
- Local binary pattern

Fisher Faces:

- Developed in 1997 by P.Belhumeur.
- Based on Fisher's Linear Discriminant Analysis (LDA)
- Faster than eigenfaces, , in some cases
- Has lower error rates
- Works well even if different illumination
- Works well even if different facial express.

Components for the system

- Raspberry Pi Board
- Power Supply
- Relay
- Power Adapter
- Camera Module

Proposed Work

A. Interfacing of camera module to capture live face image.

B. Create a database of authorized person.

C. Capture current face, save it and compare with database.

D. Interface relay as output module.

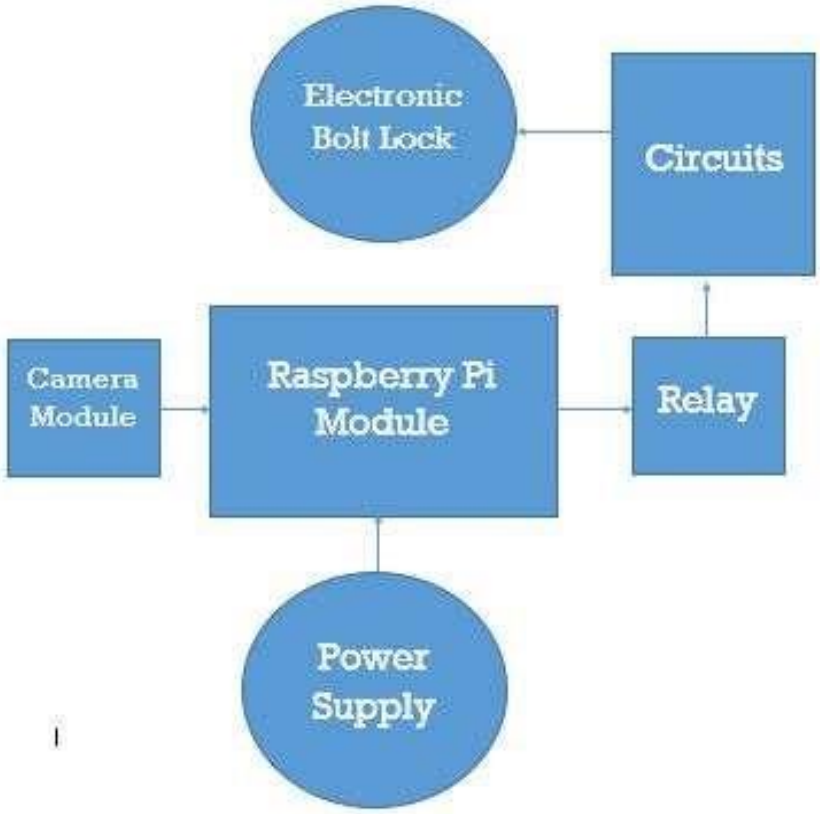

Fig.3.21 –Flow chart 2

Working and Methodology

1. Camera Module.
2. Raspberry Pi Module.
3. Electronic lock.

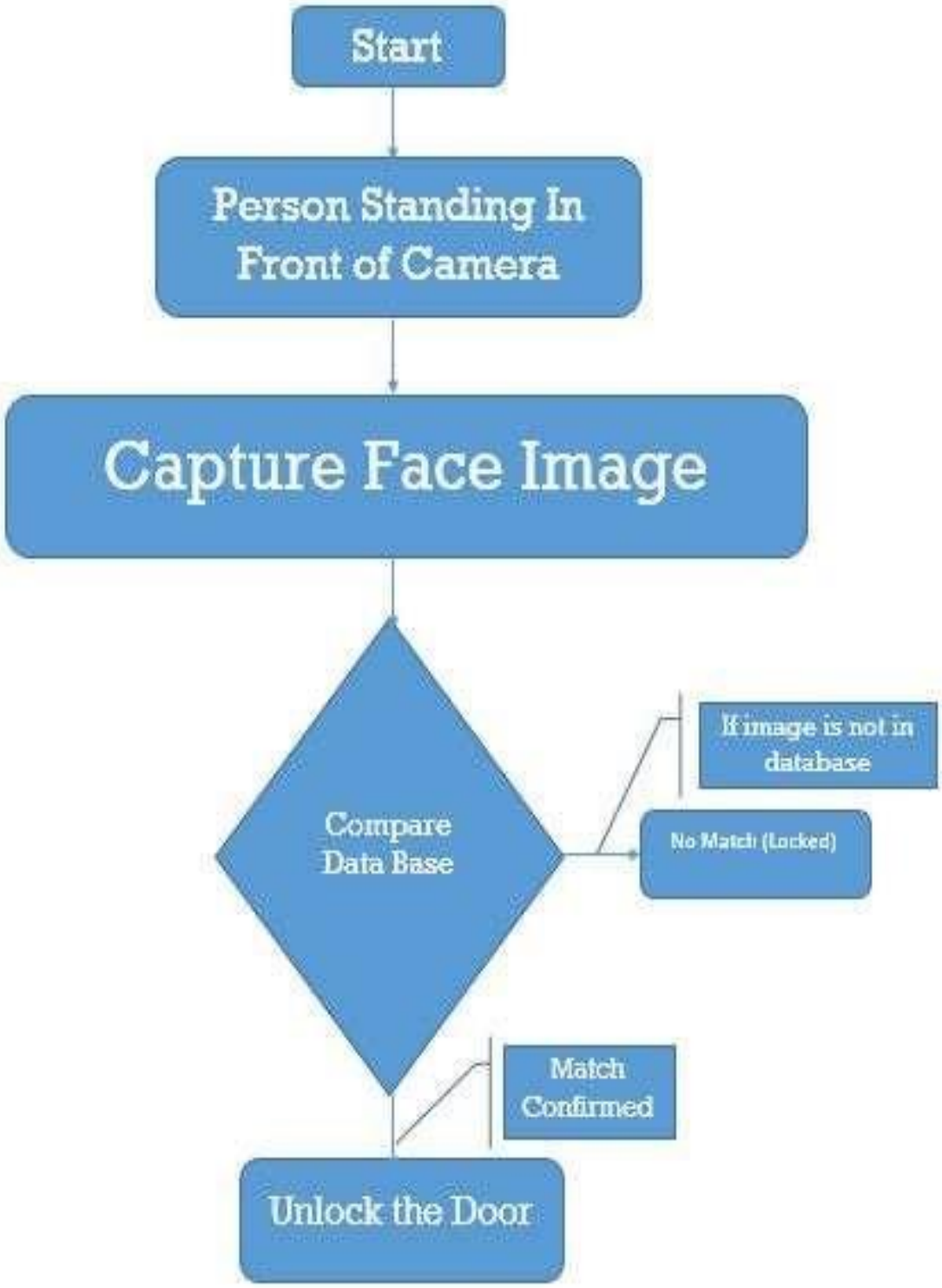

Fig.3.22 - Flow chart 1

I will explain the 3 phases now starting with the first one which is

1. Saving the Face Portion:

– Take Frame

- Detect Face
- Extract the Face
- Resize
- Save Extracted Face

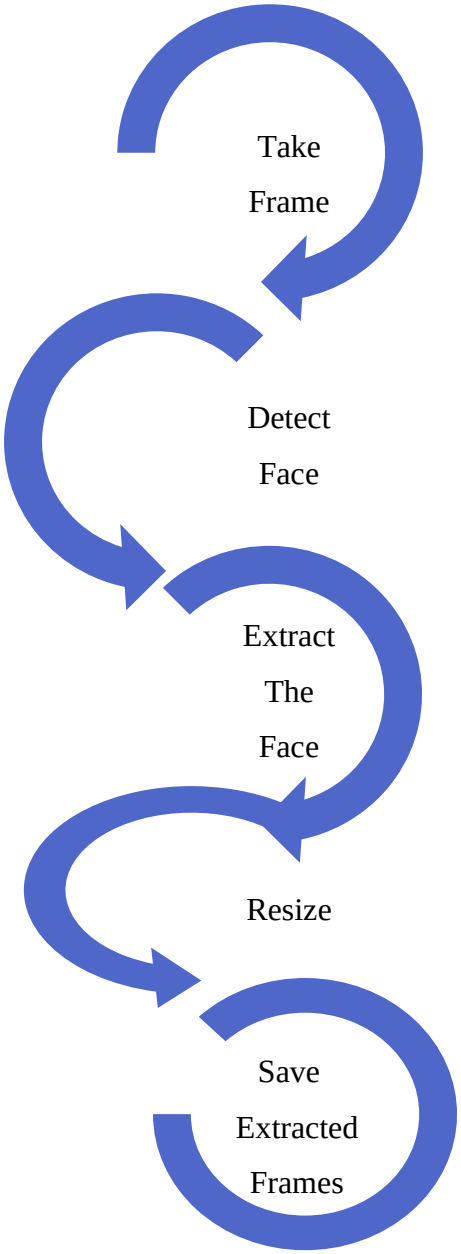

Fig.3.23 - Flow chart 2

Real Time Training:

- Input person name
- Take frame for 14s

– Detect face
– Save face
– Loop terminated.

At last the Essences of people will be saved in demonstrate envelope with the names of the people, those countenances will at that point be utilized to perceive face.

Real Time Face Recognition

– Get Appearances from Preparing organize.
– Compute Model
– Detect Face
– Match that face
– Get Constant edge
– Predict name

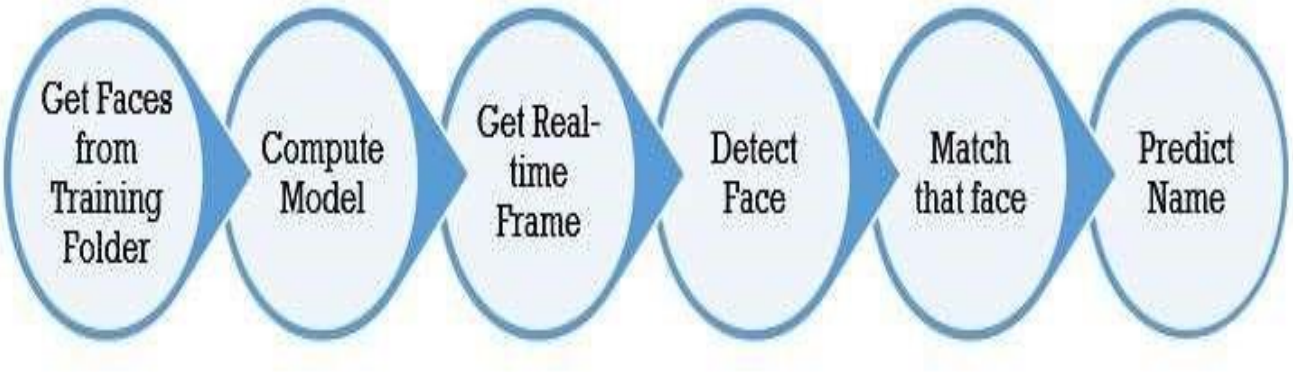

Fig.3.24 -Flow chart for the process of face recognition

3.4 Sofware components

3.4.1 Raspbian Stretch: Install OpenCV 3 + Python on myRaspberry Pi

Now the phase of programing, for that I had installed OpenCV 3 in my Raspberry Pi 3 I followed the following steps.

