NAVIGATION SYSTEM FOR ASSISTING THE BLIND

A PROJECT REPORT

SUBMITTED BY

Vishnu Remesh (14BCE1059)

Harish T (14BCE1154)

Jude Osbert K (14BCE1023)

in partial fulfilment for the award of the degree of

BACHELOR OF TECHNOLOGY

in

COMPUTER SCIENCE AND ENGINEERING

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School Of Computer Science and Engineering

Vandalur- Kelambakkam Road, Chennai- 600 127

April 2018

School of Computing Sciences and Engineering

DECLARATION BY THE CANDIDATES

We hereby declare that the Project entitled " Navigation system for assisting the blind " submitted by us to VIT, Chennai in partial fulfilment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a record of bonafide project work undertaken by us under the supervision of Dr.Aparna V. I further declare that the work reported in this report has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Place: Chennai   Signature of the Candidates

Date:

VIT®

Vellore Institute of Technology

(Deemed to be University under section 3 of UGC Act, 1956)

School of Computing Sciences and Engineering

CERTIFICATE

This is to certify that the Report entitled “**Navigation system for assisting the blind”** submitted by **JUDE OSBERT(14BCE1023) HARISH T(14BCE1154) and VISHNU REMESH(14BCE1059)** to VIT, Chennai in partial fulfilment of the requirement for the award of the degree of Bachelor of Technology in Computer Science and Engineering is a bona-fide record carried out under my guidance. The project fulfils the requirements as per the regulations of this university and in my opinion meets the necessary standards for submission. The contents of this report have not been submitted and will not be submitted either in part or in full, for the award of any other degree or diploma in this institute or any other institute or university.

Guide

Name:

Date:

External Examiner	Internal Examiner	Internal Examiner
Signature:	Signature:	Signature:
Name:	Name:	Name:
Date:	Date:	Date:

ACKNOWLEDGEMENT

We wish to express our sincere thanks to a number of people without whom we couldn't have completed the Thesis successfully.

We would like to place on record our deep sense of gratitude and thanks to our guide Dr. Aparna V, SCSE, VIT, Chennai campus whose esteemed support and immense guidance encouraged us to complete the project successfully.

We would like to thank our program chair Dr. B Rajesh Kanna and Co-Chair, Dr. C Sweetlin Hemalatha, B.Tech CSE and Project Co-ordinator Dr. B V A N S S Prabhakar Rao, VIT Chennai Campus, for their valuable support and encouragement to take up and complete the thesis.

Special mention to our Dean Dr. Vaidehi Vijayakumar, Associate deans Dr. Vijayakumar V and Dr. Subbulakshmi T, SCSE, VIT Chennai Campus, for spending their valuable time and efforts in sharing their knowledge and for helping us.

We thank our management of VIT, Chennai Campus for permitting us to use the library resources. We also thank all the faculty members for giving us the courage and the strength that we needed to complete our goal. This acknowledgement would be incomplete without expressing whole-hearted thanks to our family and friends who motivated us during the course of our work.

Jude Osbert K (Reg. No:14BCE1023)

Harish T (Reg. No:14BCE1154)

Vishnu Remesh (Reg. No:14BCE1059)

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

A major problem faced by the visually impaired these days is the lack of a simple small and accurate navigation system that can help them with their day to day lives. Guide dogs and white canes has limited potentials while a system integrating modern technologies like machine learning has a vast scope.

This project is a blind navigation system that helps the visually impaired people with their navigation and surroundings. Unlike other conventional systems that helps the blind ours is much better in the sense that this is faster and more accurate than those systems. Other systems using machine learning takes time to compute objects and give results which can be a problem as split second decisions have to be made in case of an imminent accident. Our system solves this issue by using two circuits where the secondary circuit does the machine learning and object detection while the primary circuit uses proximity sensors and an accelerometer to find out if an imminent accident will take place using an Arduino. This will then send an interrupt to the Raspberry Pi (Secondary Circuit) and a decision is made. This is much faster as there is no high end processing. Now, the system provides real time audio instructions to the user using an earpiece so that the responses are not limited like the smart walker which uses vibrations to convey the message. Since the Raspberry Pi uses image detection and classification the user will be able to get a better clearer understanding of his surroundings rather than be told an object is present this device can give a detailed reply saying what the object is. Also, the cost of production is comparatively less and can further be reduced if the Raspberry Pi and the Arduino can be replaced by embedded systems.

1.INTRODUCTION

Our system works the same way a human body works, there is an immediate response system for responding to "reflexes" like the spinal cord which is basically an array of sensors combined with an Arduino that calls an interrupt which lets the Raspberry Pi to send an audio output to the user. The secondary system is a combination of a camera, sensors and machine learning algorithms that gives the user an exact idea of his surroundings, like a brain. This system takes more time compared to the logical circuit and so, all the immediate decisions are made by the detector circuit just like the human body. This data is then sent to the brain (Raspberry Pi) and it is processed.

1.1.OBJECTIVE

Our objective is to develop a better cheaper more accurate blind navigation system that could someday replace guide dogs and white canes and set standards for the assistance of the visually impaired. This system should be able to accurately identify and help the user perceive his/her surroundings and be able to convey this to the user in real time so that he/she can avoid accidents.

