A
Project Report
On

Facial Recognition Smart Glasses for Visually Challenged People

Submitted to

Sant Gadge Baba Amravati University, Amravati

Submitted in partial fulfillment of

the requirements for the Degree of

Bachelor of Engineering in

Electronics and Telecommunication Engineering

Submitted by

Abrar Khan Aazam Khan Pathan	**Vaibhav Deshmukh**
(PRN: 203120443)	(PRN: 193120333)
Hrituja Chandane	**Bhargavi Deshmukh**
(PRN: 193120261)	(PRN: 193120245)

Under the Guidance of
Prof. D. L. Bhombe
Associate Professor & Dean Academics

Department of Electronics & Telecommunication Engg.
Shri Sant Gajanan Maharaj College of Engineering,
Shegaon – 444 203 (M.S.)
2022-2023

Department of Electronics & Telecommunication Engineering

Shri Sant Gajanan Maharaj College of Engineering,

Shegaon – 444203, Maharashtra, India

(Recognized by AICTE, Accredited by N.B.A, New Delhi)

Certificate

This is to certify that the project report entitled **"Facial Recognition Smart Glasses for Visually Challenged People "** is hereby approved as a creditable study carried out and presented by

Abrar khan Aazam Khan Pathan (PRN: 203120443)

Hrituja Chandane (PRN: 193120261)

Bhargvi Deshmukh (PRN: 193120245)

Vaibhav Deshmukh (PRN: 193120333)

in a manner satisfactory to warrant of its acceptance as a pre-requisite in a partial fulfillment of the requirements for the degree of Bachelor of Engineering in Electronics & Telecommunication Engineering of Sant Gadge Baba Amravati University, Amravati during the Session 2021-22.

Prof. D. L. Bhombe
Project Guide

Prof. Ashwini Deshmukh
Internal Examiner

External Examiner

Dr. M. N. Tibdewal
Professor & Head, E & TC Dept.

Dr. S. B. Somani
Principal

ABSTRACT

There are millions of disabled people in this world who are always in need of helping hands. People with visual impairment face many challenges in their daily life as modern assistive devices often do not meet consumer requirements in terms of price and level of assistance. Here we propose a new design of assistive smart glasses for visually impaired persons. The aim of the project of Blind assistance is to promote a wide challenge in computer vision such as recognition of persons of the surrounding practice by the blind on a daily basis. A dataset of persons gathered from daily scenes is generated to apply the required recognition. The camera is used to detect any person and the main object of the project is to design and implement real-time object recognition using blind glass.

Keywords : OpenCV, Haar cascade algorithm, Raspberry pi, database, camera, audio feedback device.

ACKNOWLEDGMENT

We would like to take this opportunity to express our heartfelt thanks to our guide Prof. D. L. Bhombe for his esteemed guidance and encouragement, especially through difficult times. His suggestions broaden our vision and guided us to succeed in this work. We are also very grateful for his guidance and comments while designing part of our project and learnt many things under his leadership. Also, we would like to thank to Dr. M. N. Tibdewal, Head of Electronics and Telecommunication Department, all teaching and non-teaching staff of EXTC Department for their encouragement and suggestions for our project.

We extend our thanks to Dr. S. B. Somani, Principal, Shri Sant Gajanan Maharaj, College of Engineering, for his valuable support.

We sincerely thank to all our friends, who helped us directly or indirectly in completing our project work. We would like to express our appreciation for the wonderful experience while completions of this project work.

Abrar Khan Aazam Khan Pathan

Hrituja Chandane

Bhargavi Deshmukh

Vaibhav Deshmukh

ABBREVIATIONS

GPS:	Global Positioning System
USB:	Universal Serial Bus
SBC:	Single Board Computers
SOC:	System on Chip
GPU:	Graphics Processing Unit
CPU:	Central Processing Unit
POE:	Power of Ethernet
IOT:	Internet of Things
ARM:	Advanced RISC Machine
RAM:	Random Access memory
NVDA:	Nonvisual Desktop Access
HDMI:	High-Definition Multimedia Interface

LIST OF FIGURES & TABLE

Figure 4.1.1 Block Diagram

Figure 4.2.1 Flow Chart

Figure 5.1.1 Raspberry Pi

Figure 5.1.2 Raspberry Pi 400 Kit

Figure 5.2.1 Raspberry Pi Pico Block Diagram

Figure 5.3.1 Processor

Figure 5.4.1 ESP32 Camera Module

Figure 5.5.1 Pin out Diagram of Push Button

Figure 6.1.1 Area of Haar-like Features

Figure 6.2.1 Cascade Classifier for Face Detection

Figure 8.1 Live Demo of Camera Module

Table 2.1 Literature Survey

CONTENTS

Abstract *i*

Acknowledgment *ii*

Abbreviations *iii*

List of Figures *iv*

Contents *v*

1. INTRODUCTION 1

2. LITERATURE SURVEY 6

3. OBJECTVES 8

4. RESEARCH METHODOLOGY 9

4.1 Proposed Syatem 9

4.2 Flow Chart 10

4.3 Working 11

5. SYSTEM REQUIREMENTS 12

5.1 Raspberry Pi 13

5.2 Raspberry Pi Pico 15

5.3 Processor 17

5.4 ESP32 camera 18

5.5 Push Button Switch 19

5.6 Python IDE 20

6. IMPLEMENTATION 22

6.1 Haar Algorithm 22

6.2 Cascade classifier 23

6.3 Libraries 24

6.3.1 eSpeak 24

6.3.2 Open CV 25

6.3.3 OS 25

7. CODE 26

7.1 Code for Recognition 26

7.2 Code for Training 31

8. RESULT & DISCUSSION 34

9. CONCLUSION & FUTURE SCOPE 35

References 36

1. INTRODUCTION

People with visual impairment face various problems in their daily life as the modern assistive devices are often not meeting the consumer requirements in terms of price and level of assistance. This project presents a new design of assistive smart glasses for visually impaired persons. The objective is to assist in multiple daily tasks using the advantage of the wearable design format. The aim of the project of Blind assistance is to promote a wide challenge in computer vision such as recognition of persons of the surrounding practiced by the blind on a daily basis. The camera is placed on a blind person's glasses. A dataset of persons gathered from daily scenes is created to apply the required recognition. The camera is used to detect any person. The proposed method for the blind aims at expanding possibilities to people with vision loss to achieve their full potential. The main object of the project is to design and implement a real-time object recognition using blind glass. According to NCBI (1986), 1.5% of the population in Saudi Arabia is blind and another 7.8% have vision difficulties. These people need some help to make their life easier and better. The goal of “Smart Glasses' ' is to assist blind people and people who have vision problems by introducing a smart and new technology that makes them able to discover the object. This project presents a new design of assistive smart glasses for visually impaired persons. The target is to help in multiple daily tasks mistreatment of the advantage of wearable style format. The sensors are used to detect any person or any object. The projected technique for the blind aims at increasing prospects to people with vision loss to attain their full potential. The main object of the project is to design and implement a real time object recognition using blind glass.

Background

Blindness is one of the most, if not the most, misunderstood types of disability. The general masses have their own preconceived notions about blind people that they firmly believe to be true without even getting in touch with a blind person. Most of the members of the non-blind community believe that blind people cannot do their work or live a normal life. 'My Son will not be a Beggar Be' by Ved Mehta is a perfect example of the contradiction between society's perspective and the reality of a blind person's life.

Blind people do lead normal life with their own style of doing things. But, they definitely face troubles due to inaccessible infrastructure and social challenges. Let us have an empathetic look at some of the daily life problems, struggles, and challenges faced by blind people.