Step 1 the first one which is to Expand file system

In case of using a brand new insall of Raspbian Stretch then the first thing to do is expand your filesystem to include all available space on your micro-SD card:

So I open my terminal and type the comande bellow

```
$ sudo raspi-config
```

And then select the “Advanced Options”

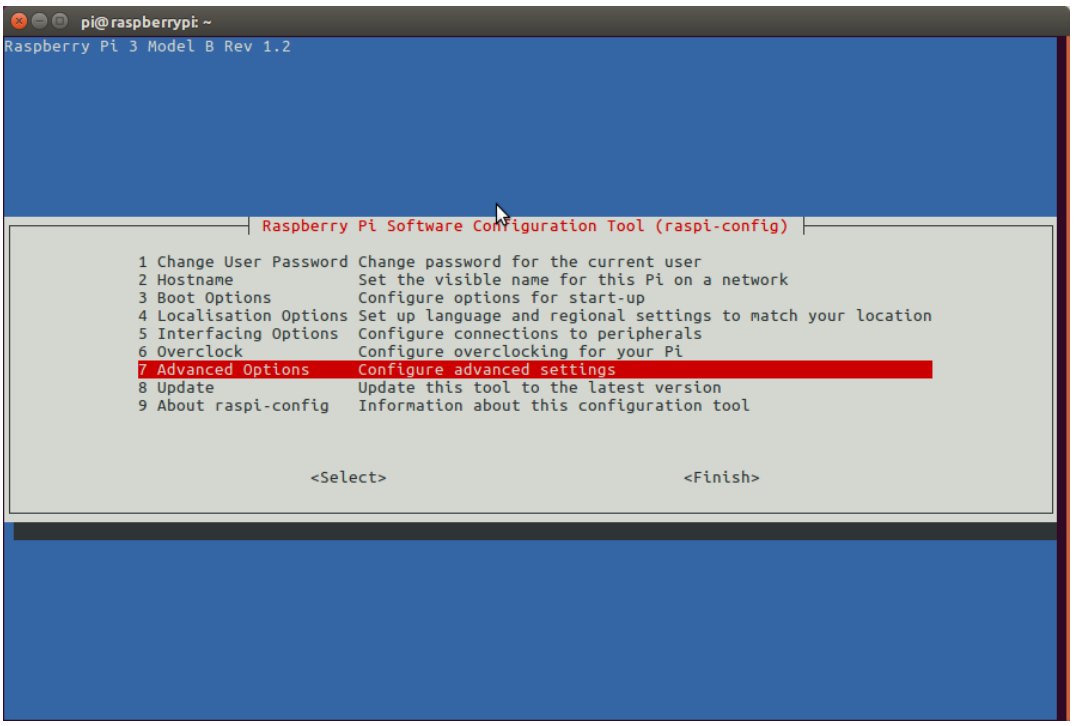

Fig.3.25 – Screenshot of the first step

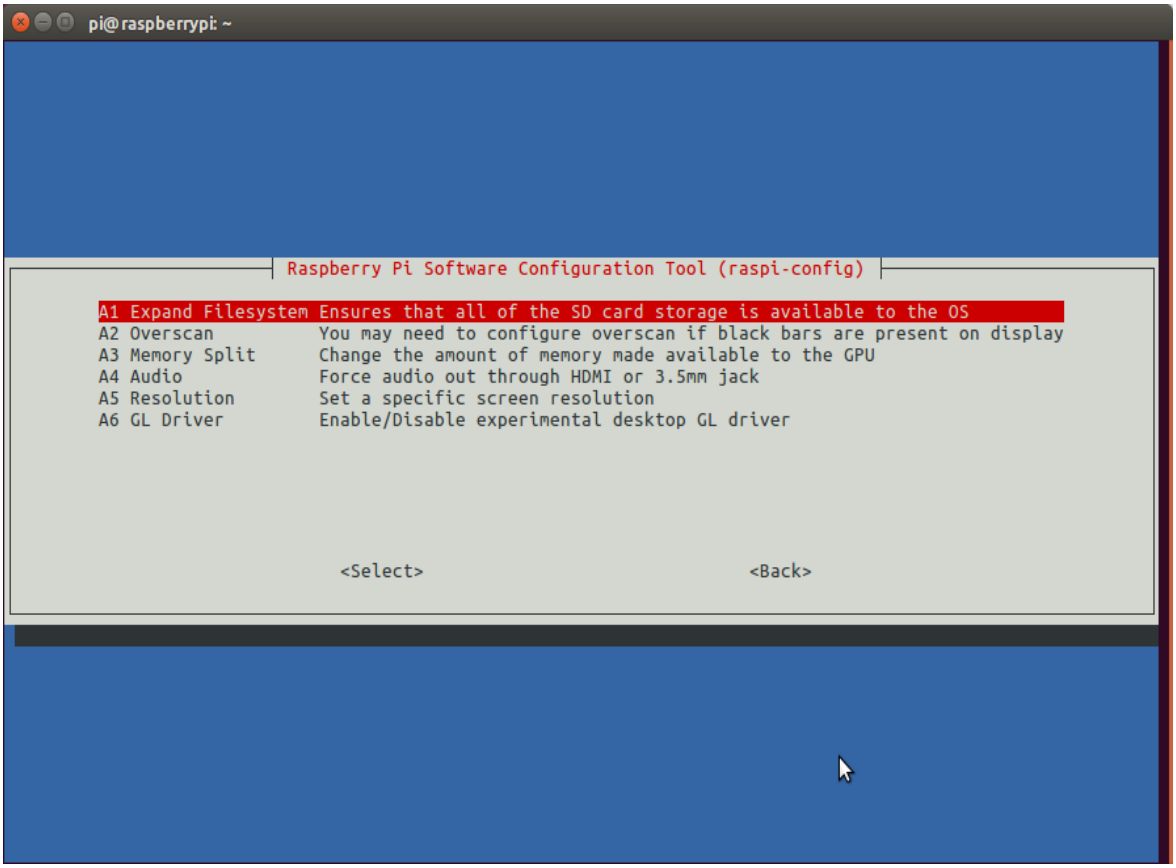

Fig.3.26 –Screenshot of the result

Once prompted, select the first option, “A1. Expand File System”, press Enter on ythe keyboard, arrow down to the “<Finish>” button, and then reboot the Pi.

After rebooting, the file system should have been expanded to include all available space on the micro-SD card.

Step 2 installing the dependencies.

The first step is to update and upgrade any existing packages by typing this comande in my terminal.

```
$ sudo apt-get update && sudo apt-get upgrade
```

Then install some developer tools, including CMake, which will helps me to configure the OpenCV build process

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

Next I will install some image I/O packages that allow me to load various image file formats from disk. Examples of such file formats include JPEG, PNG, and TIFF.

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

Also video I/O packages. These libraries allow me to read various video file formats from disk as well as work directly 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 comes with a sub-module named ‘highgui’ which is used to display images to our screen and build basic GUIs. In order to compile the ‘highgui’ module, i need to install the GTK development library bwllow I will show how

```
$ sudo apt-get install libgtk2.0-dev libgtk-3-dev
```

Now Matrix operations

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

Instalation of Python 2.7 and Python 3 header files

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

Step 3 I will download the OpenCV source code

Now that i have my dependencies installed, I will downoald the 3.3.0 archive of OpenCV from the official OpenCV repository.

```
$ cd ~
$ wget -O opencv.zip https://github.com/Itseez/opencv/archive/3.3.0.zip
$ unzip opencv.zip
```

Opencv_contrib repository

```
$ wget -O opencv_contrib.zip
https://github.com/Itseez/opencv_contrib/archive/3.3.0.zip
$ unzip opencv_contrib.zip
```

Step4 which is Python 2.7 or Python 3

Installing the pip

```
$ wget https://bootstrap.pypa.io/get-pip.py
$ sudo python get-pip.py
$ sudo python3 get-pip.py
```

Creating my Python virtual environment

Python virtual environment named CV using Python 2.7.

```
$ mkvirtualenv cv -p python2
```

This command will create a new Python virtual environment named CV using Python 3

```
$ mkvirtualenv cv -p python3
```

In case of reboot the Raspberry Pi; log out and log back in; or open up a new terminal, use the workon command to re-access the CV virtual environment. The mkvirtualenv command is meant to be executed only once: to actually create the virtual environment.

After that, I will jus use use workon and i'll be dropped down into my virtual environment.

```
$ source ~/.profile
$ workon cv
```

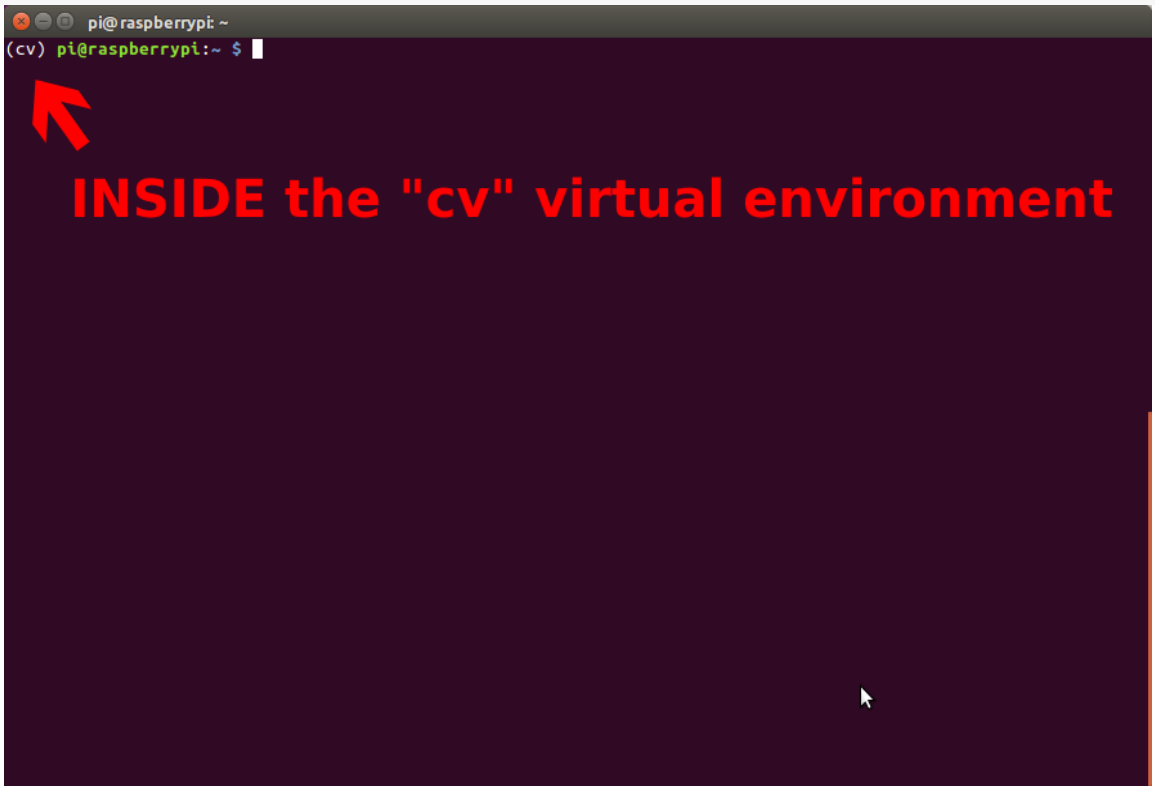

Fig.3.27 –Screenshot of the result

Installing NumPy on my Raspberry Pi

NumPy is a Python package used for numerical processing:

```
$ pip install numpy
```

Step 5 Compile and Install OpenCV

```
$ workon cv
```

Once i have ensured I am in the CV virtual environment, I will now setup my build using CMake:

```
$ cd ~/opencv-3.3.0/
$ 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-3.3.0/modules \
-D BUILD_EXAMPLES=ON ..
```