1.2.MOTIVATION

The main problems faced by a visually impaired person today is the lack of a reliable low cost portable navigation system. People are very unlikely to buy half baked inaccurate systems since they will mostly depend on this system for assistance and a single flaw could mean an accident. Current systems like the smart walker uses path finding algorithms and robotics to provide help it doesn't account for dynamic changes in the environment. Another system was 'Drishti' an indoor and outdoor navigation system but this was costlier due to a lot of hardware and was also less flexible. So the need for a better trustworthy system grew and we still don't have a worthy system for the blind.

An estimate number of 285 million people are visually impaired and they still don't have a promising system that can be used instead of being at the mercy of others.

1.3.BACKGROUND

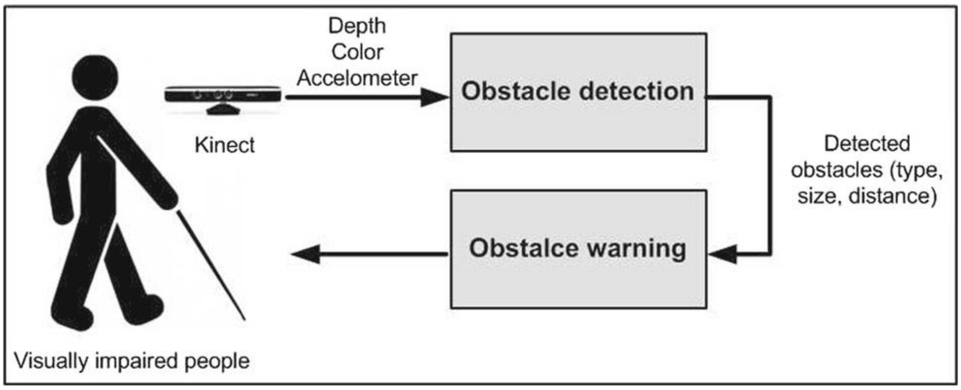
1.3.1.EXISTING SYSTEMS

1.3.1.1.OBSTACLE DETECTION AND WARNING USING ELECTRODE MATRIX AND KINECT

A paper by Van Nam Hoang, Thi-Lan Le and four others titled Obstacle detection and warning system for visually impaired people based on electrode matrix and mobile kinect has two components. The first component captures the environment using the kinect and analyze it to detect the predefined obstacles while the second component tries to represent obstacle's information under the form of an electrode matrix.

Their proposed system had two modules namely obstacle detection and obstacle warning. The obstacle detection part focused on determining the presence of obstacles in front of the user. The obstacle warning part on the other hand represents and sends this information to the user. In this paper they used a Kinect with a battery and it was connected to a laptop in the user's backpack. The laptop was running the obstacle detection program. The prototype had a laptop an RF transmitter and a belt anchor that holds the Kinect. This system is very bulky and heavy considering a user will have to carry this all around the places he visits it will be quite tiresome.

Also, their system uses a point cloud to figure out an object from the kinect data and uses this information for object detection. This process is outdated and unreliable compared to machine learning algorithms that are currently available that ca detect and classify the objects in less time.



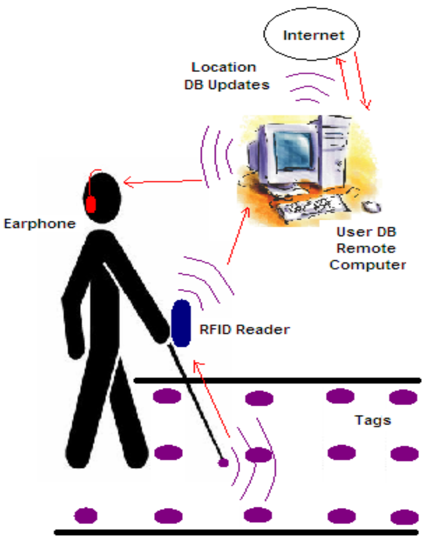
The above image shows how their system intend to work. This system as mentioned was clumsy, too heavy and also doesn't account for dynamic changes in the environment. This would result in the person backing off from a lot of situations provided the point cloud takes the 360 degrees into account. Using a laptop would also mean that a lot of potential is being wasted as a laptop consumes a lot of power even for running a simple object detection program.

Another factor is the production cost. This entire setup will cost a lot as the kinect alone will cost around 20000 rupees and other sensors will add to the cost. This is excluding the cost of the laptop. So in effect their experimental setup costs more than 50000 rupees. This doesn't help the poor as this wil be a luxury for them and they would prefer cheaper modes like guide dogs.

1.3.1.2.RFID BASED NAVIGATION SYSTEM

A paper by Kanchan Varpe and M P Wankhade titled Survey of Visually impaired assistive system proposed an idea of using electronic tags placed under the tiles of the path and the blind person has a reader on him that reads the tag and sends the data to a database that contains the identity and location of the tags. This database returns the current position of the user and the user hears it as an audio output.

This mechanism is possible only in small areas as a lot of RFID tags has to be placed inside the floor for the system to work.



The above image explains the working of the system where the data is read by the reader on the person the data is transmitted to a remote computer using wireless communication techniques and then the computer executes a query that fetches the required data from the database. This data is then transmitted to the user and the user hears it through the earphones.