Navigating Around Places

The biggest challenge for blind people, especially the ones with complete loss of vision, is to navigate around places. Obviously, blind people roam easily around their house without any help because they know the position of everything in the house. People living with and visiting blind people must make sure not to move things around without informing or asking the blind person. Commercial places can be made easily accessible for the blinds with tactile tiles. But, unfortunately, this is not done in most of the places. This creates a big problem for blind people who might want to visit the place.

Finding Reading Material

Blind people have a tough time finding good reading materials in accessible formats. Millions of people in India are blind but we do not have even the proper textbooks in braille, leave alone the novels and other leisure reading materials. Internet, the treasure trove of information and reading materials, too is mostly inaccessible for the blind people. Even though a blind person can use screen reading software but it does not make the Internet surfing experience very smooth if the websites are not designed accordingly. A blind person depends on the image description for understanding whatever is represented through pictures. But most of the time, websites do not provide clear image descriptions.

Arranging Clothes

As most the blind people depend on objects' shape and texture to identify them — arranging the laundry becomes a challenging task. Although a majority of blind people devise their own technique to recognize and arrange at least their own clothes it still is a challenging chore. This becomes a daredevil task if it's about pairing and arranging the socks. All this is because recognizing colors is almost impossible for persons with total blindness.

Overaly Helpful Individuals

It is good to be kind and help others. But overly helpful individuals often create problems for the blind person. There are lots of individuals who get so excited to help a disabled person that they forget even to ask the person whether she needs help or not. A blind person might be doing something painfully slow (from your perspective) but you should not hurry in doing the work without asking the person properly. You might end up creating some trouble for the blind person.

Getting Device To Become Independent

The most valuable thing for a disabled person is gaining independence. A blind person can lead an independent life with some specifically designed adaptive things for them. There are lots of adaptive equipment that can enable blind person to live their life independently but they are not easily available in the local shops or markets. Refreshable Braille Display is an example of such useful device. A blind person needs to hunt and put much effort to get each piece of equipment that can take them one step closer to independence. Everyone faces challenges in their life… blind people face a lot more. But, this certainly does not mean that you can show sympathy to blind persons. They too, just like any individual, take up life's challenges and live a normal life, even if it does not seem normal to the sighted individuals.

Existing System

The role of blind assistance is to promote a wide range of challenges in computer vision, including mapping and navigation. In this paper, two ultrasonic sensors are placed on a blind person's glasses to provide the necessary information about the surrounding environment. One IR sensor is used to detect human radiation in the environment. Obtaining objects is used to find objects in the real world from the image of the world such as faces, bicycles, chairs, doors, or tables that are common in blind squares. Buzzer and vibration motor is used to alert the person when sensors detect any object in front of them. We are using a NODE MCU microcontroller for this project the proposed method for the blind aims at expanding possibilities for people with vision loss to achieve their full potential. The test results reveal the performance of the proposed activity approximately in real-time.

Blind Walking Stick

Blind stick is an innovative stick designed for visually disabled people for improved navigation. The blind stick is integrated with an ultrasonic sensor along with light and water sensing. This uses ultrasonic sensors to detect obstacles ahead using ultrasonic waves. On sensing obstacles, the sensor passes this data to the microcontroller. The microcontroller then processes this data and calculates if the obstacle is close enough. If the obstacle is not that close the circuit does nothing. If the obstacle is closed the microcontroller sends a signal to sound a buzzer. It also detects and sounds a different buzzer if it where detects water and alerts the blind. The blind stick may be efficient in some cases, but it has some drawbacks as well. It may be difficult to detect the hanging obstacles which may lead to a collision.

Smart Shoes

It is a shoe that navigates the route from source to destination. Since the system is implemented in shoes, a battery is used for the power supply. Bluetooth is used to get the location coordinates from a mobile phone by using GPS set from a mobile. Need an android app for searching the route destination to the source route. Here a novel technique is proposed to assist a blind person to track routes in an efficient way. The shoes sync up with a Smartphone app that uses maps and vibrates to tell users when and where to turn to reach their destinations. When any blind person needs to go to a destination, the target is set in maps using a mobile and the Smart shoe gives the directions to the destination with the vibrations.

Face recognition

A facial recognition system is a technology capable of identifying or verifying a person from a digital image or a video frame from a video source. It is a biometric software application capable of uniquely identifying or verifying a person by comparing and analyzing patterns based on the person's facial contours. Facial recognition is mostly used for security purposes, though there is increasing interest in other areas of use. Most facial recognition systems function based on the different nodal points on a human face. The values measured against the variable associated with points of a person’s face help in uniquely identifying or verifying the person. With this technique, applications can use data captured from faces and can accurately and quickly identify target individuals. There are many advantages associated with facial recognition. Compared to other biometric techniques, facial recognition is of a non-contact nature. Face images can be captured from a distance and can be analyzed without ever requiring any interaction with the user/person. As a result, no user can successfully imitate another person.

Problem Statement

Visual impairments limit the ways the person can interact with others, access information or develop his/her own knowledge and experience, therefore a need for an assisting aid with multitasks features to cope with different situations is an important issue. - The available aids and technologies in the market nowadays are expensive to the normal or low level of income people which are the majority of the users, therefore new devices with similar tasks and cheaper prices are needed.

2. LITERATURE SURVEY

Sr.No	Author	Title	Journal	Finding
1	Abate, A. F., Nappi, M., Riccio, D., & Sabatino, G.	2D and 3D face recognition:	A survey. Pattern recognition letters,	They carried out a survey of two and three dimensional face images
2	Ding, C., Choi, J., Tao, D., & Davis, L. S.	Multi-directional multi-level dual-cross patterns for robust face recognition	IEEE transactions on pattern analysis and machine intelligence	They retrieve Multi Directional Multi-Level Dual-Cross Patterns (MDML-DCPs) from face images.
3	Ganguly, S., Bhattacharjee, D., & Nasipuri, M	llumination, pose and occlusion invariant face recognition from range images using ERFI model.	International Journal of System Dynamics Applications (IJSDA),	They performed face recognition of the 3d face images in an unconstrained environment with variations in pose, occlusion and lighting.
4	Zhu, X., Lei, Z., Yan, J., Yi, D., & Li, S. Z	High-fidelity pose and	In Proceedings of the IEEE	They presented a facial recognition

	Zhu, X., Lei, Z., Yan, J., Yi, D., & Li, S. Z	expression normalization for face recognition in the wild.	Conference on Computer Vision and Pattern Recognition	algorithm by morphing the input images to the model.
5	Schroff, F., Treibitz, T., Kriegman, D., & Belongie,S	illumination and expression invariant pairwise face-similarity measure via doppelgänger list comparison.	In Computer Vision (ICCV), 2020 IEEE International Conference on (pp. 2494- 2501). IEEE.	They retrieve Multi Directional Multi-Level Dual-Cross Patterns (MDML-DCPs) from face images.

Table 2.1 Literature Survey

3. OBJECTIVES

- To design and implement smart glasses that can be use by the visually impaired people easily, with an emphasis on cost-effectiveness.
- To prove the feasibility of the image processing techniques with the audio description as a tool helping the visually impaired people and giving them more independence in their education life.
- The objective is to assist in multiple daily tasks using the advantage of wearable design format.
- Expanding possibilities to people with vision loss to achieve their full potential.
- To design and implement a real time object recognition using blind glass.
- To create a facial recognition and audio output system using Raspberry pi and USB camera module for visually impaired persons.