Then examination the ouput of CMake, for both python 2.7 and python 3

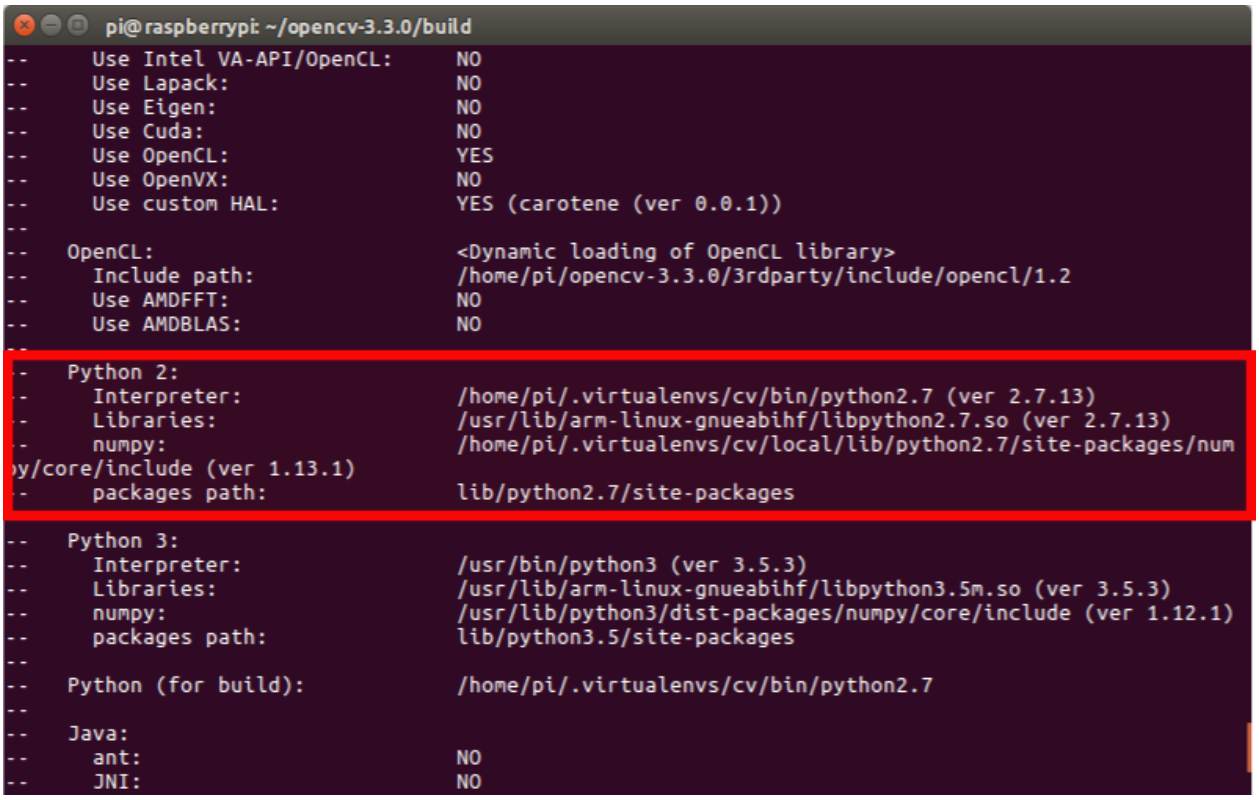

Fig.3.28 –Screenshot of the Result

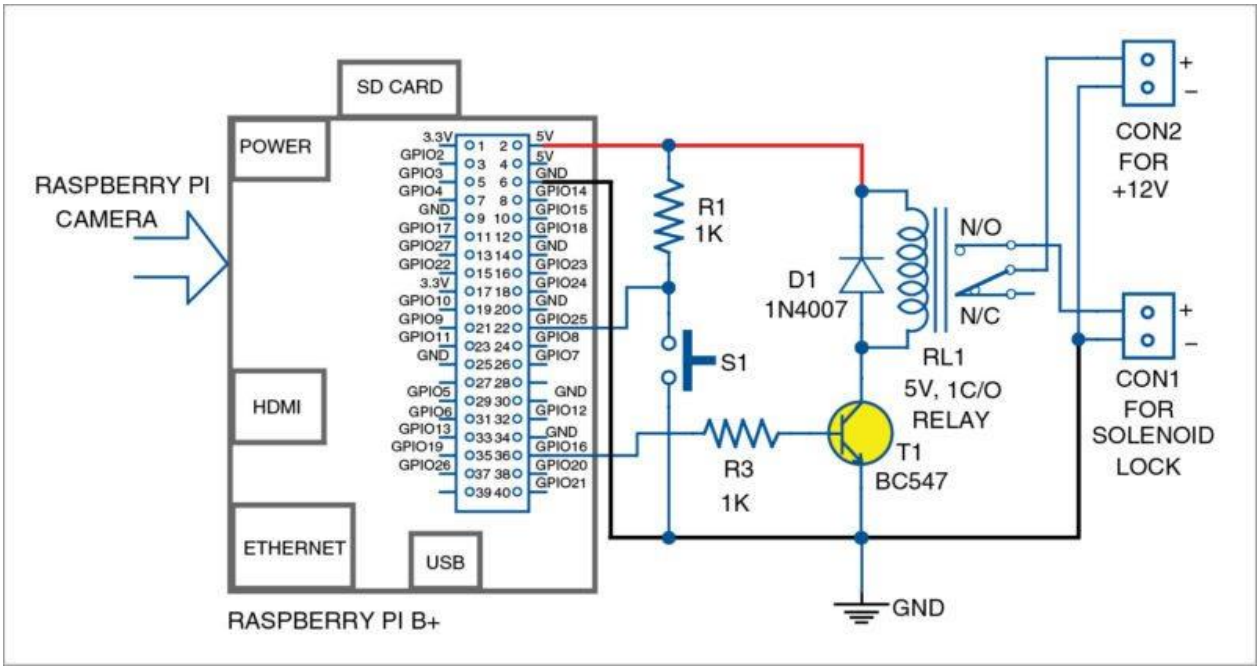

Fig.3.29 –Shematic of the Raspberry Pi circuits

```
pi@raspberrypi: ~/opencv-3.3.0/build
--     Use Cuda:                    NO
--     Use OpenCL:                  YES
--     Use OpenVX:                  NO
--     Use custom HAL:              YES (carotene (ver 0.0.1))
--
--   OpenCL:                        <Dynamic loading of OpenCL library>
--     Include path:                /home/pi/opencv-3.3.0/3rdparty/include/opencl/1.2
--     Use AMDFFT:                  NO
--     Use AMDBLAS:                 NO
--
--   Python 2:
--     Interpreter:                 /usr/bin/python2.7 (ver 2.7.13)
--     Libraries:                   /usr/lib/arm-linux-gnueabihf/libpython2.7.so (ver 2.7.13)
--     numpy:                       /usr/lib/python2.7/dist-packages/numpy/core/include (ver 1.12.1)
--     packages path:               lib/python2.7/dist-packages
--
--   Python 3:
--     Interpreter:                 /home/pi/.virtualenvs/cv/bin/python3 (ver 3.5.3)
--     Libraries:                   /usr/lib/arm-linux-gnueabihf/libpython3.5m.so (ver 3.5.3)
--     numpy:                       /home/pi/.virtualenvs/cv/lib/python3.5/site-packages/numpy/core/include (v
r 1.13.1)
--     packages path:               lib/python3.5/site-packages
--
--   Python (for build):            /usr/bin/python2.7
--
--   Java:
--     ant:                         NO
--     JNI:                         NO
--     Java wrappers:               NO
--     Java tests:                  NO
--
--   Matlab:                        Matlab not found or implicitly disabled
--
--   Documentation:
--     Doxygen:                     NO
--
--   Tests and samples:
--     Tests:                       YES
```

Fig.3.30 - Checking that Python 3 will be used when compiling OpenCV 3 for Raspbian Stretch on the Raspberry Pi 3.

Again, the Interpreter points to the python3.5 binary located in the CV virtual environment while numpy points to the NumPy install.

Configuration of the swap space size before compiling it

Increase our swap space size. This enables OpenCV to compile with all four cores of the Raspberry PI without the compile hanging due to memory problems.

In "/etc/dphys-swapfile" and then edit the "CONF_SWAPSIZE" variable

```
# set size to absolute value, leaving empty (default) then uses computed value
# you most likely don't want this, unless you have an special disk situation
# CONF_SWAPSIZE=100
CONF_SWAPSIZE=1024
```

I've commented and added a 1024MB line.

To activate the new swap spacewe should just restart the swap service

```
$ sudo /etc/init.d/dphys-swapfile stop
$ sudo /etc/init.d/dphys-swapfile start
```

Now the OpenCV ready to compile

```
$ make -j4
```

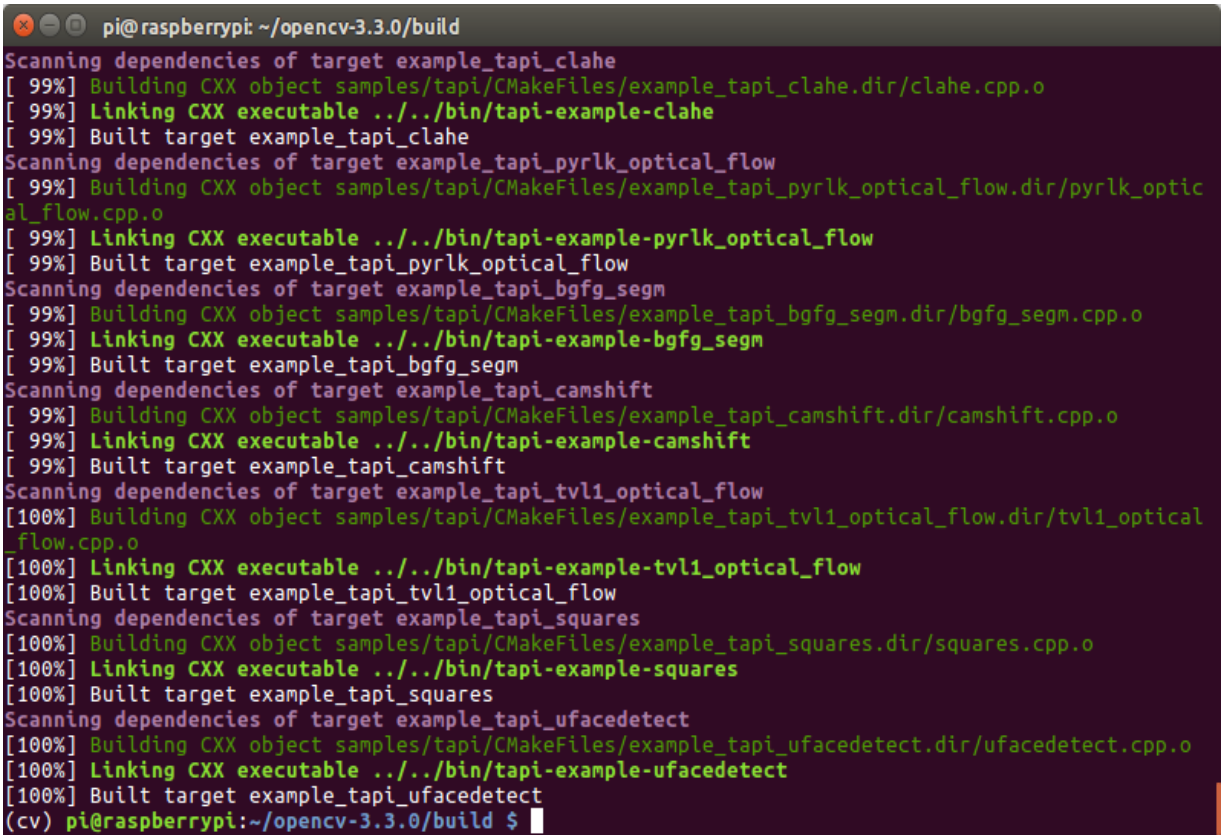

Fig.3.31 - OpenCV 3 compile on Rasp Stretch has completed successfully.