This system has a lot of problems mainly it is costly and tiresome to place RFID tags all over a place and maintain all the data in a database. This will be costly and have to be planned ahead before the pavement is made so that it can be embedded in the pavement which will cause less wear and tear. Also, less tags would mean constraints on the person's movement while a lot of tags would mean that the database will take time to fetch the data which will decrease accuracy.

Another a major problem is the time lag. Since the system uses wireless communications to transmit the data, It would take time and in case of blind navigation time is everything since by the time the person receives the information it would no longer be valid as he would have already crossed that landmark.

So, although an RFID tag costs around 20 rupees and an RFID reader around 1000 rupees the system wouldn't be viable as a lot of tags has to be purchased and planted in the pavements and streets. then, the data has to be systematically stored in a database and fetched with the minimal time lag using the best queries possible. This would also mean that the server containing the database should be up and running the whole time and delays are not acceptable. So, such a system is impractical and time consuming and also has a large error margin in developing countries where the price for uninterrupted fast data is high and poor people can't afford it. This would mean that this system would be viable only if the government placed the tags and created and maintained the database. Also, a rich person can implement this inside his house and maintain a private database that will help him navigate indoors.

1.3.1.3.SMART WALKER

A paper by Wachaja, A., Agarwal, P., Zink, M., Adame, M. R., Möller, K., & Burgard, W. titled Navigating blind people with walking impairments using a smart walker uses a walker which is clumsy and was for a specific population- people with blindness and walking impairment. The system of the smart walker gives instructions using vibrations and so the responses will be limited. This system further relies on robotics and path finding algorithms as opposed to machine learning.

In this system they used a walker incorporated with LIDAR sensors, processing capabilities and vibrating motors on each handles to give the output. They use two planar laser range finders for perception and estimation of the ego motion. The first laser scanner is fixed with respect to the walker. We calculate the ego motion of the walker based on the measurements of this sensor by laser scan matching. The second laser scanner is continuously tilted by a servo motor to sense the three-dimensional environment. We fuse the ego motion estimation with the measurements of the second scanner and data from its motor to obtain a dense three-dimensional point cloud.

A point cloud as discussed had a lot of defects as compared to machine learning. It lacks decision making during dynamic subject changes and in case of a blind person there is always dynamic changes like a person walking or a car moving etc. So, such a system will be less preferable by the blind. Also, a walker is a clumsy bulky tool and is not preferred by the blind. They prefer something compact and wearable. This also is costly and since it uses robotics and path finding is outdated and useless compared to machine learning and image classification.

People prefer faster better system and this is not one as it takes time to compute and find a path. Also this system doesn't recognize the objects but just knows that there is an object. That is, unlike image classifiers that tells you what the object is this system just tells you how to avoid it. So the user wouldn't know if this was the object he had to interact with.

Another major problem is the communication issue where this uses vibration motors on the handles to inform the user as compared to audio signals. This limits the information passed to the user and thus increases error and danger. Using an audio output would further increase the communication between the user and the machine and would help better in guiding the person.

1.3.1.4.NAVIGATION USING RFID AND GPS

A paper by Chaitali K. Lakde and Dr. Prakash S. Prasad titled Review Paper on Navigation System for Visually Impaired People discussed a lot of alternatives that ranged from white canes to RFID aided navigation systems. Some of the systems used are given below.

- A framework "Roshni" decides the client's situation in the building, route through sound messages by keys on the portable unit. It utilizes sonar to distinguish the position of client by mounting ultrasonic modules on roof at regular intervals. This framework is compact, simple to work and isn't influenced by natural changes. In any case this framework is restricted just for indoor route since it requires inside guide of the building.

- RFID based map-reading system which provides technical solution for the visually impaired to pass through public locations easily using RFID tag grid, RFID cane Reader, Bluetooth interface and personal digital assistance. But its initial development cost is quite high and chances of interference in heavy traffic.

- A voice operated outdoor navigation system developed using GPS, voice and ultrasonic sensor. It can alert user’s current position and provide verbal guidelines for travelling to a remote destination but fails to give obstacle detection and warning alert.

- Another real-time technology developed to alert visually impaired user by the presence of static / dynamic obstacles in a few meters surrounding, which works without depending on any Smartphone, uses camera for background motion detection. This system is robust to complex camera and background motion and does not required any prior knowledge about the obstacle size, shape or position. This camera based image processing system can be a better option but it requires lot processing power and hence system becomes bulky, costly and it must be transportable.

All these systems had some flaws that would make the user doubt the system. Since complete truest is required for the blind these systems failed.
some other systems that are used is given below.

DRISHTI: Helps the blind navigate through both indoor and outdoor environments but overlooks the fact that a blind person doesn't leave his comfort zone. That is, he/she always has a grocery shop, a hair salon etc in his mind and will not use the GPS system often. Also a lot of hardware means it becomes difficult to repair or replace a component.

MICROSOFT'S NAVIGATION TECH: Same drawbacks as the Drishti as it overlooks the fact that a blind person is very protective and careful and does not just try a new place because an app recommended it. By far this is the most relatable systems but it uses maps and an active internet connection.

