**Safe Local Navigation for Visually Impaired Users**

**A PROJECT REPORT**

***Submitted by***

**K.SAKTHIVEL (311616106131)**

**K.SATISH (311616106141)**

**R.SARAVANAN (311616106138)**

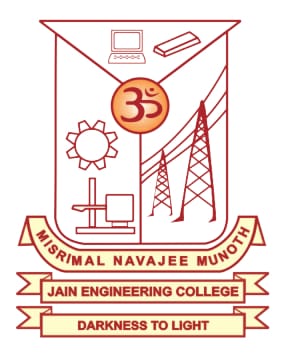
#### in partial fulfillment for the award of the degree of

**BACHELOR OF ENGINEERING**

***in***

### ELECTRONICS AND COMMUNICATION ENGINEERING

**MISRIMAL NAVAJEE MUNOTH JAIN ENGINEERING COLLEGE**

****

**ANNA UNIVERSITY:: CHENNAI 600 025**

**APRIL 2020**

**Safe Local Navigation for Visually Impaired Users**

**A PROJECT REPORT**

***Submitted by***

**K.SAKTHIVEL (311616106131)**

**K.SATISH (311616106141)**

**R.SARAVANAN (311616106138)**

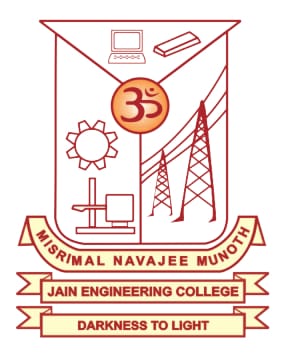
#### in partial fulfillment for the award of the degree of

**BACHELOR OF ENGINEERING**

***in***

### ELECTRONICS AND COMMUNICATION ENGINEERING

**MISRIMAL NAVAJEE MUNOTH JAIN ENGINEERING COLLEGE**

****

**ANNA UNIVERSITY:: CHENNAI 600 025**

**APRIL 2020**

**BONAFIDE CERTIFICATE**

Certified that this project report **“SAFE LOCAL NAVIGATION FOR VISUALLY IMPAIRED USERS”**is the bonafide work of “**K SAKTHIVEL (311616106131)** ,**K SATISH (311616106141)** and **R SARAVANAN (311616106138)** who carried out the project work under my supervision.

###### **SIGNATURE SIGNATURE**

###### **Dr.N.JAISANKAR, M.E., PhD, Mr.ARUNPRASAD M.E.,**

###### **PROFESSOR AND HEAD SUPERVISOR**

###### **Assisstant Professor**

###### Department of Electronics and Department of Electronics and

###### Communication Engineering Communication Engineering

###### M.N.M Jain Engineering M.N.M Jain Engineering

###### College,Thoraipakkam College,Thoraipakkam

###### Chennai-600 097. Chennai-600 097.

Submitted for the project work and viva examination held on ……………………

**INTERNAL EXAMINER EXTERNAL** **EXAMINER**

**ACKNOWLEDGEMENT**

We express our gratitude and sincere thanks to our respected, Secretary **Dr. Harish L Metha** (Admin), Secretary **Shri.L.Jaswant Munoth** (Academic) and our beloved Principal **Dr.C.Chandrasekar Christopher** for providing us with all kind of infrastructure for the successful completion of this project.

We are indebted to our Professor and Head, Department of Electronics and Communication Engineering, **Dr.N.Jaisankar** and our project coordinator **Dr.K.Jayabharathi**, Associate Professor, for their enthusiastic motivation and continuous encouragement which inspired us a lot in completing this project.

We would like to express our gratitude to our guide **Mr.P.ARUNPRASAD**, Assisstant Professor of Electronics and Communication Department for his valuable suggestions and constant encouragement that led to the successful completion of this project.

Last but not least we are grateful to our parents and friends for this constant Encouragement and support to complete project successfully.

**ABSTRACT**

Obstacle detection and warning can improve the camera as well as the safety of visually impaired and deaf people especially in unfamiliar environments. For this, firstly, obstacles are detected and localized and then the information of the obstacles will be sent to the visually impaired and deaf people by using different modalities such as voice, vibration. In this paper, we present an assistive system for visually impaired people based camera. This system consists of two main components: environment information acquisition and analysis and information representation. The first component aims at capturing the environment by using a camera and analyzing it in order to detect the predefined obstacles for visually impaired and deaf people, while the second component when there is an obstacle to the right of the user; he or she feels strong frequent pulses on the right side through the haptic strap. For the user, this leads to the perception of an obstacle to the right. The user can then avoid the obstacle by stepping left. After stepping left, the pulses on the right side stop and the user perceives that there is no longer an obstacle on his or her right side