Then installing OpenCV 3 on Raspberry

```
$ sudo make install
$ sudo ldconfig
```

Step 6 Finish installing Opencv on the Pi

As python 3 after running make install, the OpenCV + Python bindings should be installed in this directory "/usr/local/lib/python3.5/site-packages"

```
$ ls -l /usr/local/lib/python3.5/site-packages/
total 1852
-rw-r--r-- 1 root staff 1895932 Mar 20 21:51 cv2.cpython-34m.so
```

Step 7 is theTesting of Opencv 3 installation

New terminal, execute the source and workon commands, finally attempting to import the Python + Opencv bindings

```
$ source ~/.profile
$ workon cv
$ python
>>> import cv2
>>> cv2.__version__
'3.3.0'
>>>
```

Here is the screenshot of my own terminal, OpenCV 3 has been successfully installed on my Raspberry Pi 3 + Python 3.5 environment:

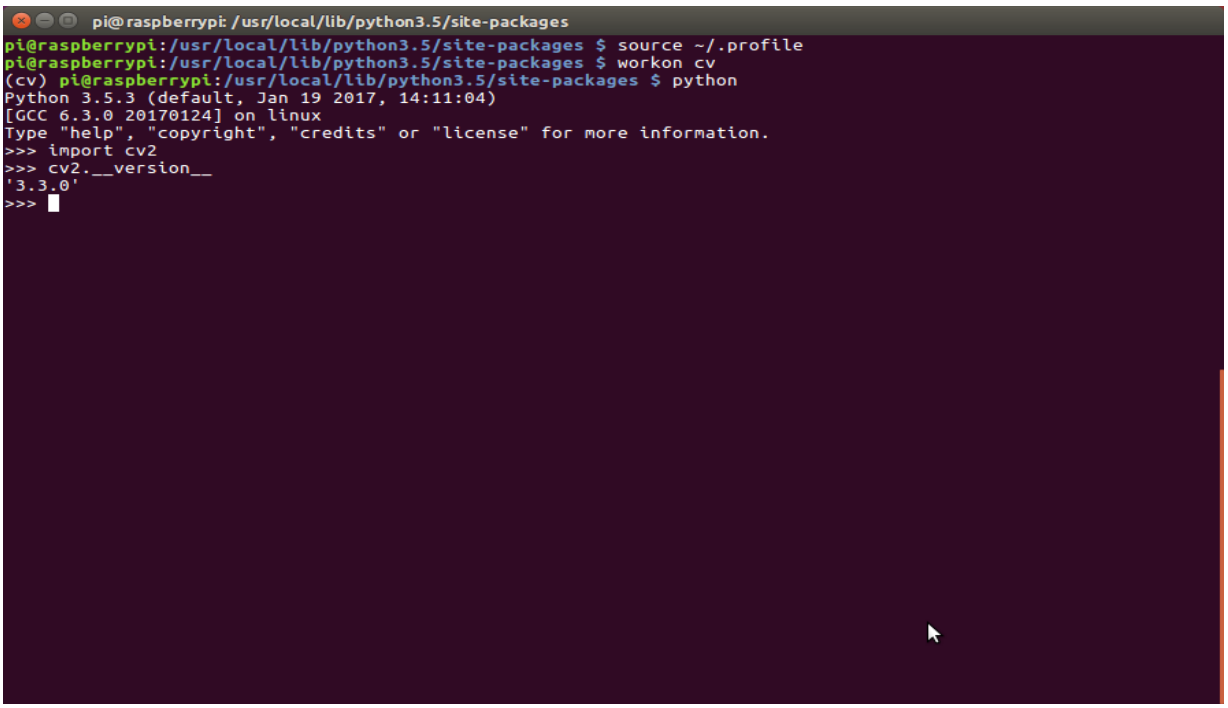

Fig.3.32 - Confirming OpenCV 3 has been successfully installed on my Raspberry Pi 3 running Raspbian Stretch.

Once OpenCV has been installed, to remove the contrib I will tupe the command bellow

```
$ rm -rf opencv-3.3.0 opencv_contrib-3.3.0
```

Changing the swap size back

Open /etc/dphys-swapfile and then edit the CONF_SWAPSIZE variable:

```
# set size to absolute value, leaving empty (default) then uses
    computed value
#   you most likely don't want this, unless you have an special disk
    situation
CONF_SWAPSIZE=100
# CONF_SWAPSIZE=1024
```

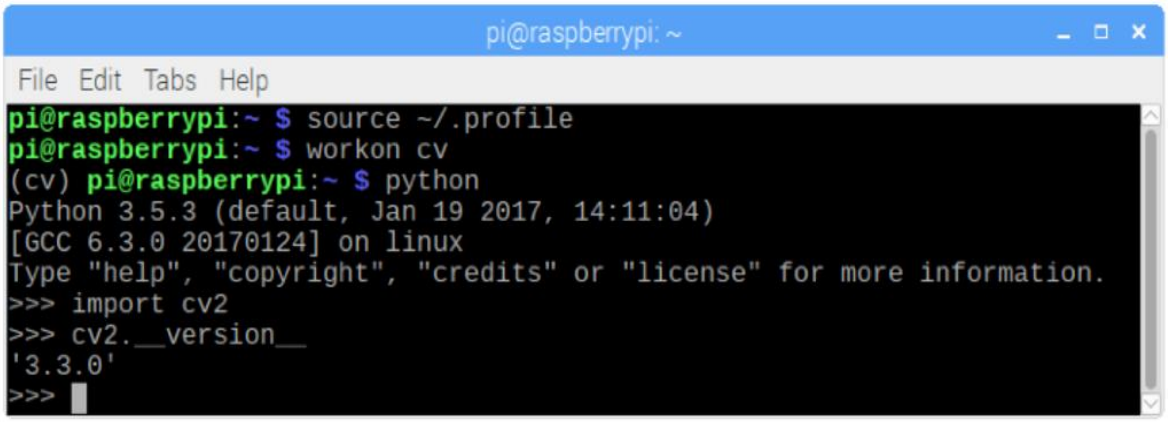

Fig.3.33 –Screeshot of the terminal

After installing Raspbian (Stretch) and the openCV 3 package next thing will be the camera tasting.

And from the picture below we can see our Harware set:

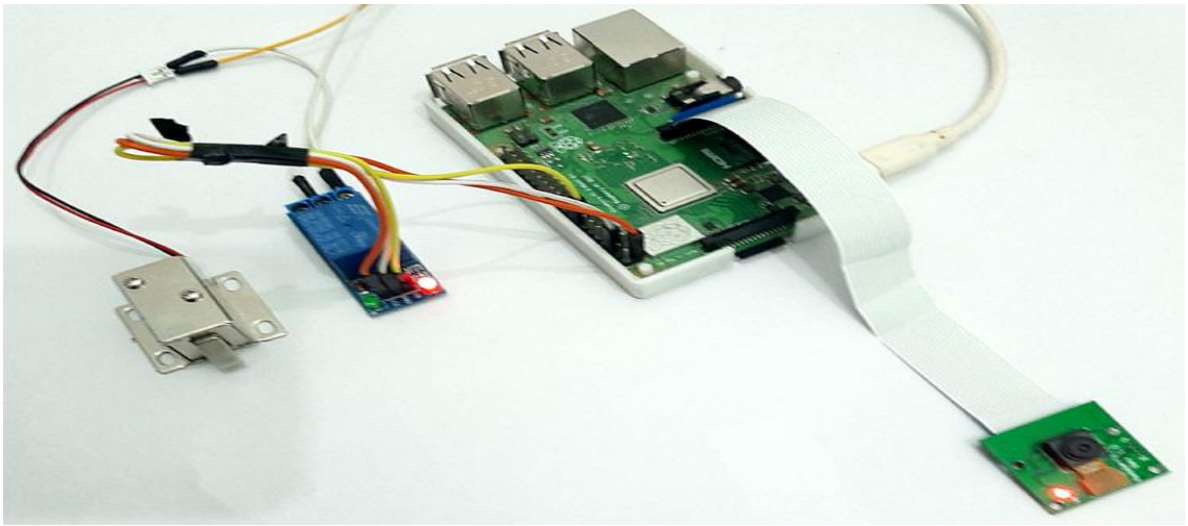

Fig.3.34 - Picture of the system hardware

Once OpenCV is installed in theRPi I will test to confirm that the camera is working properly.Camera testing

Typing the below python code

```
import numpy as np
import cv2


        cap = cv2.VideoCapture(0)
        cap.set(3,640) # set Width
        cap.set(4,480) # set Height

        while(True):
            ret, frame = cap.read()
            frame = cv2.flip(frame, -1) # Flip camera vertically
            gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

            cv2.imshow('frame', frame)
            cv2.imshow('gray', gray)

            k = cv2.waitKey(30) & 0xff
            if k == 27: # press 'ESC' to quit
                break

        cap.release()
        cv2.destroyAllWindows()
```

The above code will capture the video stream that will be generated by the PiCam, displaying both, in BGR color and Gray mode.

To execute

```
python simpleCamTest.py
```

In order to finish the program press [ESC] key. Face Detection

The most essential errand on Face Acknowledgment is obviously, \"Face Detecting\". Prior to anything, you should \"capture\" a face (Stage 1) to remember it, when contrasted and another face caught on future (Stage 3).

The most widely recognized approach to identify a face (or any items), is utilizing the \"Haar Course classifier\"

Thing Area using Haar feature based course classifiers is a suitable article recognizable proof method proposed by Paul Viola and Michael Jones in their paper, \\\"Rapid Article Disclosure using an Upheld Course of Fundamental Features\\\" in 2001. If you would incline toward not to make your own classifier,OpenCV as of now contains different pre-orchestrated classifiers for face, eyes, grin, and so forth Those XML records can be download from haarcascades stock.

As of now we should call our classifier work, passing it some huge limits, as scale factoHere we will work with face distinguishing proof. From the start, the estimation needs a huge load of positive (pictures of faces) and negative (pictures without faces) to set up the classifier. By then we need to isolate features from it. Luckily OpenCV goes with a mentor similarly as a locater. In case you need to set up your own classifier for any thing like vehicle, planes, etc you can use OpenCV to make one. Its full nuances are given here: Course Classifier Planning. If you would incline toward not to make your own classifier, OpenCV as of now contains various pre-arranged classifiers for face, eyes, smile, etc Those XML records can be download from haarcascades inventory.

The code bellow:

```
import numpy as np
import cv2

faceCascade =
cv2.CascadeClassifier('Cascades/haarcascade_frontalface_default.xml')

cap = cv2.VideoCapture(0)
cap.set(3,640) # set Width
cap.set(4,480) # set Height

while True:
    ret, img = cap.read()
    img = cv2.flip(img, -1)
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```

```
faces = faceCascade.detectMultiScale(
        gray,
        scaleFactor=1.2,
        minNeighbors=5,
        minSize=(20, 20)
    )

    for (x,y,w,h) in faces:
        cv2.rectangle(img,(x,y),(x+w,y+h),(255,0,0),2)
        roi_gray = gray[y:y+h, x:x+w]
        roi_color = img[y:y+h, x:x+w]

    cv2.imshow('video',img)

    k = cv2.waitKey(30) & 0xff
    if k == 27: # press 'ESC' to quit
        break

cap.release()
cv2.destroyAllWindows()
```

Note the line below

```
faceCascade =
    cv2.CascadeClassifier('Cascades/haarcascade_frontalface_default.
    xml')
```

This is the line that heaps the classifier.

At that point, i will set my camera and inside the circle, load my infos video in grayscale mode.

By and by i will call the classifier work, passing it some huge limits, as scale factor, number of neighbors and least size of the recognized face.its important informations.