4. RESEARCH METHODOLOGY

4.1 Proposed System

Our main motive for this project is to help visually impaired people, not accurately but to make their life a little bit easier and become self-dependent. In this project, the glasses we will be using would be able to take pictures via camera. "Glasses can recognize images and determine each object in the images". It can determine the distance between the blind person and each object. Conversion of captured image information into the voice will be provided to the user through headphones that help blind people to know who is in front of him/her. It will also give a notification to the user if the object is very close or far away from him/her.

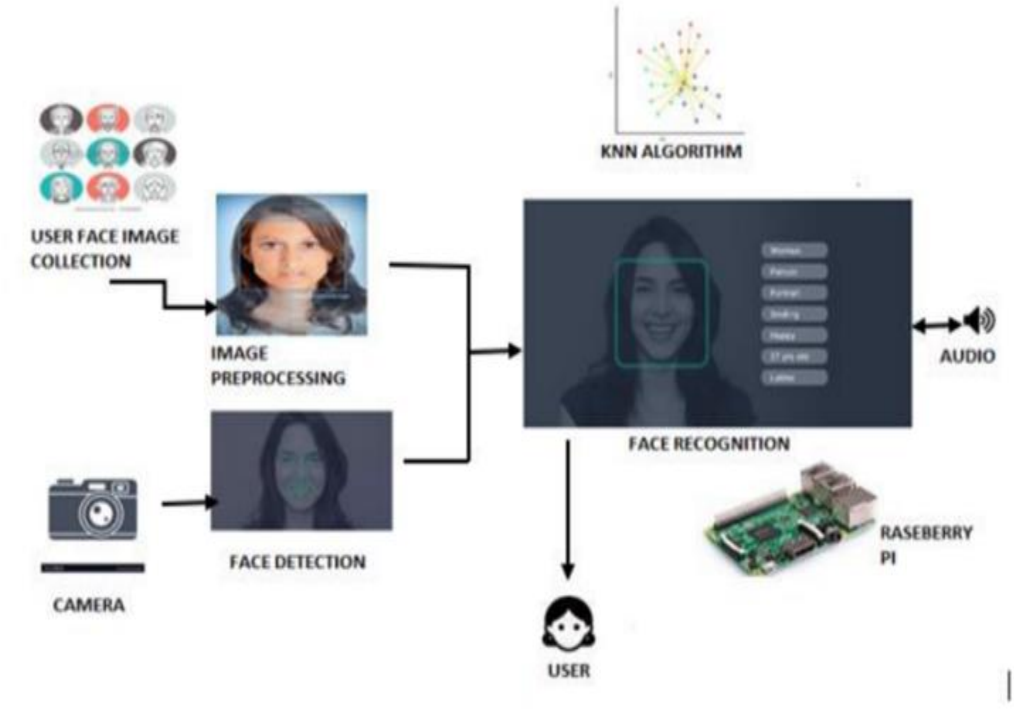


Fig 4.1.1: Block Diagram

4.2 Flow Chart

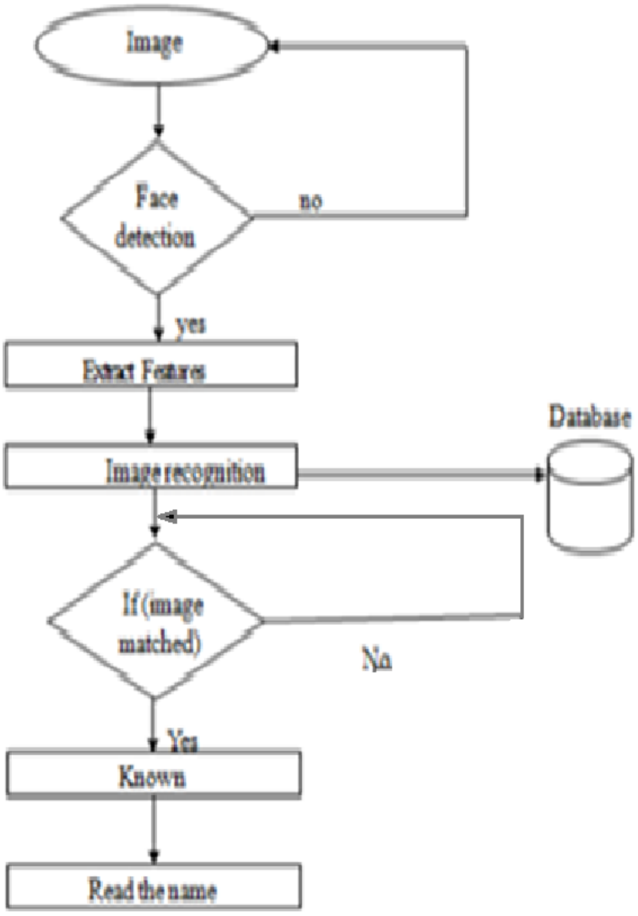


Fig 4.2.1: Flow Chart

4.3 Working

In this project we have used raspberry Pi 3B, one USB camera, and wireless earphones. We can also use wired as well as Bluetooth earphones because raspberry pi supports both the wired or wireless earphones. when we switch on the system the system will initiated and we used to run the python code in our system. once the python code starts executing the camera window gets open. In the camera, window the faces can be seen which is nothing but the real time detection window. It detects the real time faces by using the USB camera and also provides a frame/window where the names of the recognized person is shown at the left corner of the frame, at the same time window starts comparing the real time captured faces with the trained faces by using the image feature extraction method. We have used the haar method which is an algorithm that extracts the image features from the real time captured images and makes the patterns of that images and compares it with stored patterns by checking each image captured at the time of training of the faces done. In our case we have trained four faces and each face is having 50 images. The comparison process reffers the total 50 images of each faces and it will check for each face and after comparison it generates the result of matched image and provides the output to the controller. The controller passes the signal in terms of sound and the name of the recognized person will be provided as an output. The blind person can hear their name by using Bluetooth headphones. This Recognition is done by using python code with the help of Raspberry pi. This leads to result in recognition of the person who is standing infront of the blind person and the blind person can easily recognize that person by hearing their name. We have implemented this project for home, Offices and limited use only where blind person can have the option to recognize the familiar faces only.

5. SYSTEM REQUIREMENT

Hardware Requirement

1. Raspberry Pi
2. SD card
3. Goggle
4. USB Camera
5. Push button
6. Earphones

Software Requirement

1. Python IDE 3.8 version
2. Machine Learning

5.1 Raspberry Pi

Fig 5.1.1: Raspberry Pi

Raspberry Pi (/paɪ/) is a series of small single-board computers (SBCs) developed in the United Kingdom by the Raspberry Pi Foundation in association with Broadcom. The Raspberry Pi project originally leaned towards the promotion of teaching basic computer science in schools and in developing countries. The original model became more popular than anticipated, selling outside its target market for uses such as robotics. It is widely used in many areas, such as for weather monitoring, because of its low cost, modularity, and open design. It is typically used by computer and electronic hobbyists, due to its adoption of HDMI and USB devices. After the release of the second board type, the Raspberry Pi Foundation set up a new entity, named Raspberry Pi Trading, and installed Eben Upton as CEO, with the responsibility of developing technology. The Foundation was rededicated as an educational charity for promoting the teaching of basic computer science in schools and developing countries. Most Pis are made in a Sony factory in Pencoed, Wales, while others are made in China and Japan.

Series & Generation

There are three series of Raspberry Pi, and several generations of each have been released. Raspberry Pi SBCs feature a Broadcom system on a chip (SoC) with an integrated ARM-compatible central processing unit (CPU) and on-chip graphics processing unit (GPU), while Raspberry Pi Pico has a RP2040 system on chip with an integrated ARM-compatible central processing unit (CPU).