**TABLE OF CONTENTS**

**CHAPTER NO. TITLE PAGENO.**

**ACKNOWLEDGEMENT iv**

**ABSTRACT v**

**LIST OF FIGURES vii**

1. **INTRODUCTION 1**

1.1 OBJECTIVE 1

1.2 EXISTING SYSTEM 1

* 1. PROPOSED SYSTEM 2

1. **LITERATURE SURVEY 3**
2. **BLOCK DIAGRAM AND REQUIREMENTS 5**

3.1 BLOCK DIAGRAM 5

3.2 HARDWARE REQUIREMENTS 6

* 1. SOFTWARE REQUIREMENTS 6

**4. HARDWARE REQUIREMENTS 7**

4.1 RASHBERRY PI 7

4.2 FEATURES 7

4.3 PROCESSOR 8

4.4 RASHBERRY PI POWER SUPPLY 8

4.4.1 RASHBERRY PI MODEL B+ GPIO PORT 10

4.5 RASHBERRY PI CONFIGURATION 11

4.6 ARCHITECTURE 14

4.6.1 HDMI VIDEO 15

4.6.2 DSI VIDEO 16

4.7 GPIO PORT 16

4.7.1 UART SERIAL BUS 18

4.7.2 FC BUS 18

4.7.3 SPI BUS 19

4.8 CAMERA 20

**5. SOFTWARE REQUIREMENTS 24**

5.1 RASHBERRY PI SOFTWARE 24

5.2 PYTHON PROGRAMMING LANGUAGE 24

5.2.1 INSTALLING OS TO RASHBERRY PI B+ 25

5.3 PYTHON IDE 29

5.4 PYTHON AND NETWORKING 33

5.5 IMAGE PROCESSING USING PYTHON 39

5.5.1 HARDWARE 39

5.5.2 SOFTWARE 40

5.6 INTRODUCTION TO OPEN CV 40

5.6.1 INSTALL PYTHON OPEN CV 40

5.7 DATA VISUALIZATION IN PYTHON 41

**6. FUTURESCOPE AND CONCLUSION 42**

6.1 APPLICATION 42

6.2 CONCLUSION 42

**LIST OF FIGURES**

**FIGURE NO. TITLE PAGE NO.**

3.1 BLOCK DIAGRAM 5

4.1 RASHBERRY PI POWER SUPPLY CIRCUIT DIAGRAM 9

4.2 RASPBERRY PI MODEL B+ GPIO PORT 10

4.3 RASHBERRY PI ARCHITECTURE 14

4.4 GPIO PINOUT DIAGRAM 17

4.5 CAMERA 21

**CHAPTER 1**

**INTRODUCTION**

Blindness is a state of lacking the visual perception due to neurological or physiological factors. The partial blindness represents the lack of integration in the growth of the optic visual or nerve centre of the eye, and total blindness is the full absence of the visual light perception. In this work, cheap, a simple friendly user, smart blind guidance system is designed and implemented to improve the mobility of both blind and visually impaired people in a specific area. The proposed work includes a wearable equipment consists of light weight blind stick and sensor based obstacle detection circuit is developed to help the blind person to navigate alone safely and to avoid any obstacles that may be encountered, whether fixed or mobile, to prevent any possible accident. The main objective of this project is to develop an application for blind people to detect the objects in various directions, detecting pits and manholes on the ground to make free to walk Detecting objects using image processing can be used in multiple industrial as well as social applications. This project is proposing to use object detection for blind people and give them audio/ vocal information about it. We are detecting an object using the mobile camera and giving voice instructions about the direction of an object.

* 1. **OBJECTIVE**

This paper proposes that mechanical feedback coupled to sensor technology as well as perception and planning algorithms is a potential solution to enable people with visual impairments to navigate their environment discreetly and effortlessly. This solution is not an attempt to recreate sight, but to process sensor information into useful navigational information that the user can rely on to create a mental map of their surroundings. Embedding small and lightweight sensors along with haptic devices in clothes allows for seamless integration of wearable navigation technology into the lives of visually impaired people.

**1.2 Existing System**

The common way for navigating of visionless person is using a walking stick cane or walking cane. The walking cane is a simple and mechanical device dedicated to detect static obstacles on the ground, uneven surfaces, and holes via simple tactile-force feedback. This device is light, portable but range limited and it is not usable for the protection from obstacles near to head area. Another option that provides the best travel aid for the blind is the guide dogs. The disabled owner and his dog, the training and the relationship to the animal are the keys to success for this method. The dog is able to detect and analyse complex situations: cross walks, stairs, potential danger, know paths and more.