```
faces = faceCascade.detectMultiScale(
        gray,
        scaleFactor=1.2,
        minNeighbors=5,
        minSize=(20, 20)
    )
```

Gray is the input grayscale image. ScaleFactor is the parameter specifying how much the image size is reduced at each image scale. It is used to create the scale pyramid.

MinSize is the base square shape size to be viewed as a faceMinSize is the base square shape size to be viewed as a face.

The capacity will distinguish faces on the picture. Next, mark the appearances in the picture, utilizing, for instance, a blue square shape. This is finished with this part of the code

```
for (x,y,w,h) in faces:
    cv2.rectangle(img,(x,y),(x+w,y+h),(255,0,0),2)
    roi_gray = gray[y:y+h, x:x+w]
    roi_color = img[y:y+h, x:x+w]
```

Recognizer will make a prediction reestablishing its id and a record, demonstrated how sure the recognizer is with this match. (X, y) and having "w" as its Width and "h" as its Height (x, y, w, h).

Once these locations are done, im going to create an ROI (drawn rectangle) for the face and present the result with imshow () function.

Using the Rpi Terminal Run the above python Script on your python environment:

```
python faceDetection.py
```

Data Gathering

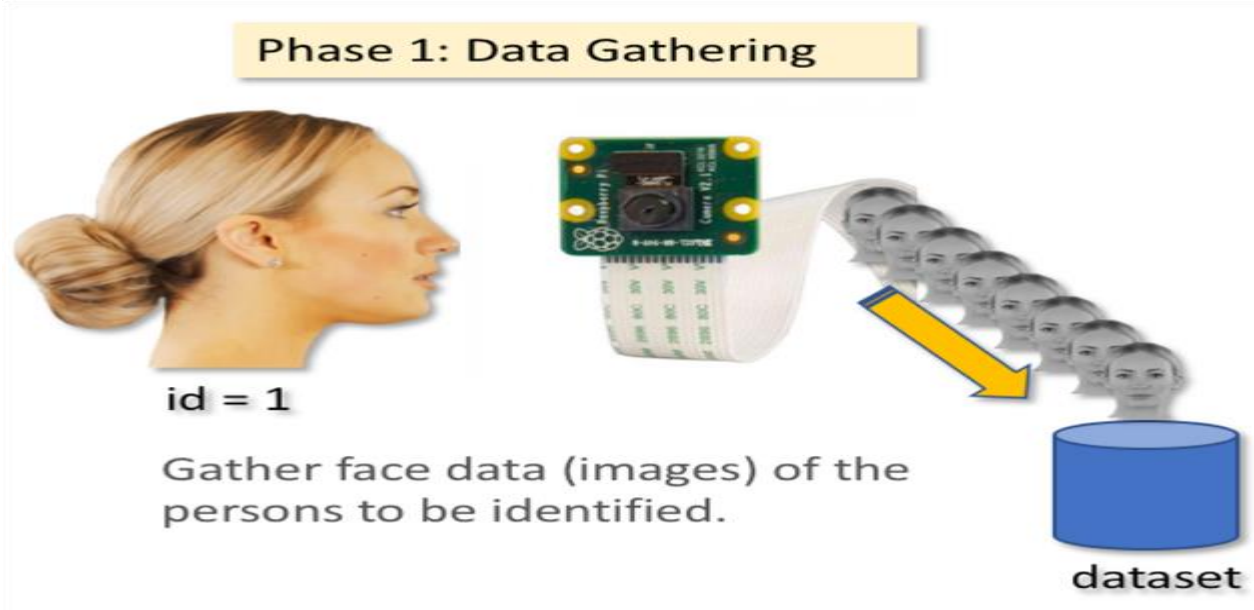

Fig.3.35 - Demonstration of data gathering

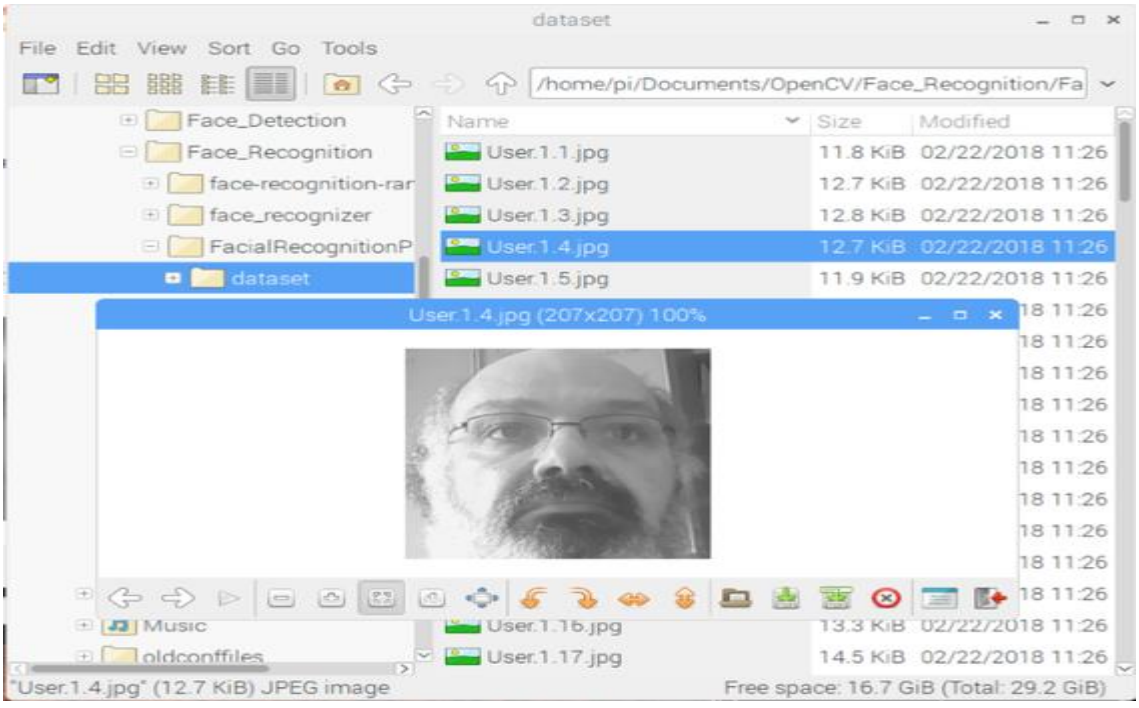

Fig.3.36 - Screenshot of database (example)

Starting from last step (Face Detecting), I will create a dataset, where i will store for each id and a group of photos in gray with the portion that was used for face detecting.

First, I create a directory where I will develop my project.

```
mkdir FacialRecognitionProject
```

In this directory, besides the 3 python scripts that we will create for our project, we must have saved on it the Facial Classifier.

Next, I will create a subdirectory where im going to store my facial samples and name it dataset as I will show here

```
mkdir dataset
```

The code bellow:

```
import cv2
import os

cam = cv2.VideoCapture(0)
cam.set(3, 640) # set video width
cam.set(4, 480) # set video height

face_detector =
    cv2.CascadeClassifier('haarcascade_frontalface_default.xml'
    )

# For each person, enter one numeric face id
face_id = input('\n enter user id end press <return> ==>  ')

print("\n [INFO] Initializing face capture. Look the camera and
    wait ...")
# Initialize individual sampling face count
count = 0

while(True):
    ret, img = cam.read()
    img = cv2.flip(img, -1) # flip video image vertically
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    faces = face_detector.detectMultiScale(gray, 1.3, 5)

    for (x,y,w,h) in faces:
        cv2.rectangle(img, (x,y), (x+w,y+h), (255,0,0), 2)
        count += 1
```

```
            # Save the captured image into the datasets folder
            cv2.imwrite("dataset/User." + str(face_id) + '.' +
        str(count) + ".jpg", gray[y:y+h,x:x+w])

            cv2.imshow('image', img)

        k = cv2.waitKey(100) & 0xff # Press 'ESC' for exiting video
        if k == 27:
            break
        elif count >= 30: # Take 30 face sample and stop video
             break

    # Do a bit of cleanup
    print("\n [INFO] Exiting Program and cleanup stuff")
    cam.release()
    cv2.destroyAllWindows()
```

The code is very similar to the code that I used for face detection. In addition I've added an input command to capture a user id, which should be an integer number A, B, C...

```
            face_id = input('\n enter user id end press  ==>  ')
```

And for each one of the captured frames, I will save it as a file on a (dataset) directory

```
            cv2.imwrite("dataset/User." + str(face_id) + '.' + str(count) +
                ".jpg", gray[y:y+h,x:x+w])
```

So, for saving the above file, I must have imported the library (os). Each file's name will follow the structure here is the command

```
    User.face_id.count.jpg
```

For example, for a user with a face_id = 1, the 4th sample file on dataset/ directory will be something like

```
User.1.4.jpg
```

The number of samples is used to break the loop where the face samples are captured. The Trainer phase

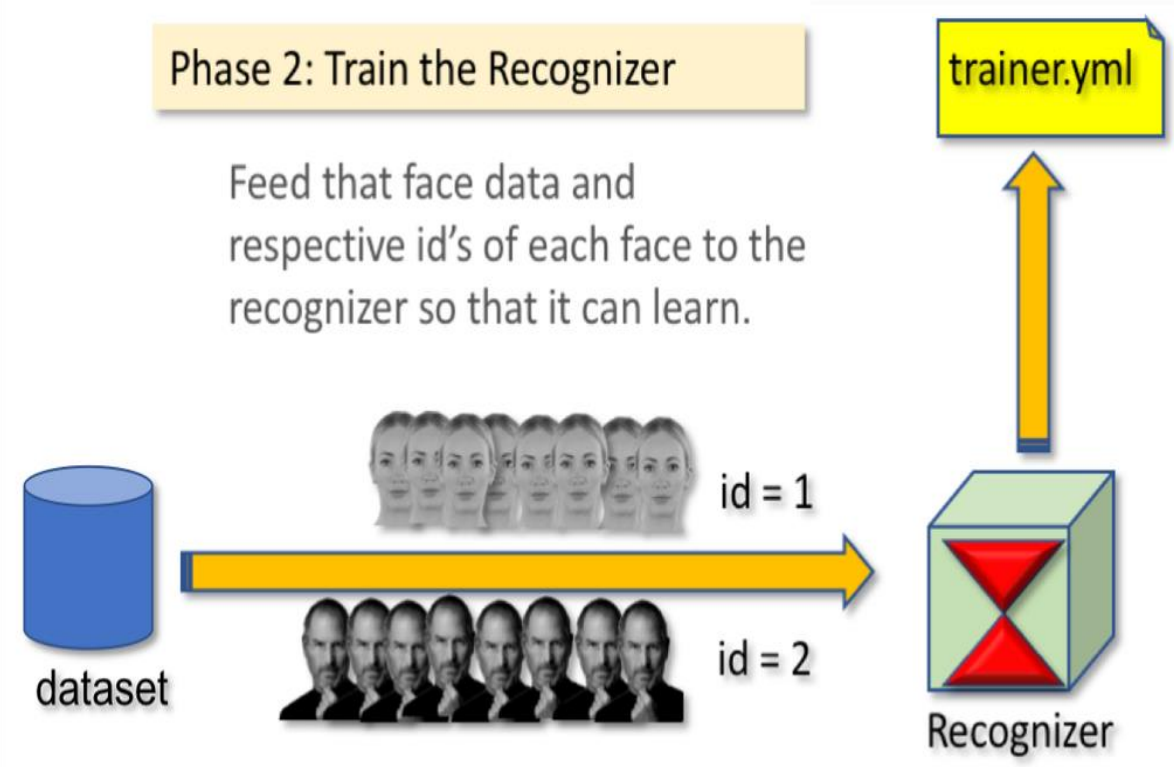

Fig.3.37 - Shecmatic demonstration of trainer

On this second phase, I am taking all user data from the dataset and trainer the OpenCV Recognizer. This is done directly by a specific OpenCV function. The result will be a .yml file that will be saved on a trainer directory.

Creating a subdirectory where iwill store the trained data

```
mkdir trainer
```

And we type the code bellow to get our result

```
        import cv2
  import numpy as np
  from PIL import Image
  import os

    # Path for face image database
path = 'dataset'

recognizer = cv2.face.LBPHFaceRecognizer_create()

        # Path for face image database
        path = 'dataset'

        recognizer = cv2.face.LBPHFaceRecognizer_create()
detector
cv2.CascadeClassifier("haarcascade_frontalface_default.xml");

# function to get the images and label data
def getImagesAndLabels(path):
    imagePaths = [os.path.join(path,f) for f in
    os.listdir(path)]
    faceSamples=[]
    ids = []
    for imagePath in imagePaths:
        PIL_img = Image.open(imagePath).convert('L') # convert
    it to grayscale
        img_numpy = np.array(PIL_img,'uint8')
        id = int(os.path.split(imagePath)[-1].split(".")[1])
        faces = detector.detectMultiScale(img_numpy)
        for (x,y,w,h) in faces:
            faceSamples.append(img_numpy[y:y+h,x:x+w])
            ids.append(id)
    return faceSamples,ids

print ("\n [INFO] Training faces. It will take a few seconds.
    Wait ...")
```

```
faces,ids = getImagesAndLabels(path)
recognizer.train(faces, np.array(ids))

# Save the model into trainer/trainer.yml
recognizer.write('trainer/trainer.yml') # recognizer.save()
    worked on Mac, but not on Pi

# Print the numer of faces trained and end program
print("\n [INFO] {0} faces trained. Exiting
    Program".format(len(np.unique(ids))))
```

=

Confirm if the PIL library installed on my Rpi. If not,I will run the below command.

```
pip install pillow
```

I will use as a recognizer, the LBPH (LOCAL BINARY PATTERNS HISTOGRAMS) Face Recognizer, included on OpenCV package.