6 The first generation (Raspberry Pi Model B) was released in February 2012, followed by the simpler and cheaper Model A.

7 In 2014, the Foundation released a board with an improved design, Raspberry Pi Model B+. These first generation boards feature ARM11 processors, are approximately credit-card sized and represent the standard mainline form factor. Improved A+ and B+ models were released a year later.A "Compute Module" was released in April 2014 for embedded applications.

8 The Raspberry Pi 2 was released in February 2015 and initially featured a 900 MHz 32-bit quad-core ARM Cortex-A7 processor with 1 GB RAM. Revision 1.2 featured a 900 MHz 64-bit quad-core ARM Cortex-A53 processor (the same as that in the Raspberry Pi 3 Model B, but underclocked to 900 MHz).

9 Raspberry Pi 3 Model B was released in February 2016 with a 1.2 GHz 64-bit quad core ARM Cortex-A53 processor, on-board 802.11n Wi-Fi, Bluetooth and USB boot capabilities.

10 On Pi Day 2018, the Raspberry Pi 3 Model B+ was launched with a faster 1.4 GHz processor, a three-times faster gigabit Ethernet (throughput limited to ca. 300 Mbit/s by the internal USB 2.0 connection), and 2.4 / 5 GHz dual-band 802.11ac Wi-Fi (100 Mbit/s). Other features are Power over Ethernet (PoE) (with the add-on PoE HAT), USB boot and network boot (an SD card is no longer required).

11 Raspberry Pi 4 Model B was released in June 2019 with a 1.5 GHz 64-bit quad core ARM Cortex-A72 processor, on-board 802.11ac Wi-Fi, Bluetooth 5, full gigabit Ethernet (throughput not limited), two USB 2.0 ports, two USB 3.0 ports, 1–8 GB of RAM, and dual-monitor support via a pair of micro HDMI (HDMI Type D) ports for up to 4K resolution. The version with 1 GB RAM has been abandoned and the prices of the 2 GB version have been reduced. The 8 GB version has a revised circuit board. The Pi 4 is also powered via a USB-C port, enabling additional power to be provided to downstream peripherals, when used with an appropriate PSU. But the Pi can only be operated with 5 volts and not 9 or 12 volts like other mini computers of this class. The initial Raspberry Pi 4 board has a design flaw where third-party e-marked USB cables, such as those used on Apple MacBooks, incorrectly identify it and refuse to provide power.Tom's Hardware tested 14 different cables and found that 11 of them turned on and powered the Pi without issue. The design flaw was fixed in revision 1.2 of the board, released in late 2019. In mid-2021, Pi 4 B models appeared with the improved Broadcom BCM2711C0. The manufacturer is now using this chip for the Pi 4 B and Pi 400. However, the tack frequency of the Pi 4 B was not increased in the factory.

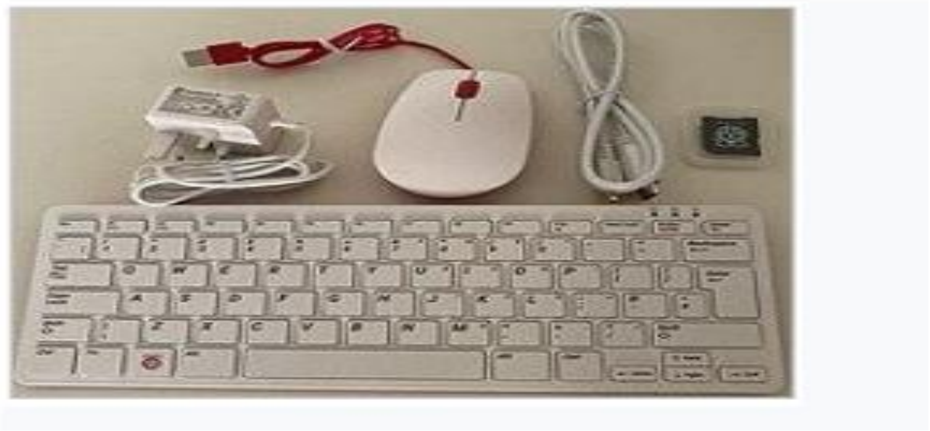


Fig 5.1.2: Raspberry Pi 400 Kit

- Raspberry Pi 400 was released in November 2020. It features a custom board that is derived from the existing Raspberry Pi 4, specifically remodeled with a keyboard attached. The case was derived from that of the Raspberry Pi Keyboard. A robust cooling solution (i.e. a broad metal plate) and an upgraded switched-mode power supply allow the Raspberry Pi 400's Broadcom BCM2711C0 processor to be clocked at 1.8 GHz, which is 20% faster than the Raspberry Pi 4 it is based on. The keyboard-computer features 4 GB of LPDDR4 RAM.

5.2 Raspberry Pi Pico

Raspberry Pi Pico was released in January 2021 with a retail price of $4. It was Raspberry Pi's first board based upon a single microcontroller chip; the RP2040, which was designed by Raspberry Pi in the UK. The Pico has 264 KB of RAM and 2 MB of flash memory. It is programmable in Micro Python, Circuit Python, C and Rust. The Raspberry Pi Foundation has partnered with Ad fruit, Pimoroni, Arduino and Spark Fun to build Accessories for Raspberry Pi Pico and variety of other boards using RP2040 Silicon Platform. Rather than perform the role of general purpose computer (like the others in the range) it is designed for physical computing, similar in concept to an Arduino.

The Raspberry Pi hardware has evolved through several versions that feature variations in the type of the central processing unit, amount of memory capacity, networking support, and peripheral-device support.

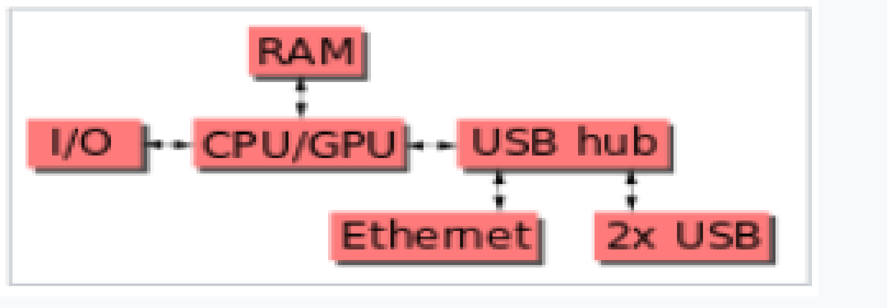


Fig 5.2.1: Raspberry Pi Pico Block Diagram

This block diagram describes models B, B+, A and A+. The Pi Zero models are similar, but lack the Ethernet and USB hub components. The Ethernet adapter is internally connected to an additional USB port. In Model A, A+, and the Pi Zero, the USB port is connected directly to the system on a chip (SoC). On the Pi 1 Model B+ and later models the USB/Ethernet chip contains a five-port USB hub, of which four ports are available, while the Pi 1 Model B only provides two. On the Pi Zero, the USB port is also connected directly to the SoC, but it uses a micro USB (OTG) port. Unlike all other Pi models, the 40 pin GPIO connector is omitted on the Pi Zero, with solder able through-holes only in the pin locations. The Pi Zero WH remedies this. Processor speed ranges from 700 MHz to 1.4 GHz for the Pi 3 Model B+ or 1.5 GHz for the Pi 4; on-board memory ranges from 256 MB to 8 GB random-access memory (RAM), with only the Raspberry Pi 4 having more than 1 GB. Secure Digital (SD) cards in MicroSDHC form factor (SDHC on early models) are used to store the operating system and program memory, however some models also come with onboard eMMC storage [48] and the Raspberry Pi 4 can also make use of USB-attached SSD storage for its operating system. The boards have one to five USB ports. For video output, HDMI and composite video are supported, with a standard 3.5 mm tip-ring-sleeve jack for audio output. Lower-level output is provided by a number of GPIO pins, which support common protocols like

I²C. The B-models have an 8P8C Ethernet port and the Pi 3, Pi 4 and Pi Zero W have on-board Wi-Fi 802.11n and Bluetooth.