**1.3 Proposed system**

Our proposed visual substitution system is based on the identification of objects around the blind and deaf person. We propose a system that recognize and locate 2D in the video. This system should find the invariant characteristic of objects, provide the recognition and reduce the complexity of detection. We propose a method based on object extraction and matching in video. A comparison between query frame and database objects is made to detect object in each frame. For each object detected an audio file containing the information about it is activate.

**CHAPTER 2**

**2.LITERATURE SURVEY**

1. **Navigation Assistive System for the Blind using a Portable Depth Sensor**

**Author:** Kumar Yelamarthi

The lightweight and low-cost 3-dimensional depth sensors have gained much attention in the computer vision and robotics community. While its performance has been proven successful in the robotics community, these sensors have not been utilized successfully for many assistive devices. Leveraging on this gap, this paper presents the design, implementation, and preliminary evaluation of a haptic feedback system for the blind using 3-D depth sensors. The proposed portable system interprets the visual scene using the depth sensor, converts it into distance map, processes, and evaluates this information using a tablet computer

1. **Digital Assistant For The Blind**

**Author**: Prince Bose, Apurva Malpthak, Utkarsh Bansal, Ashish Harsola.

The blind and the visually impaired have little to no internet presence because of the absence of cheap solutions to get them online which can be both, hardware and software. Existing technology used for enabling the blind or visually impaired to use the internet or any digital form of information is dependent on Braille displays and keyboards which are expensive and scarce. Another shortcoming of existing technology is that out of all the visually impaired population, less than 2% know how to interpret Braille. Hence a voice controlled system for the blind and the visually impaired was designed, which transceivers information in the form of audio.

**3.Android Assistant EyeMate for Blind and Blind Tracker**

**Author:** Md. Siddiqur Rahman Tanveer, M.M.A. Hashem and Md. Kowsar.

At present many blind assistive systems have been implemented but there is no such kind of good system to navigate a blind person and also to track the movement of a blind person and rescue him/her if he/she is lost. In this paper, we have presented a blind assistive and tracking embedded system. In this system the blind person is navigated through a spectacle interfaced with an android application. The blind person is guided through Bengali/English voice commands generated by the application according to the obstacle position. Using voice command a blind person can establish voice call to a predefined number without touching the phone just by pressing the headset button.

1. **A vision and speech enabled, customizable, virtual assistant for smart environments**

**Author**: Giancarlo Iannizzotto,Lucia Lo BelloAndrea,Nucita,Giorgio Mario Grasso.

Recent developments in smart assistants and smart home automation are lately attracting the interest and curiosity of consumers and researchers. Speech enabled virtual assistants (often named smart speakers) offer a wide variety of network oriented services and, in some cases, can connect to smart environments, thus enhancing them with new and effective user interfaces. However, such devices also reveal new needs and some weaknesses. In particular, they represent faceless and blind assistants, unable to show a face, and therefore an emotion, and unable to ‘see’ the user.

1. **A new Approach for Pedestrian Detection in Vehicles by Ultrasonic Signal Analysis**

**Author** : Andreas H. Pech.

The ultrasonic sensor evaluates signals back scattered from the obstacle by task specific signal analysis methods. It consists of an ultrasonic transducer, an embedded system for transducer control and signal acquisition as well as a computer for signal analysis and feature extraction. Unlike Radar systems this pedestrian detection approach is also able to detect non-moving persons and even children.

1. **Multi sensor – based Object Detection in Indoor Environment forVisually Impaired People**

**Author** : Charmi T.Patel.

The system consists of a multi sensor-based system for object detection on a captured image using statistical parameters, which is further validated using support vector machine algorithm. To increase the accuracy, multi-sensor concept is employed by interfacing ultrasonic sensor.

1. **Hot Glass Human Face, Object and Textual Recognition For Visually Challenged**

**Author** : Diwakar Srikanth A.

The system was proposed in order to make the visually impaired people virtually present. The kit contained a camera, a microphone and a system where the processing is carried out. The system has a database which consists text information for each image which is converted into an audio note. This process involves Feature extraction, Gray Scale Conversion, PCA algorithm, SIFT algorithm and OCR technique.

**8.Face Detection and Tracking using Image Processing on Raspberry Pi**

**Author**: Vivek Kishore Bhanshe.

The kituses Raspberry Pi as the processor control and a web camera to process and detect the image using Open CV, which then tracks and recognizes the human face. The image is captured as frames per second and using the haar cascading methods the image is processed and then detected.