2.TECHNICAL SPECIFICATIONS

2.1.HARDWARE

This section lists the specifications of various hardware components used by us during the creation of our project. We have used proximity sensors and Arduino in the primary circuit and Raspberry Pi and camera and a stepper motor in the secondary circuits. We also have an accelerometer with the primary circuit to determine the absolute speeds of the objects instead of relative speeds.

2.1.1.RASPBERRY PI3

SoC: Broadcom BCM2837 (roughly 50% faster than the Pi 2)

CPU: 1.2 GHZ quad-core ARM Cortex A53 (ARMv8 Instruction Set)

GPU: Broadcom VideoCore IV @ 400 MHz.

Memory: 1 GB LPDDR2-900 SDRAM.

USB ports: 4.

Network: 10/100 MBPS Ethernet, 802.11n Wireless LAN, Bluetooth 4.0

Ethernet Port: 1 Port

SD Card: 1 Micro SD-Card Slot

2.1.2.ARDUINO UNO

Microcontroller: ATmega328

Operating Voltage: 5V

Input Voltage (recommended): 7-12V

Input Voltage (limits): 6-20V

Digital I/O Pins: 14 (of which 6 provide PWM output)

Analog Input Pins: 6

DC Current per I/O Pin: 40 mA

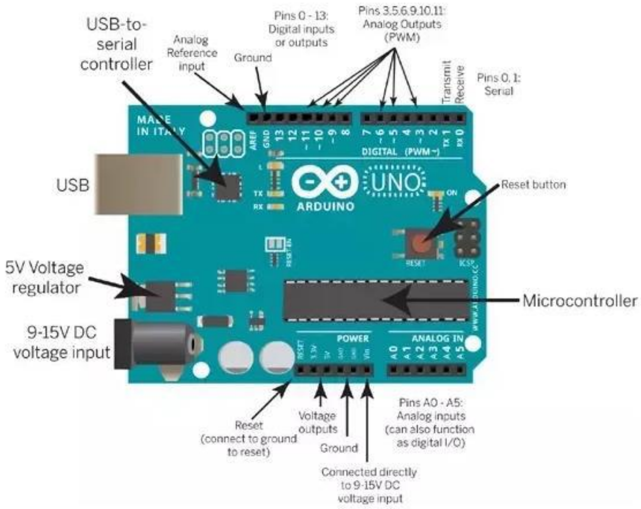
DC Current for 3.3V Pin: 50 mA

Flash Memory: 32 KB of which 0.5 KB used by bootloader

SRAM: 2 KB (ATmega328)

EEPROM: 1 KB (ATmega328)

Clock Speed: 16 MHz



2.1.3.STEPPER MOTOR SM57STH76-2804A

Rated voltage: 3.2V

Current/Phase:2.8A

Resistance/Phase:1.13Ohms

Inductance/Phase:3.6mH

Holding Torque:18.9kg-cm

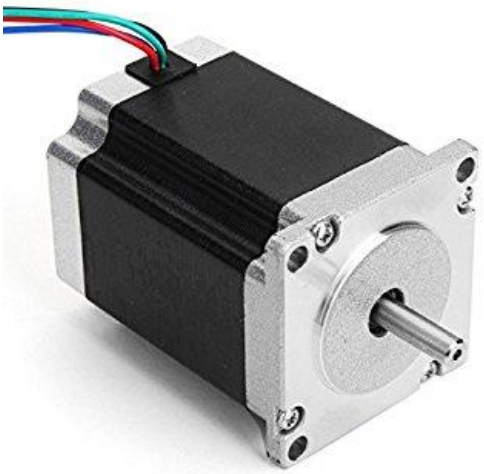
Number of leads:4

Rotor Inertia:480g-cm2

Weight:1kg

Detent Torque:0.68kg-cm

Length:76mm



2.1.4.DRV8825 STEPPER DRIVER MOTOR

8.2-V to 45-V Operating Supply Voltage Range

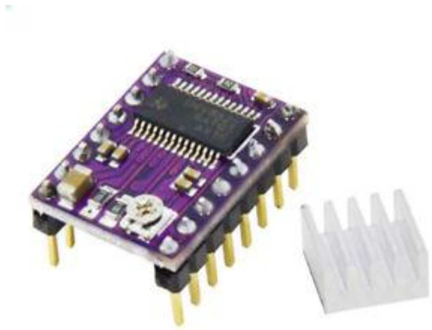
2.5-A Maximum Drive Current at 24 V and TA = 25°C

Built-In 3.3-V Reference Output

ƒSTEP Step frequency 250 kHz

tWH(STEP) Pulse duration, STEP high 1.9 µs

tWL(STEP) Pulse duration, STEP low 1.9 µs



2.1.5.HC-SR04 ULTRASONIC DISTANCE SENSOR MODULE

Operating Voltage: 5V DC

Operating Current: 15mA

Measure Angle: 15°

Ranging Distance: 2cm - 4m

2.1.6.RASPBERRY PI CAMERA BOARD

Weight:3g

Still resolution:5 Megapixels

Video modes:1080p30, 720p60 and 640 × 480p60/90

Linux integration:V4L2 driver available

C programming API:OpenMAX IL and others available

Sensor:OmniVision OV5647

Sensor resolution:2592 × 1944 pixels

Sensor image area:3.76 × 2.74 mm

Pixel size:1.4 µm × 1.4 µm

Optical size:1/4"