5.3 Processor

Fig 5.3.1: Processor

The Raspberry Pi 2B uses a 32-bit 900 MHz quad-core ARM Cortex-A7 processor. The Broadcom BCM2835 SoC used in the first generation Raspberry Pi includes a 700 MHz ARM1176JZF-S processor, Video Core IV graphics processing unit (GPU), and RAM. It has a level 1 (L1) cache of 16 KB and a level 2 (L2) cache of 128 KB. The level 2 cache is used primarily by the GPU. The SoC is stacked underneath the RAM chip, so only its edge is visible. The ARM1176JZ (F)-S is the same CPU used in the original iPhone, although at a higher clock rate, and mated with a much faster GPU. The earlier V1.1 model of the Raspberry Pi 2 used a Broadcom BCM2836 SoC with a 900 MHz 32-bit, quad-core ARM Cortex-A7 processor, with 256 KB shared L2 cache. The Raspberry Pi 2 V1.2 was upgraded to a Broadcom BCM2837 SoC with a 1.2 GHz 64-bit quad-core ARM Cortex-A53 processor, the same one which is used on the Raspberry Pi 3, but under clocked (by default) to the same 900 MHz CPU clock speed as the V1.1. The BCM2836 SoC is no longer in production as of late 2016. The Raspberry Pi 3 Model B uses a Broadcom BCM2837 SoC with a 1.2 GHz 64-bit quad-core ARM Cortex-A53 processor, with 512 KB shared L2 cache. The Model A+ and B+ are 1.4 GHz The Raspberry Pi 4 uses a Broadcom BCM2711 SoC with a 1.5 GHz (later models: 1.8 GHz) 64-bit quad-core ARM Cortex-A72 processor, with 1 MB shared

L2 cache. Unlike previous models, which all used a custom interrupt controller poorly suited for virtualization, the interrupt controller on this SoC is compatible with the ARM Generic Interrupt Controller (GIC) architecture 2.0, providing hardware support for interrupt distribution when using ARM virtualization capabilities. The Raspberry Pi Zero and Zero W use the same Broadcom BCM2835 SoC as the first generation Raspberry Pi, although now running at 1 GHz CPU clock speed.The Raspberry Pi Zero W 2 uses the RP3A0-AU CPU, a 1 GHz 64 bit ARM Cortex A53, on 512MB of SDRAM. Documentation states this "system-on-package" is a Broadcom BCM2710A1 package, using a BCM2837 Broadcom chip as core, which is an ARM v8 quad-core. The RPi3 also uses the BCM2837, but at 1.2 GHz, since the Pi Zero W 2 clock is 1 GHz. The Raspberry Pi Pico uses the RP2040 running at 133 MHz.

5.4 ESP32 camera

The ESP32-CAM is a small-size, low-power consumption camera module based on ESP32. It comes with an OV2640 camera and provides an onboard TF card slot.

The ESP32-CAM can be widely used in intelligent IoT applications such as wireless video monitoring, WiFi image upload, QR identification, and so on.

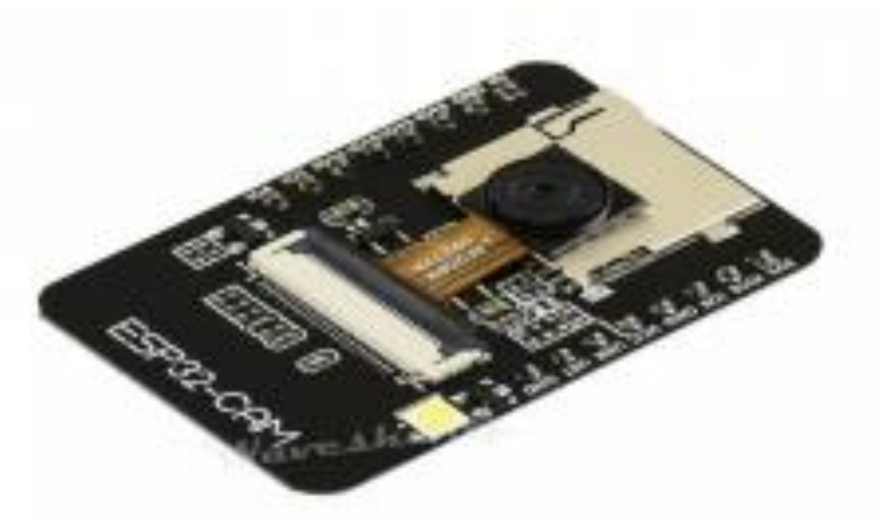


Fig 5.4.1: ESP32 Camera Module

Features

- Onboard ESP32-S module supports Wi-Fi + Bluetooth
- OV2640 camera with flash
- Onboard TF card slot, supports up to 4G TF card for data storage
- Supports Wi-Fi video monitoring and Wi-Fi image upload
- Supports multi sleep modes, deep sleep current as low as 6mA
- Control interface is accessible via pin header, easy to be integrated and embedded into user products.

Applications

The ESP32-CAM suit for IOT applications such as:

- Smart home devices image upload
- Wireless monitoring
- Intelligent agriculture
- QR wireless identification
- Facial recognition

5.5 Push Button Switch

Push Buttons are normally-open tactile switches. Push buttons allow us to power the circuit or make any particular connection only when we press the button. Simply, it makes the circuit connected when pressed and breaks when released. A push button is also used for triggering the SCR by the gate terminal.

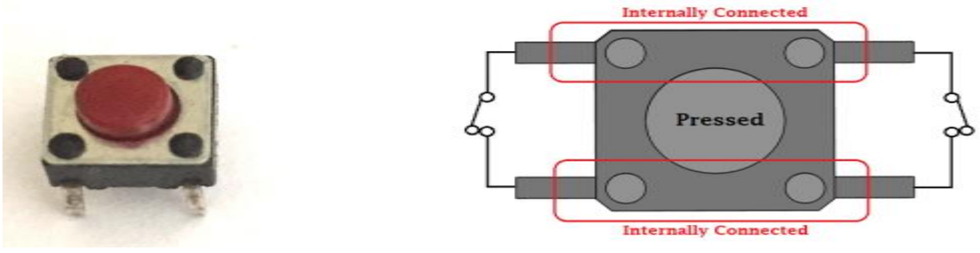


Fig5.5.1: Pin out Diagram of Push Button

Push Button Features

- Prevent flux rise by the insert-molded terminal.
- Snap-in mount terminal.
- Contact Bounce: MAX 5mS.
- Crisp clicking by tactile feedback.
- Dielectric Withstanding Voltage 250V AC for 1 minute.

Technical Specifications

- Mode of Operation: Tactile feedback
- Power Rating: MAX 50mA 24V DC
- Insulation Resistance: 100Mohm at 100v
- Operating Force: 2.55±0.69 N
- Contact Resistance: MAX 100mOhm
- Operating Temperature Range: -20 to +70 ℃
- Storage Temperature Range: -20 to +70 ℃

5.6 Python Software

Python is a popular programming language. It was created by Guido van Rossum, and released in 1991.

Application

- Web development (server-side)
- Software development
- Mathematics
- System scripting.
- Python can be used on a server to create web applications.
- Python can be used alongside software to create workflows.
- Python can connect to database systems. It can also read and modify files.
- Python can be used to handle big data and perform complex mathematics.