```
recognizer = cv2.face.LBPHFaceRecognizer_create()
```

The function "getImagesAndLabels (path)", will take all photos on directory: dataset, returning 2 arrays: Ids and faces. With those arrays as input, I will train my recognizer

```
recognizer.train(faces, ids)
```

Recognizer will make a prediction reestablishing its id and a record, demonstrated how sure the recognizer is with this match. Now “Recognizer”

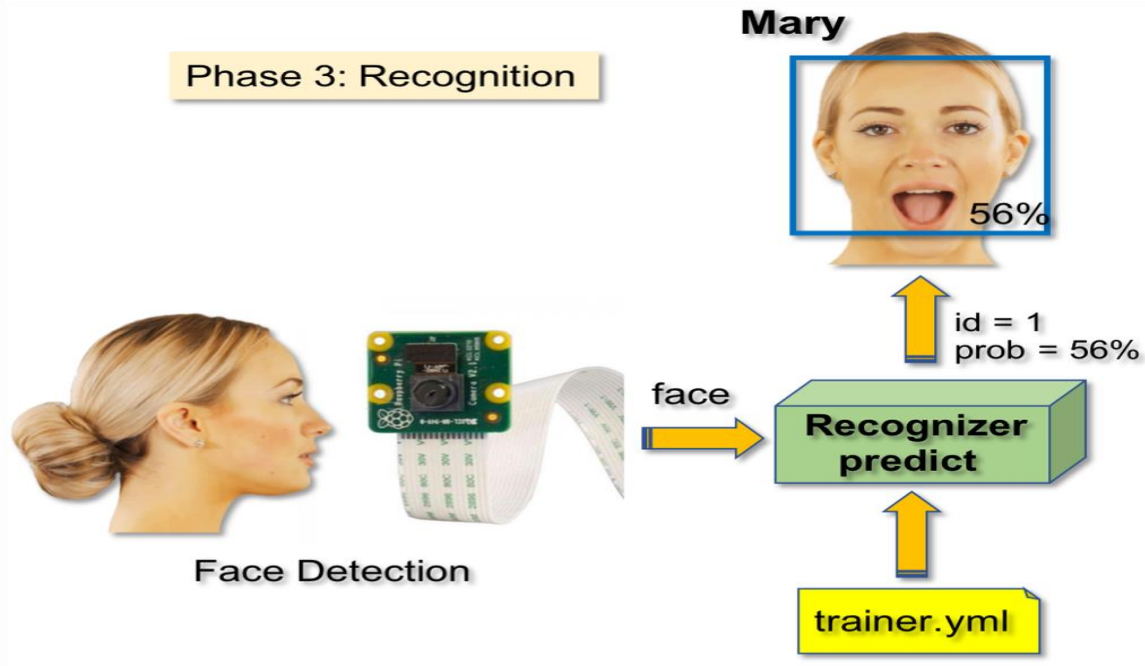

Fig.3.38 - Phase of recognition

Presently the last period of my undertaking. Here, I will catch a new face on my camera and if this individual had his face caught and prepared previously, my recognizer will make a \"prediction\" restoring its id and a file, indicated how sure the recognizer is with this match.Presently the last period of my undertaking. Here, I will catch a new face on my camera and if this individual

```
import cv2
import numpy as np
import os

recognizer = cv2.face.LBPHFaceRecognizer_create()
recognizer.read('trainer/trainer.yml')
cascadePath = "haarcascade_frontalface_default.xml"
faceCascade = cv2.CascadeClassifier(cascadePath);

font = cv2.FONT_HERSHEY_SIMPLEX

#iniciate id counter
id = 0

# names related to ids: example ==> Marcelo: id=1,  etc
names = ['None', 'Marcelo', 'Paula', 'Ilza', 'Z', 'W']

# Initialize and start realtime video capture
cam = cv2.VideoCapture(0)
cam.set(3, 640) # set video widht
cam.set(4, 480) # set video height

# Define min window size to be recognized as a face
minW = 0.1*cam.get(3)
minH = 0.1*cam.get(4)

while True:
    ret, img =cam.read()
    img = cv2.flip(img, -1) # Flip vertically
    gray = cv2.cvtColor(img,cv2.COLOR_BGR2GRAY)

    faces = faceCascade.detectMultiScale(
        gray,
        scaleFactor = 1.2,
        minNeighbors = 5,
```

```
        for(x,y,w,h) in faces:
                cv2.rectangle(img, (x,y), (x+w,y+h), (0,255,0), 2)
                id, confidence = recognizer.predict(gray[y:y+h,x:x+w])

                # Check if confidence is less them 100 ==> "0" is
            perfect match
                if (confidence < 100):
                    id = names[id]
                    confidence = "  {0}%".format(round(100 -
            confidence))
                else:
                    id = "unknown"
                    confidence = "  {0}%".format(round(100 -
            confidence))

                cv2.putText(img, str(id), (x+5,y-5), font, 1,
            (255,255,255), 2)
                cv2.putText(img, str(confidence), (x+5,y+h-5), font, 1,
            (255,255,0), 1)

            cv2.imshow('camera',img)

            k = cv2.waitKey(10) & 0xff # Press 'ESC' for exiting video
            if k == 27:
                break

        # Do a bit of cleanup
        print("\n [INFO] Exiting Program and cleanup stuff")
        cam.release()
        cv2.destroyAllWindows()
```

I am including here another exhibit, so I will show \"names\", rather than numbered ids

```
names = ['None', 'Mohamed', 'Ira', 'Ilza', 'Z', 'W']
```

So, for example: Mohamed will the user with id = 1; Iryna: id=2, etc.

Next, we will detect a face, same we did before with the haasCascade classifier. Having a detected face we can call the most important function in the above code:

```
id, confidence = recognizer.predict(gray portion of the face)
```

The recognizer.predict (), will take as a parameter a captured portion of the face to be analyzed and will return its probable owner, indicating its id and how much confidence the recognizer is in relation with this match.

And at last, if the recognizer could predict a face, we put a text over the image with the probable id and how much is the "probability" in % that the match is correct ("probability" = 100 - confidence index). If not, an "unknow" label is put on the face.

I obtained the results in the picture bellow.

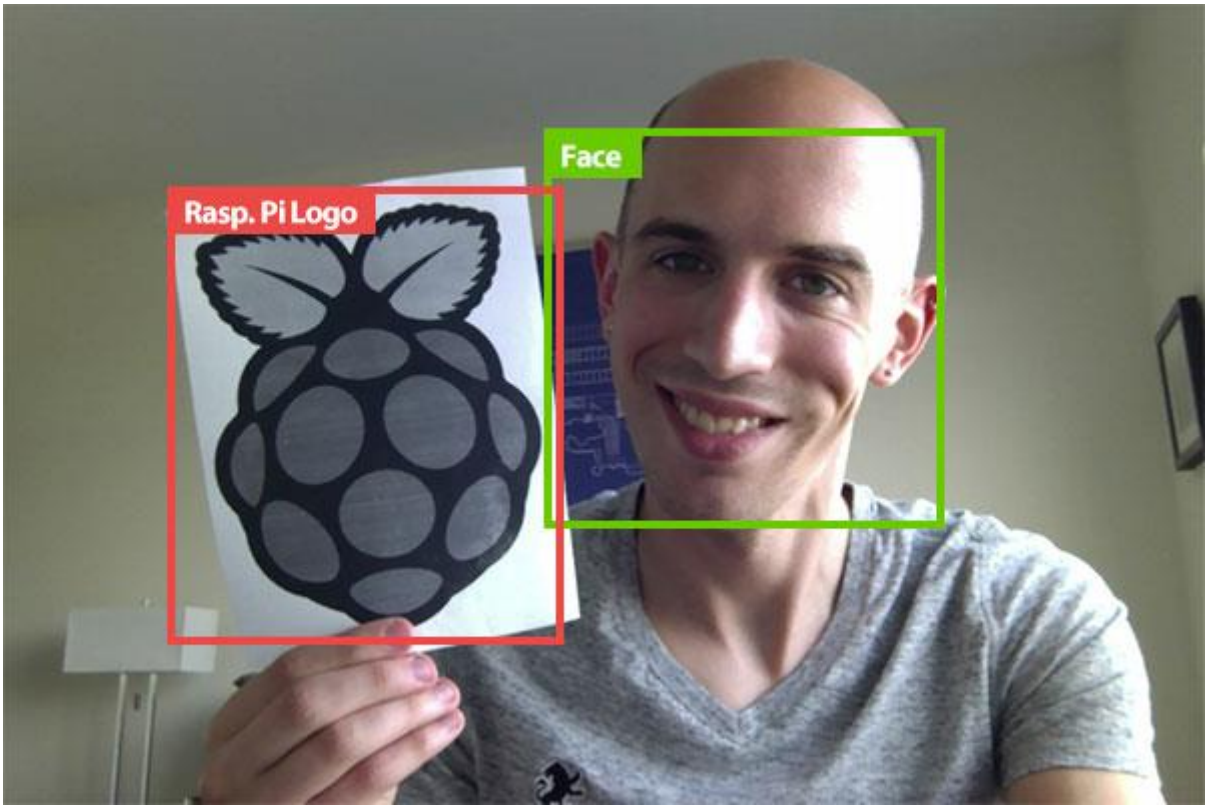

Fig.3.39 - Photo with the result

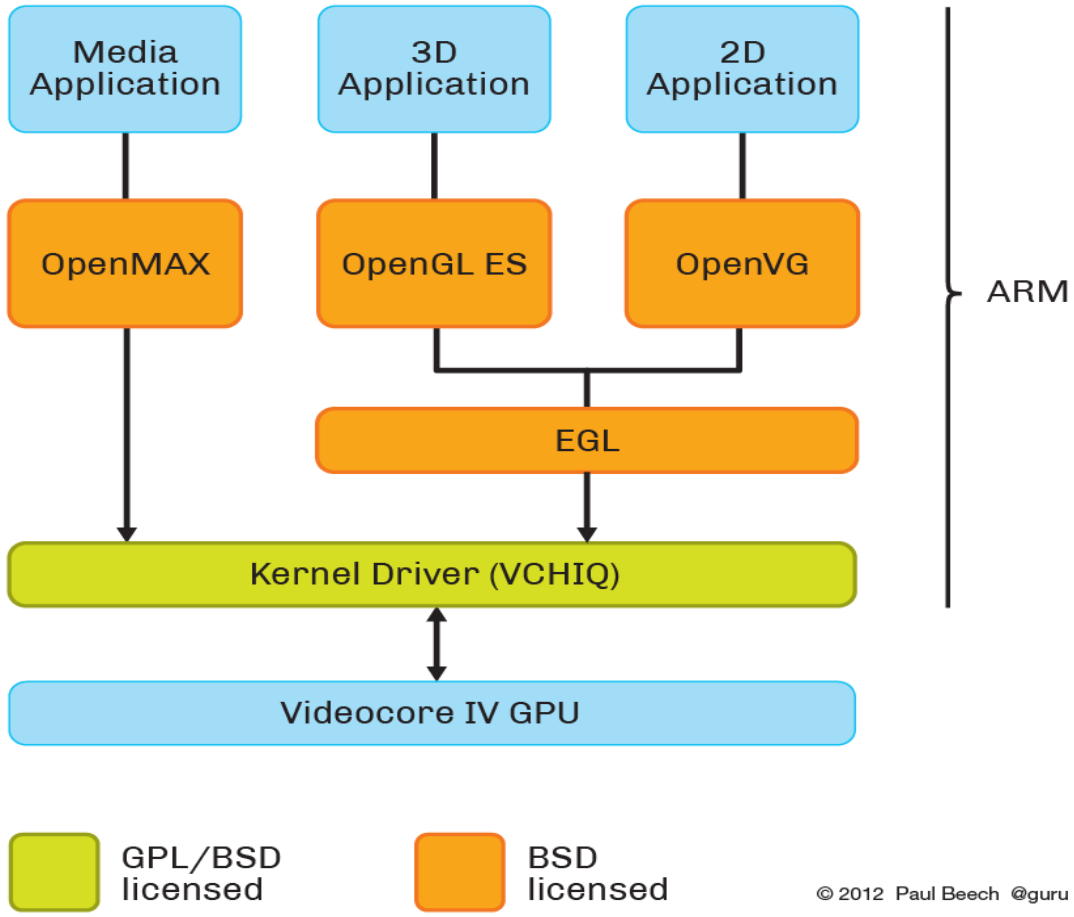

Fig.3.40 –Shematic of the Raspberry Pi Software Architecture

4. OCCUPATIONAL SAFETY AND HEALTH

With usage beyond comprehension and preference beyond inscription, computers are one of the greatest inventions of humans. Owing to high precision, speed, accuracy, regular updating of information and presence of world wide web of knowledge of current and historical concepts, computers do play an indispensable part in the lives of about 1 billion people, who have been reported to be using them daily throughout the world. There was a time when there was minimal role of technology in public health and physicians only were the sole caretakers and saviors of people. Nowadays, technology is at its peak and there is a boom in the availability of the computers to an extent that nearly 500 million computers are available throughout the world. No field today is uninfluenced by computers, and so is public health, which has benefitted through the pioneering concept of public health informatics i.e., systematic application of information, computer science, and technology to public health practice, research, and learning, which eventually facilitates transmission of data from healthcare staff to local health agencies, then to state health agencies and finally to National Center of Disease Control leading to formation of Public Health Information Network (PHIN).

Presently, computers play a crucial role by providing care in all aspects of health. Apart from the formation and maintenance of patient's records, they play a judicious role in public health surveillance. On one hand, computers are the key functionaries of health management information system and on the other hand, they are key handlers of geographic information system, electronic medical records, bioengineering, education, and research. They are also being used in statistical analysis of various data and hence, play an innovative role in leveraging the quality standards of public health professionals and workers.

Disease Surveillance: Disease surveillance projects are computer-based surveillance systems, which are used for rapid transmission and analysis of morbidity data. These are used to monitor the progress of ongoing disease control programs at national level. Initiated in the United States (US) in 1992 and in India in 2004, these aimed to detect early warning signals of impending outbreaks and help initiate an effective response in a timely manner.

In these projects, weekly reports are prepared by reporting units (using computers) on every Monday, which includes data of the previous week beginning from Sunday and ending on Saturday. These data are transmitted by state health department computers to central surveillance units through electronic mail (e-mail) facility (MINET) for rapid analysis and display of maps and demographic and epidemiological characteristics of diseases for prevention of outbreaks.

Telemedicine: The World Health Organization (WHO) defines telemedicine as "The delivery of healthcare services, where distance is a critical factor, by all healthcare professionals using information and communication technologies for the exchange of valid information for diagnosis, treatment and prevention of disease and injuries, research and evaluation and for the continuing education of healthcare providers, all in the interests of advancing the health of individuals and their communities.

Health Management Information System (HMIS): HMIS is the key component of any health program and is defined as a tool that helps in gathering, aggregating, analyzing, and using the information generated for taking actions to improve performance of health systems.

So I made a list of healthy habits while working at a computer:

Get up and stretch regularly. Do you ever go into one of those work-time-warps and look up from your computer screen 5 hours later? Setting a timer in your calendar at hourly intervals can be a really great solution, even if that mini-break only means putting the kettle on, you're still encouraging yourself to get up from the screen for a few moments, move around and think about something else.

Drink plenty of water. Now, I hate to admit it but common sense might be onto something here. We are made up of 70% water after all so it's hard to expect our brains and bodies to be in tip-top shape if their constantly dehydrated! As a reminder to drink more (water… not gin) try having a big glass of water next to your desk when you sit down to work...

Drink herbal tea rather than caffeinated drinks. If you know me then you'll know that I am a coffee and tea connoisseur, however the last 6 months I've been switching over to mainly decaf and I tell you, it really can make a huge difference to your concentration and energy levels. Many of the women on Team Shine also try to drink less caffeine and stick to herbal tea or decaf and have noticed it makes a difference too.

Starting off the day with a walk. I may be the least physically active person I know, so I try to start each work day by leaving the house and going for a stomp.

Writing an end of day review specifically for my work day. Call it an 'end of day review' or just a scribble in your diary, writing a few words about what you are grateful to yourself for today can be a really supportive practice. Here at Team Shine love to emphasise what we're grateful for and have found it to be so supportive when it's so easy to spend all our time dwelling on the negative / sh*t you didn't get done that day!

Getting a screen that's the right height for you. Do you find yourself slumping over or rounding your shoulders when you work? I had the worst posture but since I got a laptop stand to prop up my laptop it has stopped my neck being at such an awkward angle, highly recommend! I found it so supportive that now all of Team Shine have laptop stands for their work and we love using them achieve the correct height for our posture. It is affordable, portable, lightweight, easy to adjust, comes in multiple colours (including gold! If only they had a pink sequin one…

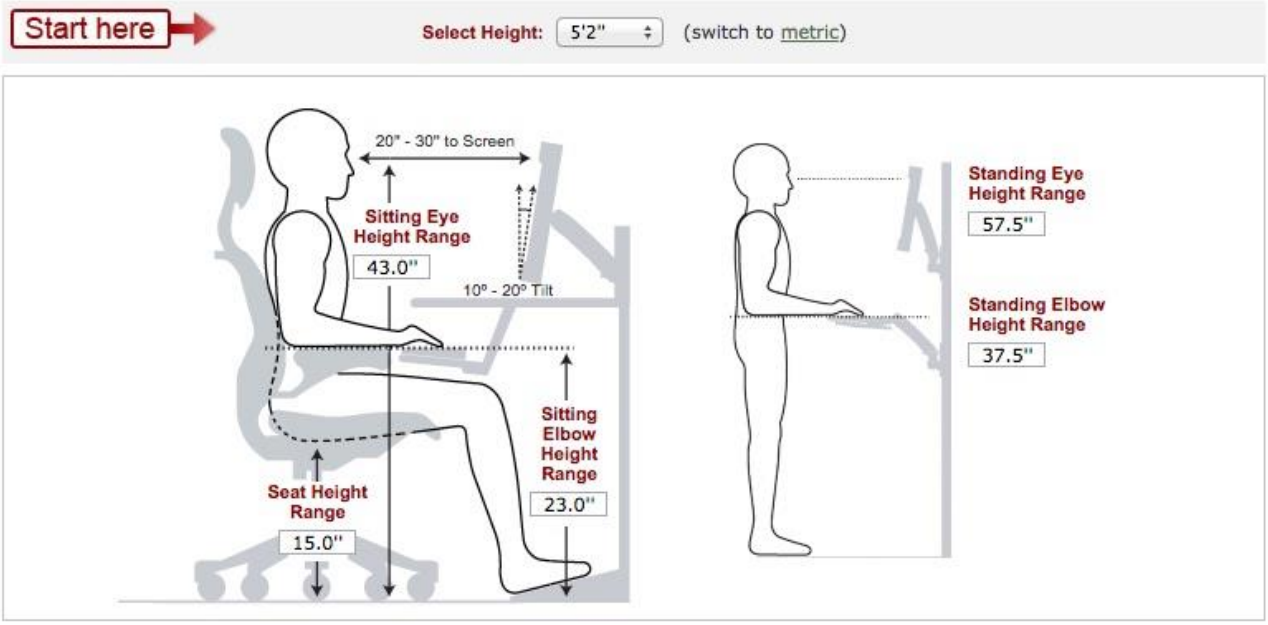

Fig.4.41 -Image to show good posture at your desk

Get a proper chair (your bed doesn't count) We all know the temptation of not getting out of bed, propping up some pillows and working in your PJ's but it really is the worst thing to do for your posture.

Using an alternative/body-friendly desk and chair

There are so many different ways that you can work, and often it can be nice to be able to switch things around to give your body a break from one position. At Team Shine you can find us working at standing desks or sit down desks, alternating chairs, sitting on a meditation cushion and a low table or even the occasional yoga ball! It's all about switching things up so that your muscles don't get worked too hard in one position.

Your hands need looking after too. Whereas some people might be terrified of clowns and buttons (seriously, look it up, it's a thing!), I'm terrified of carpal tunnel and RSI (repetitive strain injuries). Investing in a cool ergonomic mouse, for example this one named the 'Penguin'is a perfect way to support our wrists.

Listening to ambient music whilst you work.

Depending on the work, I like to listen to meditation talks or classical film soundtracks on Spotify, it really helps and is totally worth the few minutes you might 'waste' Googling the film trivia.

Take care of your feet/knees/hips and heating! I don't have an adjustable office chair, so will often put a rolled up blanket under my feet to keep good hip/knee/foot alignment and I love to have a hot water bottle in the mix when it is even a touch chilly.

Remember to streeeeetch. Going back to our suggestion earlier of taking short timed breaks, here is a program that you can use on your computer that will remind you to take a 1min break with stretches and movements – exercise videos included – it's very funny, and you can set the timing to suit your needs which is really helpful!

Go outside for a few minutes. If you're feeling like you need a breath, just do it! Go outside or open your windows and get some fresh air in your lungs. When anxiety, stress or tension comes up this can be a really great way to just refresh. We love to do this at Team Shine, especially when it feels like you really really need to hurry and get something done, these can be the best times just to take a moment, and breath.

Look outside the window. I have my table positioned so that I can look out the window whenever I can, this helps to relax the eye, as by gazing into the sky you will relax the muscles that continue to strain while looking at the screen. Being a global team that loves to travel, there are always different seasons and different countries to look around at whilst we're working!

Go chat with someone in real life. I find that the best breaks – along with the dancing – can be catchups with friends. Real life connections really do bring fresh air into the brain and sometimes even new ideas and solutions!

Rely on your support network speaking of chatting, invest in those who invest in you and by that I mean those group chats with your besties, your family, your partner, whoever it is that really gets you, share. We aren't designed to try and exist on our own, and (shout out to my fellow singles) you don't need to.

Make time to switch off. Of course at the end of a busy day you want to sit and zone out to Netflix but what I've noticed really helps me is only having one screen at a time near me, so if I'm watching TV, I don't scroll through my phone.

If I’m on Instagram, I have the TV off. It’s a simple thing but it really helps my brain, my eyes and my mental health.

Block out your time. I cannot tell you how much time I have spent in my life learning what makes me more productive. It would seem that for quite some time I thought learning about being productive was being productive and so it went on and on in a vicious circle. It did however bring me to time blocking, my true love which helps me through everything. By setting a start and finish time each day I have a much healthier relationship with working.

I know as much as anyone how much our work days can feel like a complete time-crunch, however, it really does pay off to take a few moments throughout our day to just relax and prioritise our mental and physical health as well!

CONCLUSIONS

Face acknowledgment innovation has progressed significantly over the most recent twenty years. Today, machines can consequently check character data for secure exchanges, for observation and security undertakings, and for access control to structures and so on These applications normally work in controlled conditions and acknowledgment calculations can exploit the ecological imperatives to acquire high acknowledgment precision. Nonetheless, cutting edge face acknowledgment frameworks will have far reaching application in brilliant conditions - where PCs and machines are more similar to supportive collaborators.

Face recognitiont is an arising innovation that can give numerous advantages. Face recognition can save assets and time, and even create new revenue sources, for organizations that execute it right.

It's hard to be sure. A few specialists foresee that our countenances will supplant IDs,passports and Mastercard pin numbers. Given the reality how helpful and savvy this innovation is, this expectation isn't implausible.

On the off chance that this expectation turns into a reality, any organization that actualized the innovation today may acquire an upper hand later on.

To accomplish this objective PCs should have the option to dependably distinguish close by individuals in a way that fits normally inside the example of typical human communications. They should not need uncommon communications and should adjust to human instincts about when acknowledgment is likely. This suggests that future brilliant conditions should utilize similar modalities as people, and have around similar impediments. These objectives currently show up in reach - in any case, generous exploration stays to be done in making individual acknowledgment innovation work dependably, in broadly changing conditions utilizing data from single or numerous modalities.

REFERENCES

1. A. J. Colmenarez and T. S. Huang, “Face detection and recognition,” NATO ASI Series F Computer and Systems Sciences, vol. 11, no. 2, pp. 208–218, 1998.

2 .T. Kondo and H. Yan, “Automatic human face detection and recognition under non-uniform illumination,” Pattern Recognition, vol. 32, no. 10, pp. 1707–1718, 1999