2.1.7. MPU-6050 GYRO SENSOR 2 + ACCELEROMETER

VDD: 2.375V-3.46V

VLOGIC: 1.71V to VDD

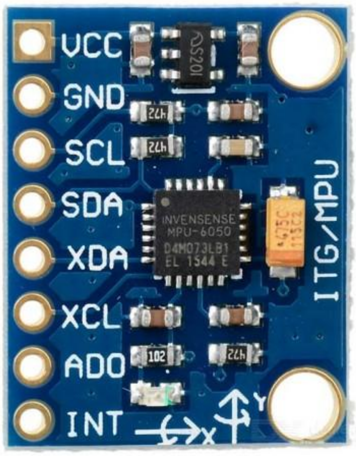
Serial Interfaces Supported :I 2C

Pin 8:VLOGIC

Pin 9: AD0

Pin 23: SCL

Pin 24: SDA



3.PROJECT DESCRIPTION AND GOALS

3.1.FUNCTIONALITIES

The following are the main functionalities of this project.

- To detect objects around the user and classify them.
- To find a dynamic path for the user by avoiding these obstacles.
- To avoid accidents by warning the user of imminent threats.
- To give continuous audio feedback to the user regarding surroundings.

3.2.PRIMARY CIRCUIT

The primary circuit consists of proximity sensors and an Arduino board with the ability to detect objects travelling towards them with certain speeds as a threat and calls an interrupt to the Raspberry Pi. This circuit is much faster compared to the Pi and the Machine Learning algorithm which will take time. In this case the Arduino can take split second decisions which is required in the case of a vehicle coming towards the user. Here, the proximity sensors collect the data and sends the data to the Arduino where the Arduino combines the data with the Accelerometer data to find out the absolute speeds of the objects. The Arduino takes consecutive distance values and calculates the speed. This speed is used to determine if the object will crash with the user or not. This will help avoid accidents in situations when the machine learning part is computing and hasn't come up with a solution yet in such a case the Arduino sends an interrupt to the Raspberry Pi and the Pi in turn sends an audio output corresponding to the interrupt. This feature is the one that makes this system unique in the way that decisions are made faster and safer than other systems.

3.3.SECONDARY CIRCUIT

This circuit contains the Raspberry Pi and the camera modules that takes images of the surroundings and uses the images and machine learning algorithm to perceive the environment. that is it helps the user know his surroundings and make decisions accordingly. For the machine learning part we will be needing object classification, object detection, counting the number of objects etc which has to be implemented on the Raspberry Pi. For object detection we will be needing a large data set, fortunately such sets are available online like ImageNet and COCO which has a combined data set of around 570000 images. All this can be done using python and TensorFlow- Google's cloud platform. We can use other features like face detection to tell him/her who the person in front of them is, thus helping them to interact well in social circles and giving them more than the current systems. It is possible for us to create classifiers and train the machine to do all these functionalities so as to help the blind.

3.4. TECHNOLOGIES USED

3.4.1.MACHINE LEARNING

Machine learning is a field of computer science that gives a system the ability to learn progressively to improve performance on a specific task with data without being explicitly programmed. Basically the machine is given training data and based on this data the machine will be able to predict the results of the data that follows. We are using machine learning for the object detection part of our project where an object has to be accurately identified for a blind person to avoid or interact. We are using the TensorFlow software for machine learning.

3.4.2.TENSORFLOW

TensorFlow is an open-source software library for dataflow programming across a range of tasks developed by the Google Brain Team. It is used for machine learning applications and we are using Python's TensorFlow library for object detection and classification. The model was trained using COCO training model.

3.4.3.COCO TRAINING SET

COCO is a large scale object detection, segmentation and captioning dataset with features like object segmentation and over 330 thousand images of which more than 200 thousand of them are labelled. We are using this dataset to train our machine learning algorithms written in Python using the TensorFlow library.

3.4.4.RASPBERRY PI

A Raspberry Pi is a credit card sized computer. Its low cost and small size makes it an acceptable product for a lot of low cost products. Here, we are using the Raspberry Pi as the main processing system where all the images are classified and our algorithms are executed. The Raspberry Pi we are using is Pi 3 Model B.

3.4.5.ARDUINO AND PROXIMITY SENSORS

Arduino is a single board microcontroller equipped with sets of digital and analogue pins that can be interfaced with the proximity sensors on a breadboard. This proximity sensor data is taken and processed in the Arduino and if required triggers an interrupt on the Raspberry Pi.

3.5.ALGORITHMS USED

3.5.1.OBJECT DETECTION

After the camera captures the image it is saved to a folder. Whenever a change is made to the folder the script is executed where the python file for object detection is called. In object detection the machine analyses the image and using a pre trained model finds out the objects in the image. This algorithm can detect up to 20 objects in an image and classify them. This image is then passed onto the depth perception function so that the relative distances of each object is computed.