- Python can be used for rapid prototyping, or for production-ready software development.

Features

- Python works on different platforms (Windows, Mac, Linux, Raspberry Pi, etc).
- Python has a simple syntax similar to the English language.
- Python has syntax that allows developers to write programs with fewer lines than some other programming languages.
- Python runs on an interpreter system, meaning that code can be executed as soon as it is written. This means that prototyping can be very quick.
- Python can be treated in a procedural way, an object-oriented way or a functional way.

Modification

- The most recent major version of Python is Python 3, which we shall be using in this tutorial. However, Python 2, although not being updated with anything other than security updates, is still quite popular.
- In this tutorial Python will be written in a text editor. It is possible to write Python in an Integrated Development Environment, such as Thonny, Pycharm, Netbeans, or Eclipse which are particularly useful when managing large.
- Collections of Python files.

6. IMPLEMENTATION

6.1 Haar Algorithm

Haar feature-based algorithm is an effective method proposed for Face detection. But it is also an efficient method for object detection by the collection of more positive and negative samples. It is a machine learning-based approach. Haar features consider adjacent rectangle-shaped regions at a specific location in a detection window, and then add up the pixel intensities in each specific region and the difference between these sums is calculated. This difference is used to categorize the portions of the image. In the detection phase, a window of the target size is slide over the input image, and for each section of the image, the Haar-like feature is calculated. This difference is then compared to a certain threshold that separates nonobjects from objects.

- **Computation of Haar-like features**
  Haar-like features are digital image feature used in object recognition. They owe their name to their intuitive similarity with Haar wavelets and were used in the first real-time face detector.
- One of the contributions of Viola and Jones was to use summed-area tables which they called integral images. Integral images can be defined as two-dimensional lookup tables in the form of a matrix with the same size of the original image. Each element of the integral image contains the sum of all pixels located on the up-left region of the original image (in relation to the element's position).

sum = I(C) + I (A) - I (B) - I (D)

where points A, B, C, D belong to the integral image I, shown in the figure.

- Each Haar-like feature may need more than four lookups, depending on how it was defined. Viola and Jones's 2-rectangle features need six lookups, 3-rectangle features need eight lookups, and 4-rectangle features need nine lookups

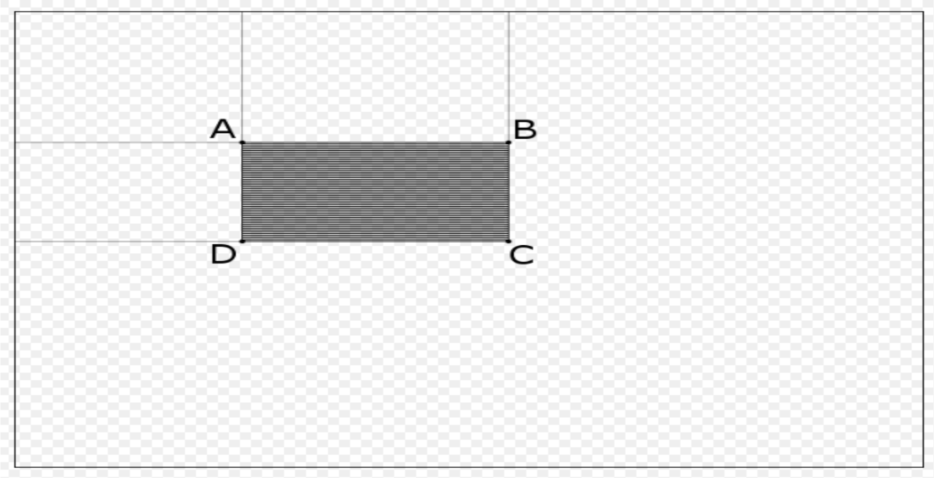


Fig 6.1.1: Area of Haar-like features

6.2 Cascade classifier

The cascade classifier consists of a list of stages, in which each stage consists of a list of weak learners. The system detects objects by moving a window over the image. Each stage of the classifier labels the specific region as either positive or negative. If object was detected then the output is positive and negative means that the specified object was not detected in the image. If the labeling yields a negative result, then the classification of the specific region is thereby complete and the location of the window is moved to the next location. If the labeling produces a positive result, then the region moves on to the next stage of classification. The classifier gives a final verdict of positive, when all the stages of the classifier, including the last one, give a result, saying that the object to be detected is found in the image.

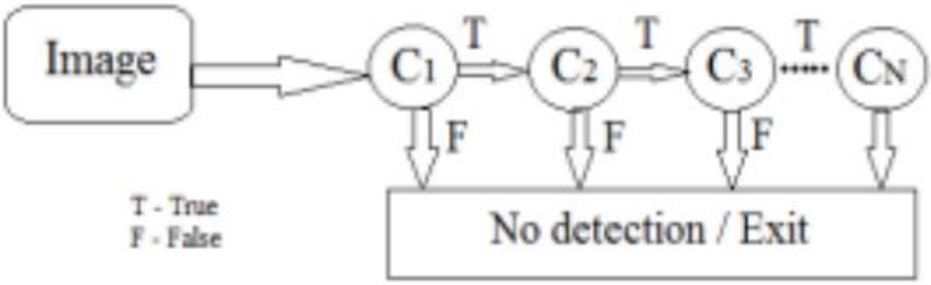


Fig 6.2.1: Cascade Classifier for Face Detection

There are four cases in output consideration, A true positive - the object in question is indeed present in the image processed and the classifier labels it as such which indicates a positive result. A false positive - the labeling process falsely determines that the object is located in the image, even though it is not. A false negative - the classifier is unable to determine the actual object from the image. A true negative – the classifier does not detect an object even it is present in the detection window.

6.3 Libraries :

6.3.1 eSpeak

eSpeak is a free and open-source, cross-platform, compact, software speech synthesizer. It uses a formant synthesis method, providing many languages in a relatively small file size. eSpeakNG (Next Generation) is a continuation of the original developer's project with more feedback from native speakers. Because of its small size and many languages, eSpeakNG is included in NVDA open source screen reader for Windows, as well as Android, Ubuntu and other Linux distributions. Its predecessor eSpeak was recommended by Microsoft in 2016 and was used by Google Translate for 27 languages in 2010, 17 of these were subsequently replaced by proprietary voices. It supports Windows and Linux. It utilizes a formant synthesis method that allows many languages to be provided in a small size. The speech is clear and can be used at high speeds.

6.3.2 Open CV

OpenCV is a huge open-source library for computer vision, machine learning, and image processing. OpenCV supports a wide variety of programming languages like Python, C++, Java, etc. It can process images and videos to identify objects, faces, or even the handwriting of a human. When it is integrated with various libraries, such as Numpy which is a highly optimized library for numerical operations, then the number of weapons increases in your Arsenal i.e whatever operations one can do in Numpy can be combined with OpenCV. This OpenCV tutorial will help you learn Image-processing from Basics to Advance, like operations on Images, Videos using a huge set of Opencv-programs and projects.

6.3.3 OS

Python OS module provides the facility to establish the interaction between the user and the operating system. It offers many useful OS functions that are used to perform OS-based tasks and get related information about the operating system. The OS comes under Python's standard utility modules. This module offers a portable way of using operating system-dependent functionality. The Python OS module lets us work with the files and directories. It is possible to automatically perform many operating system tasks. The OS module in Python provides functions for creating and removing a directory (folder), fetching its contents, changing and identifying the current directory, etc.