**CHAPTER 3**

**SYSTEM DESIGN**

**3.BLOCK DIAGRAM AND REQUIREMENTS**

Input video

Frames reading

Features Extraction

Reference Image

Feature Matching

Final classification

Features Extraction

Object detection

Camera

**Raspberry pi**

Audio output

Haptic strip

**3.1 BLOCK DIAGRAM**

**Working**

FIG 3.1

Fig 3.1 Block diagram

The blind person taking video of the path where he was walking the application will give voice message to that blind and deaf person and it will help to that person for identifying he’s path. The object gets detected by the key matching technique which is used in the algorithm. And match that object with the database images to confirm the obstacle that comes into the way. When object is matched with database objects the application gives the voice instruction by using the Speech synthesizer. So, Blind user gets the direction from the application. They output form the haptic strip. Shorter distances to obstacles are relayed by increased pulse rates with higher vibration strength. For example, when there is an obstacle to the right of the user, he or she feels strong frequent pulses on the right side through the haptic strap. For the user, this leads to the perception of an obstacle to the right. The user can then avoid the obstacle by stepping left. After stepping left, the pulses on the right side stop and the user perceives that there is no longer an obstacle on his or her right side.

3.2 CIRCUIT DIAGRAM

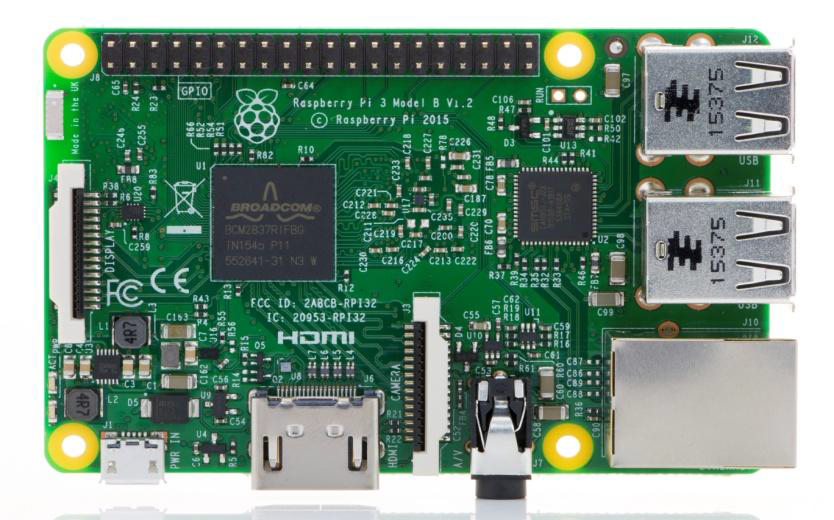
CONNECTION:

3.3 FLOW CHART REPRESENATION

**3.4 HARDWARE REQUIREMENT**

**3.4.1 Raspberry pi 3+(Model B)**

The Raspberry Pi 3 Model B is the third generation Raspberry Pi. This powerful credit-card sized single board computer can be used for many applications and supersedes the original Raspberry Pi Model B+ and Raspberry Pi 2 Model B. Whilst maintaining the popular board format the Raspberry Pi 3 Model B brings you a more powerful processer, 10x faster than the first generation Raspberry Pi. Additionally it adds wireless LAN & Bluetooth connectivity making it the ideal solution for powerful connected designs.

****

**FIG 3.4** Raspberry pi model b 3+

**Specification**

**Processor:** Broadcom BCM2837 SOC with 1.2GHz 64-bit quad-core –A53 with 512 kb shared L2 cache (64-bit instruction set ARMv8).

**Memory:**1GB LPDDR2 SDRAM.

**Access:**Extended 40- PIN GPIO Header.

**Input power:**5V/2.5A DC via micro USB connector, 5V DC via GPIO header, Power over Ethernet (PoE)–enabled (requires separate PoE HAT)