3.5.2.DEPTH PERCEPTION

After the objects are identified in the image we have to know the depth of the objects from the camera, for this we create a disparity map to find the relative depths of each object. With the object detected and the disparity map we can find the relative distances of the objects from the user. That is, we find the objects and how close they are so that a possible path could be found for navigation of the user. The pseudo code is as follows

Create a "minSSD" array equal to the size of the image, with large initial values.

Create a "disparity" array equal to the size of the image.

for k = 0 to MAX_SHIFT do

Step 1: Shift right i mage to the right by k pixels

Step 2: Perform Sum of Squared Differences (SSD) b etween left image and shifted right image

Step 3: Update the minSSD and disparity array.

for each pixel coordinate (i,j) do

if ssd(i,j) < minSSD(i,j) do

minSSD(i,j) <= ssd(i,j)

disparity(i,j) <= k

end

end

end

3.5.3.PATH FINDING

After the relative depths of the objects are found we now have the object dimension the x and y coordinates of the object and a relative depth. With these as the input the algorithm works to find the best path for the user to take. The algorithm computes the path each time a new image is taken and so a dynamic path could be found. Whenever a new image is obtained the function receives a set of values describing the length and breadth of the image, the x and y coordinates and a number corresponding to the relative depth. Using this a path is found that lets the user navigate through these mazes. Here, the algorithm computes the path using the following pseudo code

```
if (new image in folder)
{
object detect();
depthfind();
create an n*5 array (5 for x, y, length, breadth and relative depth)
}
array passed to the pathfinder function
for(x, y not in array x, y being the current position)
{
if(x+1 not in array)
move forward;
else
if(y+1 not in array)
move left;
else
if(y-1 not in array)
move right;
} this should be executed indefinitely until a new image is accepted and a new array is generated.
```

4.DESIGN APPROACH AND DETAILS

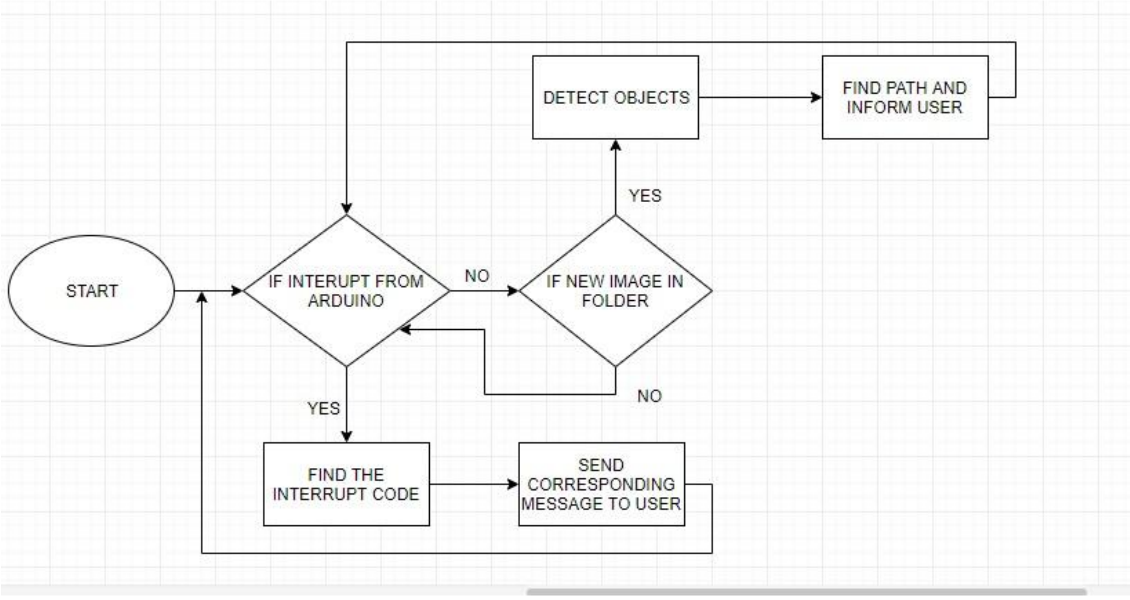
This system as explained above has two parts. The primary circuit having the Arduino and the proximity sensors work like the spinal cord in our body that takes immediate actions like warning the user of an accident or something that requires his immediate attention. This works by the proximity sensors finding out the difference in distances so as to figure out if an object is about to hit you or will stop before that. This is done to avoid unnecessary interrupts to the Raspberry Pi as it will cause delays in the normal working of the system.

The primary circuit consists of four proximity sensors facing each of the four directions so as to capture data from all around the user. An accelerometer is also provided in this circuit so that the speed of the user can be determined and hence the absolute speeds of the objects can be determined. This data is the sent to the Arduino board that takes the data and processes it to figure out if it is an immediate threat that has to be alerted or a normal change in the environment. This data if important triggers an interrupt in the Raspberry Pi so as to take an immediate action like warning the user of the threat and providing countermeasures. Such a circuit can help the user much better as unlike other systems this will be faster at responding to the threat thus making the entire system more faster and hence effective.