Python OS module provides the facility to establish the interaction between the user and the operating system. It offers many useful OS functions that are used to perform OS-based tasks and get related information about operating system. The OS comes under Python's standard utility modules.

7. CODE

7.1 Code for Recognition

```
import cv2, sys, numpy, os,scipy
size =
fn_haar = 'haarcascade4_frontalface_default.xml'
fn_dir = 'att_faces'
import os
import time
from collections import Counter
name ="abrar"
import pyttsx3
import pytesseract
engine = pyttsx3.init()

# Part 1: Create fisherRecognizer
print('Training...')
start = time.time() + 60
# Create a list of images and a list of corresponding names
(images, lables, names, id) = ([], [], {}, 0)
#sd = ""
# Get the folders containing the training data
for (subdirs, dirs, files) in os.walk(fn_dir):

# Loop through each folder named after the subject in the photos
for subdir in dirs:
names[id] = subdir
subjectpath = os.path.join(fn_dir, subdir)

# Loop through each photo in the folder
```

```
for filename in os.listdir(subjectpath):

# Skip non-image formates
f_name, f_extension = os.path.splitext(filename)
if(f_extension.lower() not in
['.png','.jpg','.jpeg','.gif','.pgm']):
print("Skipping "+filename+", wrong file type")
continue
path = subjectpath + '/' + filename
lable = id

# Add to training data
images.append(cv2.imread(path, 0))
lables.append(int(lable))
id += 1
(im_width, im_height) = (112, 92)

# Create a Numpy array from the two lists above
(images, lables) = [numpy.array(lis) for lis in [images, lables]]

# OpenCV trains a model from the images
# NOTE FOR OpenCV2: remove '.face'
#cv2.face.createLBPHFaceRecognizer()
#model = cv2.face.createLBPHFaceRecognizer()
#model = cv2.face.createLBPHFaceRecognizer()
model = cv2.face.LBPHFaceRecognizer_create()

#model = cv2.face.createFisherFaceRecognizer()
model.train(images, lables)
```

```
# Part 2: Use fisherRecognizer on camera stream
haar_cascade = cv2.CascadeClassifier(fn_haar)
webcam = cv2.VideoCapture(0)
while True:
#print ("##############################################")
#start = time.time()
# Loop until the camera is working
rval = False
while(not rval):
# Put the image from the webcam into 'frame'
(rval, frame) = webcam.read()
if(not rval):
print("Failed to open webcam. Trying again...")

# Flip the image (optional)
frame=cv2.flip(frame,1,0)

# Convert to grayscalel
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

# Resize to speed up detection (optinal, change size above)
mini = cv2.resize(gray, (int(gray.shape[1] / size), int(gray.shape[0] / size)))

# Detect faces and loop through each one
faces = haar_cascade.detectMultiScale(mini)
for i in range(len(faces)):
face_i = faces[i]

# Coordinates of face after scaling back by `size`
(x, y, w, h) = [v * size for v in face_i]
```

```
face = gray[y:y + h, x:x + w]
face_resize = cv2.resize(face, (im_width, im_height))

# Try to recognize the face
prediction = model.predict(face_resize)
#print (prediction)
cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 3)
cv2.putText(frame,
'%s - %.0f' % (names[prediction[0]],prediction[1]),
(x-10, y-10), cv2.FONT_HERSHEY_PLAIN,1,(0, 255, 0))
print (names[prediction[0]])

if (names[prediction[0]]) == "abrar":
#print "priyam"
name = "abrar"

engine.setProperty('rate', 150)    # Speed percent
engine.setProperty('volume', 0.9) # Volume 0-1
engine.say(name)
engine.runAndWait()

elif (names[prediction[0]]) == "vaibhav":
name = "vaibhav"
engine.setProperty('rate', 150)    # Speed percent
engine.setProperty('volume', 0.9) # Volume 0-1
engine.say(name)
engine.runAndWait()
print (name)

elif (names[prediction[0]]) == "rutuja":
```

```
name = "rutuja"
engine.setProperty('rate', 150)    # Speed percent
engine.setProperty('volume', 0.9) # Volume 0-1
engine.say(name)
engine.runAndWait()
print (name)

elif (names[prediction[0]]) == "bhargavi":
name = "bhargavi"
engine.setProperty('rate', 150)    # Speed percent
engine.setProperty('volume', 0.9) # Volume 0-1
engine.say(name)
engine.runAndWait()
print (name)

your_name = str(name)+ "\n"

cv2.imshow('OpenCV', frame)
key = cv2.waitKey(10) & 0xFF
if key == ord("q"):
pass
break
```

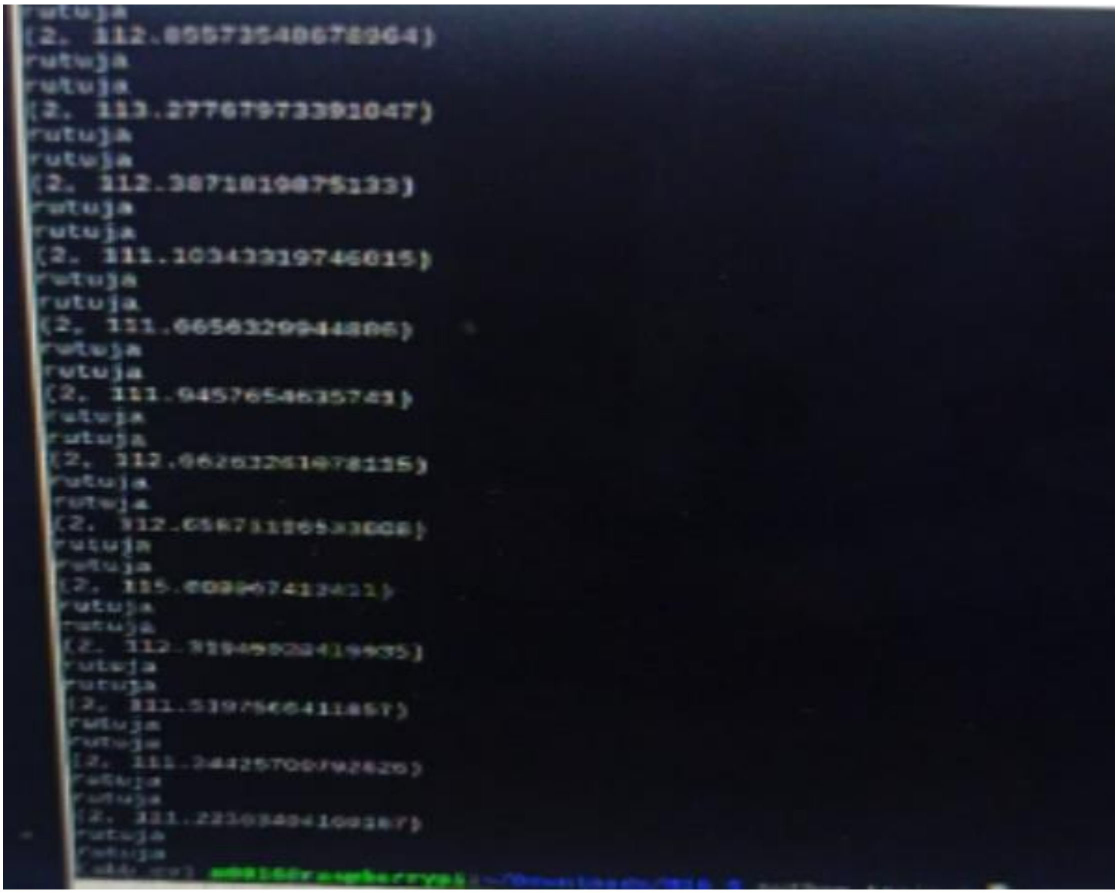
7.2 Code for Training

```
import cv2, sys, numpy, os
size = 4
fn_haar = 'haarcascade_frontalface_default.xml'
fn_dir = 'att_faces'
try:
fn_name = 'Abrar'
except:
print("You must provide a name")
sys.exit(0)
path = os.path.join(fn_dir, fn_name)
if not os.path.isdir(path):
os.mkdir(path)
(im_width, im_height) = (112, 92)
haar_cascade = cv2.CascadeClassifier(fn_haar)
webcam = cv2.VideoCapture(0)

# Generate name for image file
pin=sorted([int(n[:n.find('.')]) for n in os.listdir(path)
if n[0]!='.' ]+[0])[-1] + 1

# Beginning message
print("\n\033[94mThe program will save 50 samples. \
Move your head around to increase while it runs.\033[0m\n")

# The program loops until it has 50 images of the face.
count = 0
pause = 0
count_max = 50
```

```
while count < count_max:

# Loop until the camera is working
rval = False
while(not rval):
# Put the image from the webcam into 'frame'
(rval, frame) = webcam.read()
if(not rval):
print("Failed to open webcam. Trying again...")

# Get image size
height, width, channels = frame.shape

# Flip frame
frame = cv2.flip(frame, 1, 0)

# Convert to grayscale
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)

# Scale down for speed
mini = cv2.resize(gray, (int(gray.shape[1] / size), int(gray.shape[0] / size)))

# Detect faces
faces = haar_cascade.detectMultiScale(mini)

# We only consider largest face
faces = sorted(faces, key=lambda x: x[3])
if faces:
face_i = faces[0]
(x, y, w, h) = [v * size for v in face_i]
```

```
face = gray[y:y + h, x:x + w]
face_resize = cv2.resize(face, (im_width, im_height))

# Draw rectangle and write name
cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 3)
cv2.putText(frame, fn_name, (x - 10, y - 10), cv2.FONT_HERSHEY_PLAIN,
1,(0, 255, 0))

# Remove false positives
if(w * 6 < width or h * 6 < height):
print("Face too small")
else:
# To create diversity, only save every fith detected image
if(pause == 0):
print("Saving training sample "+str(count+1)+"/"+str(count_max))
# Save image file
cv2.imwrite('%s/%s.png' % (path, pin), face_resize)
pin += 1
count += 1
pause = 1
if(pause > 0):
pause = (pause + 1) % 5
cv2.imshow('OpenCV', frame)
key = cv2.waitKey(10)
if key == 27:
break
```