**Environment:**Operating temperature, 0–50°C.

**3.4.1.1 PIN DIAGRAM**

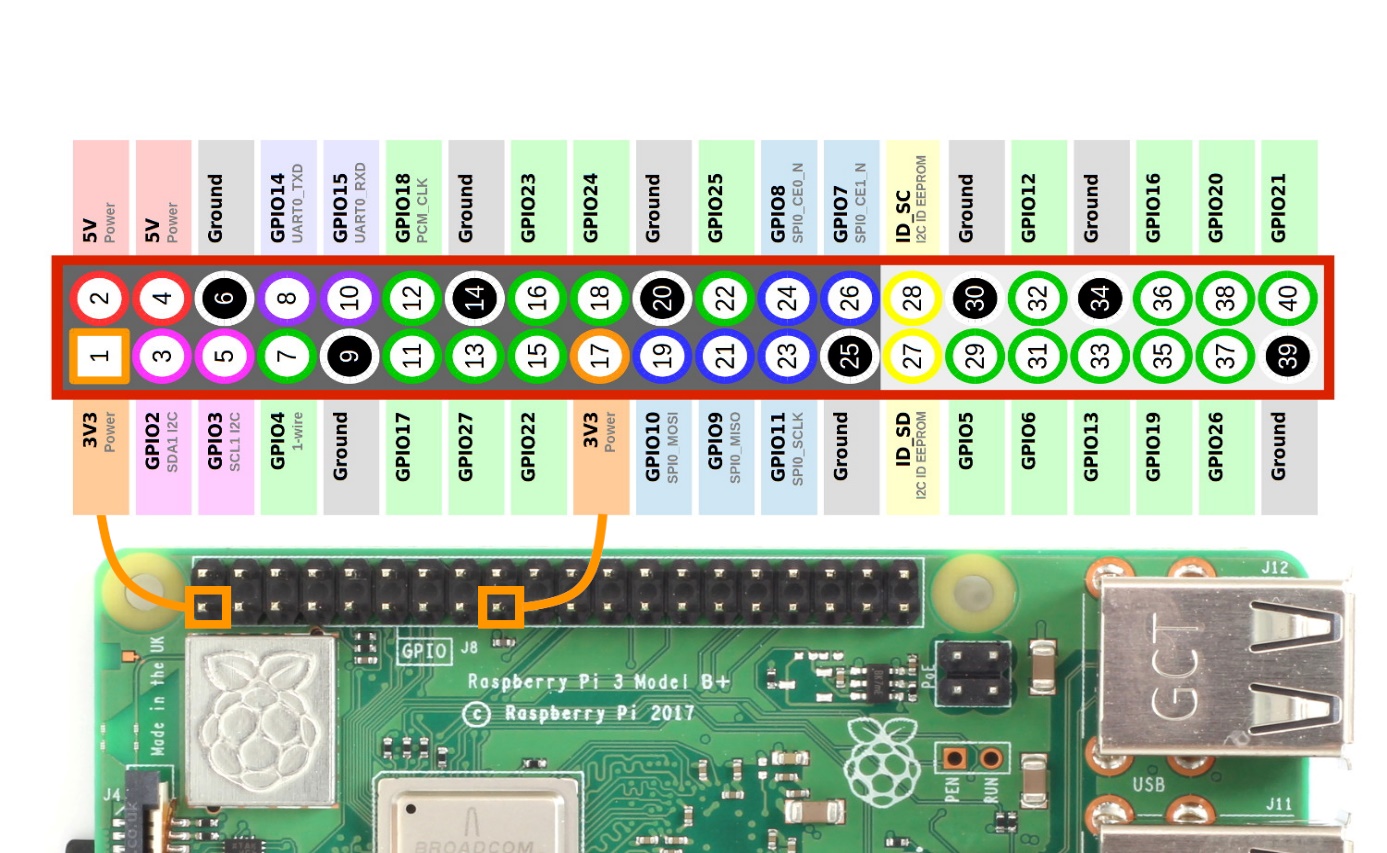


FIG 3.4.1 Pin Diagram for Raspberry pi

The Raspberry Pi’s GPIO port is located on the top-left of the printed circuit board, labelled P1. It’s a 54-pin port, fitted with two rows of 13 male 2.54 mm headers at the factory. The spacing of these headers is particularly important: 2.54 mm pin spacing (0.1 inches in imperial measurements) is a very common sight in electronics, and is the standard spacing for prototyping platforms that include stripboards and breadboards. Each pin of the GPIO port has its own purpose, with several pins working together to form particular circuits. The layout of the GPIO port. The Raspberry Pi’s GPIO port and its pin definitions Pin numbers for the GPIO port are split into two rows, with the bottom row taking the odd numbers and the top row the evennumbers. It’s important to keep this in mind when working with the Pi’s GPIO port: most other devices use a different system for numbering pins, and because there are no markings on the Pi itself, it’s easy to get confused as to which pin is which.

**GPIO PIN FUNCTIONS**

1. PWM (Pulse Width Modulation):

* Software on all pins.
* Hardware on GPIO12, GPIO13, GPIO18, GPIO19.

2. SPI (Serial Peripheral Interface):

* SPI0: MOSI (GPIO10), MISO (GPIO9). SCLK: (GPIO11). CE0: (GPIO8). CE1: (GPIO7)
* SPI1: MOSI (GPIO20), MISO (GPIO19). SCLK (GPIO21). CE0:

(GPIO18). CE1: (GPIO17). CE2 (GPIO16).