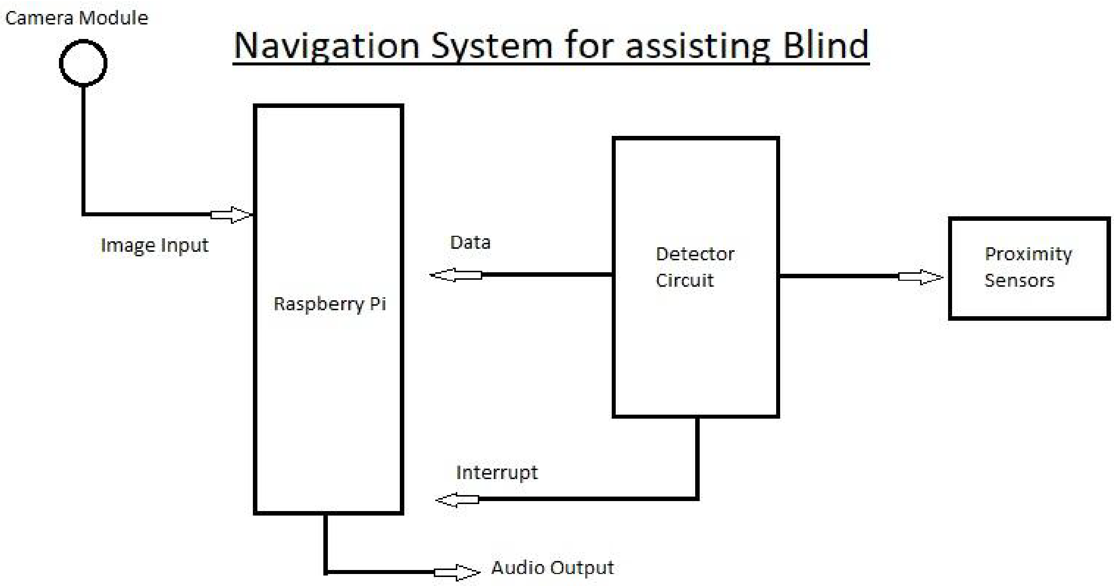
The secondary circuit normally works by running a python script that checks for any changes in the images folder which stores the images captured by the camera. Thus, when the camera captures a new picture the script calls a function to identify the objects in the image. The camera module is mounted on a stepper motor so that the system is able to get images from all around the user. There will be four threads corresponding to each direction running in the Raspberry Pi at any instant. Now, when the camera captures a new image of the left side then the thread corresponding to the left side is triggered and it finds all the classifiable objects in the image. This is done continuously and the classified images are given to the user as audio outputs until it encounters an interrupt from the Arduino. This will make the Raspberry Pi give an emergency audio output so that a quick action can be taken by the user. Once the process is complete the system resumes the threads and works as usual. The system can also provide navigation with stereo camera setup and will help the user navigate through traffic and busy streets. The current algorithm we used can detect about twenty objects in a frame including cars, humans, plants etc. Almost everything a normal blind person will encounter is in the training data and the system can recognize these objects with precision. After object detection the distance to the object is calculated and a possible navigation route is computed. This path is then given to the user as audio outputs and the algorithm runs each time a change in the image is detected. This will help the user avoid being stuck or colliding with objects. Since a visually impaired person knows almost all the necessary things in his locality this navigation helps him navigate through dynamically changing environments i.e; it doesn't let the user enter a destination or use GPS it simply calculates the best route based on where the user is standing. So, if the person has to go to the grocery store the system will not give him directions to the grocery store since the user knows the way to the store. The system provides dynamic routes while he is on the pavement by providing a path through busy streets filled with people and vehicles.