8. RESULT & DISCUSSION



```
rutuja
(2, 112.85573548678964)
rutuja
rutuja
(2, 113.27767973392047)
rutuja
rutuja
(2, 112.3871819875133)
rutuja
rutuja
(2, 111.10343319746815)
rutuja
rutuja
(2, 111.6656329944886)
rutuja
rutuja
(2, 111.9457654635741)
rutuja
rutuja
(2, 112.06263261078115)
rutuja
rutuja
(2, 112.65671196533608)
rutuja
rutuja
(2, 115.608967413411)
rutuja
rutuja
(2, 112.71949928419935)
rutuja
rutuja
(2, 111.5397566411857)
rutuja
rutuja
(2, 111.24425700792626)
rutuja
rutuja
(2, 111.22103494109187)
rutuja
rutuja
[illegible]
```

Fig8.1: Live Demo of Camera Module

The screenshot of the live demo of the camera module is shown. When a person comes in In front of the smart glasses, the camera module i.e., Pi camera captures the image of the person and will run an analysis with the Face Dataset folder, and if any matches are shown then the name of the person is given as voice command through earphones connected using the Facial recognition.

9. CONCLUSION & FUTURE SCOPE

This project presents a proposed idea of smart guiding devices for visually impaired users, which help them move safely and efficiently in complicated indoor and outdoor environments. The depth image and the multi-sensor fusion based algorithms solve the problems of small and transparent obstacle avoidance. Three main auditory cues for the totally blind users were developed and tested in different scenarios, and results show that the beep sound based guiding instructions are the most efficient and well adapted. For weak sighted users, visual enhancement based on AR technique was adopted to integrate the traversable direction into the binocular images and it helps he users to walk more quickly and safely. The computation is fast enough for the detection and display of obstacles. Experimental results of the proposed prototype shows that the proposed smart guiding glasses can enhance the travelling experience of the visually impaired people. The use of simple and low cost sensors, make it widely usable in cons.

Design and simulation of Smart Glasses for visually challenged people has been done by using Raspberry-pi, camera. Smart Glasses for Visually Disabled People is currently an existing Technology outside of India. As computer vision algorithms, and hardware have been used together, the idea of developing wearable or portable assistive technologies for visually impaired people evolved. The device has been developed by us at a low cost of manufacturing, comes with audio output, and is convenient to use for day-to-day activities. The primitive version of these systems used basic image processing and computer vision techniques, while the recent versions are smart enough to draw a safe path for user navigation

Future Scope

The system capability, however, can be easily extended to multiple tasks by adding more models to the core program but is restricted by the size of the raspberry pi SD card. Each model represents a specific task or mode. The user can have the desired task run independently from the other tasks. The system design, working mechanism, and principles were discussed along with some experimental results. This new concept is expected to improve visually impaired students ‘lives despite their economic situations. Immediate future work includes assessing user-friendliness and optimizing the power management of the computing unit.

References

1. Abate, A. F., Nappi, M., Riccio, D., & Sabatino, G. (2021). 2D and 3D face recognition: A survey. Pattern recognition letters, 28(14)

2. Ding, C., Choi, J., Tao, D., & Davis, L. S. (2020). Multi-directional multi-level dual-cross patterns for robust face recognition. IEEE transactions on pattern analysis and machine intelligence, 38(3), 518-531.

3. Ganguly, S., Bhattacharjee, D., & Nasipuri, M. (2021). Illumination, pose and occlusion invariant face recognition from range images using ERFI model. International Journal of System Dynamics Applications (IJSDA), 4(2), 1-20.

4. Zhu, X., Lei, Z., Yan, J., Yi, D., & Li, S. Z. (2019). High-fidelity pose and expression normalization for face recognition in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (pp. 787-796).

5. Schroff, F., Treibitz, T., Kriegman, D., & Belongie, S. (2020, November). Pose, illumination and expression invariant pairwise face-similarity measure via doppelgänger list comparison. In Computer Vision (ICCV), 2020 IEEE International Conference on (pp. 2494- 2501). IEEE.

6. Wright, J., Yang, A. Y., Ganesh, A., Sastry, S. S., & Ma, Y. (2018). Robust face recognition via sparse representation. IEEE transactions on pattern analysis and machine intelligence, 31(2), 210-227.

7. Min, R., Hadid, A., & Dugelay, J. L. (2020, March). Improving the recognition of faces occluded by facial accessories. In Automatic Face & Gesture Recognition and Workshops (FG 2020), 2020 IEEE International Conference on (pp. 442-447). IEEE.

8. Li, X. X., Dai, D. Q., Zhang, X. F., & Ren, C. X. (2021). Structured sparse error coding for face recognition with occlusion. IEEE transactions on image processing, 22(5), 1889-1900.

9. Li, X. X., Dai, D. Q., Zhang, X. F., & Ren, C. X. (2021). Structured sparse error coding for face recognition with occlusion. IEEE transactions on image processing, 22(5), 1889-1900.

10. Zou, W. W., & Yuen, P. C. (2021). Very low resolution face recognition problem. IEEE Transactions on Image Processing, 21(1), 327- 340.