3. L. H. Koh, S. Ranganath, and Y. V. Venkatesh, “An integrated automatic face detection and recognition system,” Pattern Recognition the Journal of the Pattern Recognition Society, vol. 35, no. 6, pp. 1259–1273, 2002.

4. S. Chaudhry and R. Chandra, “Face detection and recognition in an unconstrained environment for mobile visual assistive system,” Applied Soft Computing, vol. 53, pp. 168–180, 2017.

5. M. H. Siddiqi, R. Ali, A. M. Khan, E. S. Kim, G. J. Kim, and S. Lee, “Facial expression recognition using active contour-based face detection, facial movement-based feature extraction, and non-linear feature selection,” Multimedia Systems, vol. 21, no. 6, pp. 541–555, 2015.

6. S. Zhang, X. Zhu, Z. Lei, X. Wang, H. Shi, and S. Z. Li, “Detecting face with densely connected face proposal network,” Neurocomputing, vol. 284, pp. 119–127, 2018.

7. S. Madhavan and N. Kumar, “Incremental methods in face recognition: a survey,” Artificial Intelligence Review, vol. 284, no. 5, pp. 119–127, 2019.

8. Y. Zhang, Y. Huang, S. Yu, and L. Wang, “Cross-view gait recognition by discriminative feature learning,” IEEE Transactions on Image Processing, vol. 99,

9. H. Shao, S. Chen, J. Zhao, W. Cui, and Y. U. Tianshu, “Face recognition based on subset selection via metric learning on manifold,” Frontiers of Information Technology & Electronic Engineering, vol. 16, no. 12, pp. 102–118, 2015.

10. A. K. Bobak, A. J. Dowsett, and S. Bate, “Solving the border control problem: evidence of enhanced face matching in individuals with extraordinary face recognition skills,” PLoS One, vol. 11, no. 2, Article ID e0148148, 2016.

11. Z. Lu, X. Jiang, and A. Kot, “Feature fusion with covariance matrix regularization in face recognition,” Signal Processing, vol. 144, pp. 296–305, 2018.

12. B. Samik and D. Sukhendu, “Mutual variation of information on transfer-CNN for face recognition with degraded probe samples,” Neurocomputing, vol. 310, pp. 299–315, 2018.

13. A. Rikhtegar, M. Pooyan, and M. T. Manzuri-Shalmani, “GA-optimized structure of cnn for face recognition applications,” IET Computer Vision, vol. 10, no. 6, pp. 559–566, 2016.

14. Y. X. Yang, C. Wen, K. Xie, F. Q. Wen, G. Q. Sheng, and X. G. Tang, “Face recognition using the SR-CNN model,” Sensors, vol. 18, no. 12, p. 1, 2018.

15. Computer-associated health complaints and sources of ergonomic instructions in computer-related issues among Finnish adolescents: A cross-sectional study

16.Paula T Hakala, Lea A Saarni, Ritva L Ketola, Erja T Rahkola, Jouko J Salminen & Arja H Rimpelä BMC Public Health volume 10, Article number: 11 (2010)

Appendix A.

Technical Task

MINISTRY OF EDUCATION AND SCIENCE OF UKRAINE
Ternopil Ivan Puluj National Technical University
Faculty of Computer Information Systems and Software Engineering

Computer Systems and Networks Department

“Approved”
Head of department
_____ Osukhivska H.M.
“___” _____2021

EDUCATIONAL GAME BASED ON ARDUINO CONTROLLER

TECHNICAL TASK

Degree Bachelor

“AGREED”	**“PERFOMER”**
Supervisor	Student of group ICI-43
_____ Phd.,Assoc. Prof. Lutskiv.A	_____ Aboulfadel Mohamed.
“____” _____ 2021	“____” _____ 2021

Ternopil 2021

1. Terms

Thisdocument describes tasks for development of image processing by porting the Opencv library to the Raspberry Pi board. Algorithm for face recognition. Main objective of the diploma project is to build a smart doorbell with raspberry Pi. Project will be based on Raspberry Pi3 B+.

This project is mainly based on image processing by porting the Opencv library to the Raspberry Pi board. Algorithm for face recognition

Full name of the diploma project: «smart door-bell with real-time face recognition».

Identification: CSDP 123.012.00.00

1.1. Order for system development

Order(№ _____ від ___.___.20__ р.).

1.2. Performer

Performer – student of ICI-43 group, department of computer systems and networks, Ternopil Ivan Puluj National Technical University, Aboulfadel Mohamed.

1.3. Input documents for system development

- specification of Raspberry pi;
- specification of Raspberry Pi camera;
- specification Pam Amplifier;
- specification of Raspbian stretch;
- specification of OpenCV;
- Documentation of Raspberry pi3 B+.

A. Interfacing of camera module to capture live face image.
B. Create a database of authorized person.
C. Capture current face, save it and compare with database.
D. Interface relay as output module

1.4. Date of start and submitting

Planning date of start – 17.09.2021.
Submitting date – 28.01.2021.

1.5. The sequence ofresultspresentation

Project consists the lists of documentation which response to the approved requirements of computer systems and networks department. Requirements response to the standards in the field of computer engineering development (ISO Standards).

Presentation of intermediate results of the diploma project is carried out according to the schedule approved by the supervisor.

1.6. Standards and regulatory documents

– Standard ANSI/EIA/TIA 568 - “Commercial Building Telecommunications Wiring Standard” and ANSI/EIA/TIA 569 - “Commercial Building Standard for Telecommunications Path waisand Spaces”.

2. Appliance and purpose of system design

2.1. Appliance of system

The Raspberry Pi is a credit card-sized computer. The Raspberry Pi 3 Model B+ is an improved version of the Raspberry Pi 3 Model B. It is based on the BCM2837B0 system-on-chip (SoC), which includes a 1.4 GHz quad-core ARMv8 64bit processor and a powerful VideoCore IV GPU.

2.2. Objective of system design

The objective is to create a smart doror bell with face recognition for affordable budget, that might help the user at home or for security purpose.

2.3. Characteristic of design object

Raspberry pi3 B+, Pi camera

3. System's requirements

3.1. Requirements in general

3.1.1. Requirements to the system structure and system operation

The structure of the hardware is based to be simple and basic to be carried easily, but system software must use the available hardware resources as efficiently as possible through the required objectif.

The functionality and flexibility of the system is ensured by the modification of the system software of the Raspberry Pi and additional components.

3.1.2. Channels of system components communication

The consistency and the role of every component to be done, for each component to be connected and transferring the actions and getting the reactions through the system.

3.1.3. Requirements to the modes of system operation (normal mode (reliability), emergency mode)

3.1.4. Requirements to the system diagnostic

In order to diagnose the system, it must be monitored using the appropriate tools included in the relevant system software. The tools should provide an easy interface for viewing diagnostic events and monitoring the program execution process.

3.1.5. Perspective of modernization

The system software can be modified to newer versions.

3.1.6. Requirements to the end users and their qualification

System administrators maintain the system in automatic or manual mode through management and monitoring. The minimum number of service personnel is one person.

3.1.7. Criteria of appliance

The system must be able to scale:

- by productivity;
- by capcity of information process;
- Scaling capabilities must be provided by the basic software and hardware used.

3.1.8. Reliability requirements

The system must be operational and restored in the following situations:

- if the power system of the hardware operating system fails, causing a reboot;
- if a hardware operation error occurs (except for data carriers and programs), entrust the restoration of system functions to the OS;
- For errors related to software (operating system and device drivers). In order to protect the equipment against overvoltage and switching disturbances, use network filters and uninterruptible power supplies.

3.1.9. Safety requirements

The external elements of the technical measures of the system, which are under voltage, must have protection against accidental contact, and the technical measures themselves must have a zeroing or protective grounding GOST 12.1.030-81 and PUE. The power supply system must provide a protective switch during overloads and short circuits in the load circuits, as well as manual emergency shutdown. General fire safety requirements must comply with the standards for household electrical equipment. In the event of fire, no poisonous gases or vapors should be produced. After disconnecting the power supply, ensure that all fire extinguishers can be used. Harmful factors should not exceed the standards of SanPiN 2.2.2./2.4.1340-03 of 06/03/2003.

3.1.10. Requirements for operation, maintenance, repair and storage of system components

The microclimate in rooms with the corresponding hardware has to correspond to norms of an industrial microclimate on (GOST 12.1.005-88).

For normal operation of the network it is necessary to support (according to GOST 23.865-85):

- air temperature in the range from + 15C to + 20C;
- relative humidity at 20C in the range from 30% to 70%;
- atmospheric pressure 760 mm Hg.

The technical means used must be regularly maintained according to the requirements of the technical documents, but not less than once a year. Regular maintenance and testing of technical means should include maintenance and testing of all used means, including workstations, servers, cable systems and network equipment, and uninterrupted power supplies. According to the test results of technical means, the reasons for the defects should be analyzed and eliminated. The location of the premises and its equipment must prevent uncontrolled entry by outsiders and ensure the security of confidential documents located in these premises and technical means.

3.1.11. Requirements to standardization and unification

Compatible with common computer interfaces.

The Pi can be run as a budget desktop, providing you have the patience.

- I CAN BROWSE THE WEB USING THE RASPBERRY PI 3
- I CAN USE THE RASPBERRY PI 3 AS A MEDIA CENTER
- THE RASPBERRY PI 3 RUN PS1, N64, SNES, NES AND OTHER CLASSIC CONSOLE GAMES
- Raspberry Pi is Lighter, Smaller and Efficient.
- It is known that it consume low power (5W) even less.
- Supports Full HD video (1080p), Multiple USB Ports etc.
- Smartcard swapping, alcohol detection and agriculture humidity sensing etc.

The Pi camera module is a portable light weight camera that supports Raspberry Pi. It communicates with Pi using the MIPI camera serial interface protocol. It is normally used in image processing, machine learning or in surveillance projects.

3.2. Requirements for types of collateral

3.2.1. Requirements to the system's hardware (technical characteristics of each devices in the system)

A Raspberry Pi is a general-purpose computer, usually with a Linux operating system, and the ability to run programs. It is also low cost (($25- RS 1550) for model B/B+.

1. RaspberryPi with low power consumption and the ability to control hardware.
2. USB cable for PC connection or any power adapter for the Raspberry Pi.
3. Raspberry Pi camera for the photos and videos.
4. PAM implifier for sound

3.2.2. Structure and contest of design system

The composition and content of system design work includes: (translate)

- design and coordination of the technical task for the system;
- system design;
- writing an explanatory note;
- design of graphic material;
- Defense of the qualifying paper.

4. Technical and economic indicators

The cost of development should not exceed 2500 UAH.

The service life of the device must be at least 18,000 thousand hours. (2 years)

* Note: the cost of development may change during the calculation during development.

5. Stages of system design

Table 1 - Stages of system design

Number of stage	Stage	Duration
1	Development and approval of the technical task	16.09-20.09.2020
2	Analysis of the technical task	21.09-21.10.2020
3	Substantiation of possible technical solutions	24.10-30.12.2020
4	System design and implementation	02.01-12.01.2021
5	Testing of the designed system	13.01-18.01.2021
7	Section of labor protection and safety in emergency situations	18.01-22.01.2021
9	Registration of the qualifiying paper	23.01-27.01.2021
10	Preliminary defense of the qualifiying paper	28.01.2021
11	Defense of the qualifiying paper	28. 01.2021

6. The order of control and acceptance

The control of the process of execution of the diploma project is carried out by the head of the diploma project.

Normocontrol of the diploma project for compliance with the requirements of the standards is carried out at the Department of Computer Systems and Networks.

The presentation of the results of the diploma project is done by defending the diploma project at the relevant meeting of the SEC, illustrating the main achievements through the graphic material.

7. Requirements for documentation

The documentation must meet the requirements of ESKD and DSTU

Set of design documentation:

– explanatory note;
– applications;
– graphic material:

a) wiring diagrams of the device board through specialized interfaces;

b) block diagram of the device components;

c) algorithms of the created software;

d) block diagram of the device software;

e) the deployment scheme of this solution.

* Note: The design documentation may be subject to change and addition during development.

8. Additional conditions

During the implementation of the thesis project, changes and additions may be made to this technical task.