3. I2C (Inter Integrated Circuit):

 Data (GPIO2), Clock (GPIO3), EEPROM Data (GPIO0), EEPROM Clock (GPIO1).

4. Serial:

* TX (GPIO14), RX (GPIO15).

**3.4.1.2 Advantages:**

* Low Cost.
* Huge Processing power in a compact board.
* Many Interfaces. ( HDMI , Multiple USB, Ethernet, On board Wi-Fi and Bluetooth, many GPIO’s, USB powered, etc.,)

**3.4.2 Ultrasonic sensor**

Ultrasonic sensor measures distance by using ultrasonic waves. The sensor head emits an ultrasonic wave and receives the wave reflected back from the target. Ultrasonic sensor measures the distance to the target by measuring the time between emission and reception.

**Specifications:**

**Operating Voltage**:+5V

**Theoretical Measuring Distance**:2cm to 450cm

**Practical Measuring Distance**:2cm to 80cm

**Accuracy**:3mm

**Measuring angle covered**:<15°

**Operating Current**:<15mA

**Operating Frequency**:40Hz

**3.4.2.1 Pin diagram:**

**Vcc**- The pin powers the sensor, typically with 5V.

**Trigger**- Trigger pin is an Input pin. This pin has to be kept high for 10us to initialize measurement by sending US wave.

**Echo**- Echo pin is an Output pin. This pin goes high for a period of time which will be equal to the time taken for the US wave to return back to the sensor.

**Ground**- This pin is connected to the Ground of the system.



Fig 3.4.2 Pin Diagram of the Ultrasonic Sensor

**3.4.2.2 Advantages**

* Used to avoid and detect obstacles with robots like biped robot, obstacle avoider robot, path finding robot etc.,
* Used to measure the distance within a wide range of 2cm to 400cm.
* Can be used to map the objects surrounding the sensor by rotating it.

**3.4.3 Web camera**

A Webcam is a video camera that feeds or streams an image or video in real time to or through a computer to a computer network, such as the Internet. Webcams are typically small cameras that sit on a desk, attach to a user's monitor, or are built into the hardware. Webcams can be used during a video chat session involving two or more people, with conversations that include live audio and video.

**3.4.3.1 Specifications**

**Image sensor:**CMOS

**Max. Image resolution:**320 X 240, 640 X 480

**Interface:**USB 2.0

**Frame rate:**30 fps.

**OS supported:**Windows XP SP2/VISTA/7, LINUX Kernel 2.6.27.7 version.

**Cable length:** 1.10 metre.

**3.4.3.2 Advantages**

* Able to interact with people across long distances.
* Both sound and video is used, making the communication more sophisticated.
* Users can use the webcam to save videos they film to watch later or send to others.



Fig 3.4.3 WebCamera

**3.4.4 Headphones**

Headphones traditionally refer to a pair of small loudspeaker drivers worn on or around the head over a user's ears. They are electro acoustics transducers, which convert an electrical signal to a corresponding sound. Headphones let a single user listen to an audio source privately, in contrast to a loudspeaker, which emits sound into the open air for anyone nearby to hear.

* + - 1. **Specifications**

**Headphone type*:*** Wired in the Ear.

**Connectivity:**Wired

* + 1. **DC Motor**

A DC Motor is any of a class of rotary Electric motor that converts direct current electrical energy into mechanical energy. The most common types rely on the forces produced by magnetic fields. Nearly all types of DC motors have some internal mechanism, either electromechanical or electronic, to periodically change the direction of current in part of the motor.



Fig 3.4.5 Dc Motor

**3.4.5. Specification**

* Operating **Voltage**: 4.5V to 9V.
* Motor Size: 27.5mm x 20mm x 15mm.
* Standard 130 Type DC motor.
  + 1. **Haptic Strap**

A haptic strap also known as tactile strap is a wearable device that provides haptic feedback to the body.This Strap is a easy to use and has robustness.This strap holds the device for easy functioning.



Fig 3.4.6 Haptic strap

**3.4.6.1 Specification**

* **Size :**  100cm x 15 cm
* **Load :** 3-5 kg

**3.5 SOFTWARE REQUIREMENTS**

**3.5.1 RASHBERRY PI IDE**

Raspberry Pi is an embedded computer used for many types of applications. In Raspberry Pi, we use a physical board for external interface and an IDE for writing the software part of the application on a computer. Raspberry Pi uses python programming language for writing the software, where IDLE is used.

**3.5.1.2 IDLE Platform**

IDLE has been with Raspbian for generations as the default editor. As such, many introductory Python tutorials out there still show screenshots with IDLE. That being said, IDLE is little more than a text editor and a terminal window. In fact, editing code happens in a separate window than the terminal.Then the python programming language is used to run the necessary program in this Ide.

**3.5.1.3** **Python’s standard library**

* Pandas
* Numpy
* Sklearn
* seaborn
* matplotlib
* Importing Datasets
* Opencv

**3.5.1.4 INSTALL PYTHON OPEN CV**

* To implement this project, the following packages of Python 3.7 have to be downloaded and installed: Python 2.7.x, NumPy and Matplotlib. According to the default location, Python will be installed to C drive. Open Python IDLE, import all the packages

and start working.

* First, a sample image in which processing is to be applied is to be read. It’s done using a pre-defined Python function: CV2.imread(). The sample image should be available in current folder or the full location of the image is to be mentioned as an argument. For reading an image, we can use functions like imread\_color, imread\_grayscale,imread\_unchanged etc.

**CHAPTER 5**

**FUTURESCOPE AND CONCLUSION**

**6.1 APPLICATION**

Computer vision which go beyond image processing, helps to obtain relevant information from images and make decisions based on that information. In other words, computer vision is making the computer see as humans do. Basic steps for a typical computer vision application as follows.

1. Image acquisition
2. Image manipulation
3. Obtaining relevant information
4. Decision making

Computer vision has still not attained a level wherein it can be directly put into use to solve life problems, as it is still in its developmental phase. With passing years and rigorous pace at which research is being done, Computer Vision or to be precise, Object detection will be completely omnipresent. Computer Vision is a sub-part on Machine Learning. Some common and widely used application of object detection are:

* Accounting Number of Objects
* Automobile Spotting
* Biometric Detection
* Medical Diagnosis
* Supervision
* Machine Man Communication

**6.2 CONCLUSION**

In this paper we have used the functions of opencv It is free for both commercial and non-commercial use. Therefore you can use the OpenCV library even for your commercial applications. It is a library mainly aimed at real time processing. Now it has several hundreds of inbuilt functions which implement image processing and computer vision algorithms which make developing advanced computer vision applications easy and efficient.

**APPENDICES**

**APPENDIX**

**SOFTWARE CODING**

**$sudo apt update**

**$ sudo apt install python 3- opencv**

**Python3 -c “import cv2 ; print (cv2\_\_version\_\_)**

**Ultrasonic.py :**

import RPi.GPIO as GPIO

import time

GPIO.setmode(GPIO.BCM)

GPIO.setup(21,GPIO.OUT)

GPIO.setup(20,GPIO.IN)

while(1):

GPIO.output(21,True)

time.sleep(.00001)

GPIO.output(21,False)

w=time.time()

while(GPIO.input(20)==0):

start=time.time()

if(time.time()>w+5):

break

while(GPIO.input(20)==1):

stop=time.time()

diff=start-stop

dis=34300\*diff

print(dis)

**Final.py**

# USAGE

# python final.py --prototxt MobileNetSSD\_deploy.prototxt.txt --model MobileNetSSD\_deploy.caffemodel

# import the necessary packages

from imutils.video import VideoStream

from imutils.video import FPS

import numpy as np

import argparse

import imutils

import time

import cv2

import subprocess

import serial

import RPi.GPIO as GPIO

import time

GPIO.setmode(GPIO.BCM)

GPIO.setup(21,GPIO.OUT)

GPIO.setup(20,GPIO.IN)

GPIO.setup(16,GPIO.OUT)

def ultra():

GPIO.output(21,True)

time.sleep(.00001)

GPIO.output(21,False)

w=time.time()

while(GPIO.input(20)==0):

start=time.time()

if(time.time()>w+5):

break

while(GPIO.input(20)==1):

stop=time.time()

diff=stop-start

dis=34300\*diff

print(dis)

if dis<20:

GPIO.output(16,True)

else:

GPIO.output(16,False)

def tts():

text = label

filename = 'python\_text.txt'

file=open(filename,'w')

file.write(text)

file.close()

subprocess.call('festival --tts '+filename, shell=True)

subprocess.call('rm -f '+filename, shell=True)

# construct the argument parse and parse the arguments

ap = argparse.ArgumentParser()

ap.add\_argument("-p", "--prototxt", required=True,

help="path to Caffe 'deploy' prototxt file")

ap.add\_argument("-m", "--model", required=True,

help="path to Caffe pre-trained model")

ap.add\_argument("-c", "--confidence", type=float, default=0.2,

help="minimum probability to filter weak detections")

args = vars(ap.parse\_args())

# initialize the list of class labels MobileNet SSD was trained to

# detect, then generate a set of bounding box colors for each class

CLASSES = ["background", "aeroplane", "bicycle", "bird", "boat",

"bottle", "bus", "car", "cat", "chair", "cow", "diningtable",

"dog", "horse", "motorbike", "person", "pottedplant", "sheep",

"sofa", "train", "tvmonitor"]

COLORS = np.random.uniform(0, 255, size=(len(CLASSES), 3))

# load our serialized model from disk

print("[INFO] loading model...")

net = cv2.dnn.readNetFromCaffe(args["prototxt"], args["model"])

# initialize the video stream, allow the cammera sensor to warmup,

# and initialize the FPS counter

print("[INFO] starting video stream...")

vs = VideoStream(src=0).start()

# vs = VideoStream(usePiCamera=True).start()

time.sleep(2.0)

fps = FPS().start()

# loop over the frames from the video stream

while True:

ultra()

# grab the frame from the threaded video stream and resize it

# to have a maximum width of 400 pixels

frame = vs.read()

frame = imutils.resize(frame, width=400)

# grab the frame dimensions and convert it to a blob

(h, w) = frame.shape[:2]

blob = cv2.dnn.blobFromImage(cv2.resize(frame, (300, 300)),

0.007843, (300, 300), 127.5)

# pass the blob through the network and obtain the detections and

# predictions

net.setInput(blob)

detections = net.forward()

# loop over the detections

for i in np.arange(0, detections.shape[2]):

# extract the confidence (i.e., probability) associated with

# the prediction

confidence = detections[0, 0, i, 2]

# filter out weak detections by ensuring the `confidence` is

# greater than the minimum confidence

if confidence > args["confidence"]:

# extract the index of the class label from the

# `detections`, then compute the (x, y)-coordinates of

# the bounding box for the object

idx = int(detections[0, 0, i, 1])

box = detections[0, 0, i, 3:7] \* np.array([w, h, w, h])

(startX, startY, endX, endY) = box.astype("int")

# draw the prediction on the frame

label = "{}: {:.2f}%".format(CLASSES[idx],

confidence \* 100)

cv2.rectangle(frame, (startX, startY), (endX, endY),

COLORS[idx], 2)

y = startY - 15 if startY - 15 > 15 else startY + 15

cv2.putText(frame, label, (startX, y),

cv2.FONT\_HERSHEY\_SIMPLEX, 0.5, COLORS[idx], 2)

tts()

# show the output frame

cv2.imshow("Frame", frame)

key = cv2.waitKey(1) & 0xFF

# if the `q` key was pressed, break from the loop

if key == ord("q"):

break

# update the FPS counter

fps.update()

# stop the timer and display FPS information

fps.stop()

print("[INFO] elapsed time: {:.2f}".format(fps.elapsed()))

print("[INFO] approx. FPS: {:.2f}".format(fps.fps()))

# do a bit of cleanup

cv2.destroyAllWindows()

vs.stop()

**REFERENCES**

[1] Durette, B., Louveton, N., Alleysson, D., and H´erault, J, 2008. Visuoauditory sensory substitution for mobility assistance: testing The VIBE. In Workshop on Computer Vision Applications for the Visually Impaired, Marseille, France.

[2] Hern´andez, A. F. R. et al, 2009. Computer Solutions on Sensory Substitution for Sensory Disabled People. In Proceedings of the 8th WSEAS International Conference on Computational Intelligence, Manmachine Systems and Cybernetics, pp. 134–138.

[3] Tang, H., and Beebe, D. J., 2006. An oral tactile interface for blind navigation. IEEE Trans Neural Syst Rehabil Eng, pp. 116–123.

[4] Auvray, M., Hanneton, S., and O’Regan, J. K., 2007. Learning to perceive with a visuo - auditory substitution system: Localisation and object recognition with ’The vOICe’. Perception, pp. 416–430.

[5] Kammoun, S. et al, 2012. Navigation and space perception assistance for the visually impaired: The NAVIG project. In IRBM, Numro spcial ANR TECSAN, 33(2), pp.182–189.

[6] S. N. Kulkarni and S. Kumar Singh, "Object Sorting Automated System using Raspberry Pi," 2018 3rd International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, 2018, pp. 217-220, doi: 10.1109/CESYS.2018.8724056.

[7]R. Kumar and S. Meher, "A Novel method for visually impaired using object recognition," 2015 International Conference on Communications and Signal Processing (ICCSP), Melmaruvathur, 2015, pp. 0772-0776, doi:10.1109/ICCSP.2015.7322596.

[8]V. K. Bhanse and M. D. Jaybhaye, "Face Detection and Tracking Using Image Processing on Raspberry Pi," 2018 International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, 2018, pp. 1099-1103, doi: 10.1109/ICIRCA.2018.8597246.