4.1. DIAGRAMS

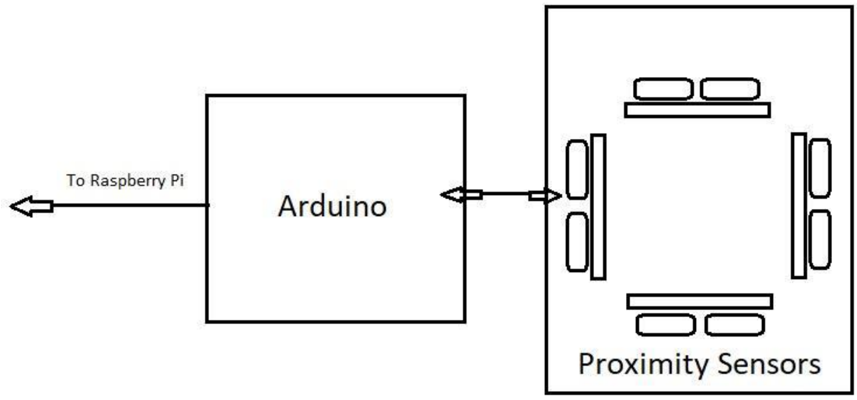
4.1.1.FLOW CHART



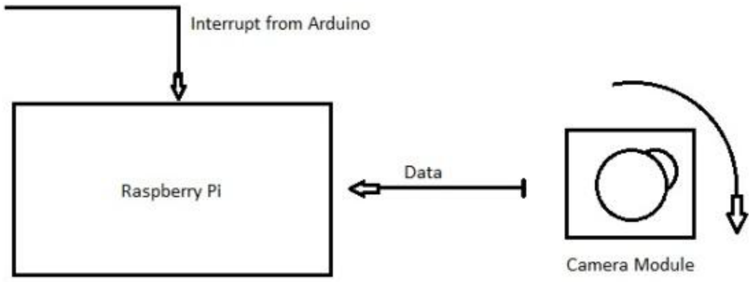
4.1.2.ENTIRE CIRCUIT



4.1.3.PRIMARY CIRCUIT

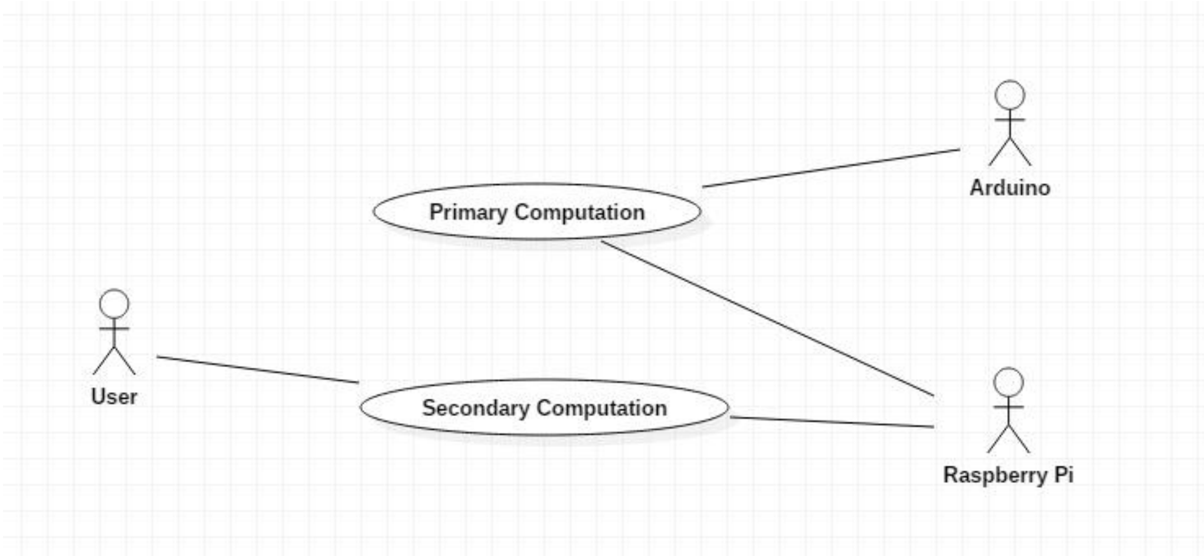


4.1.4.SECONDARY CIRCUIT

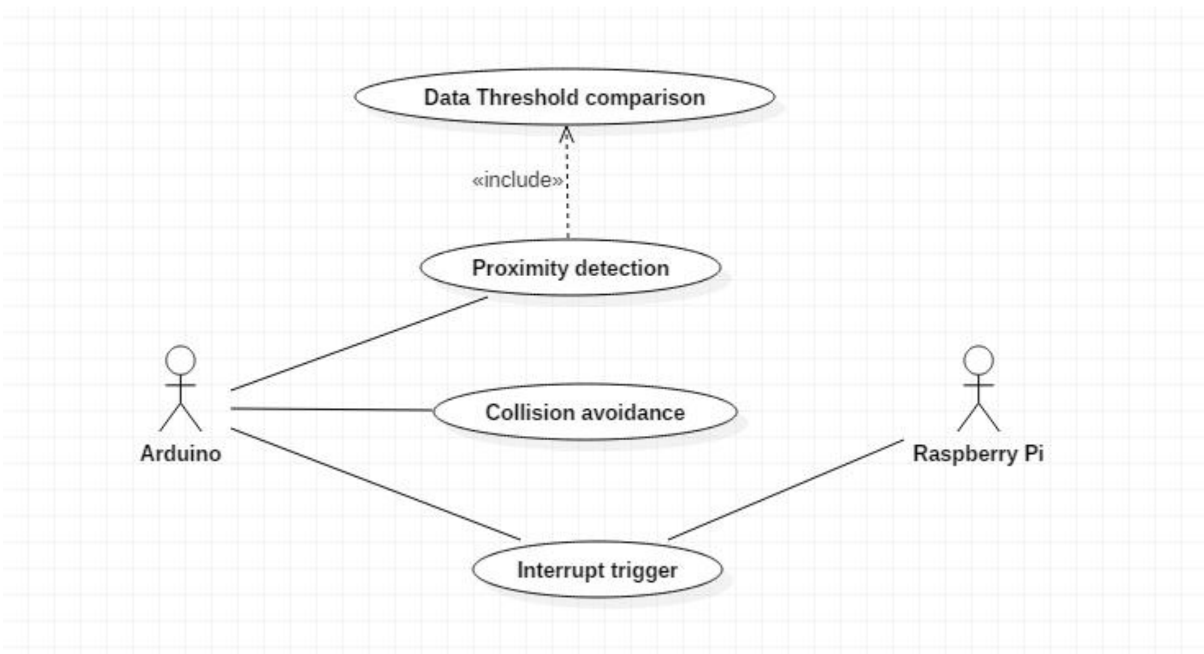


4.2.USECASE DIAGRAMS

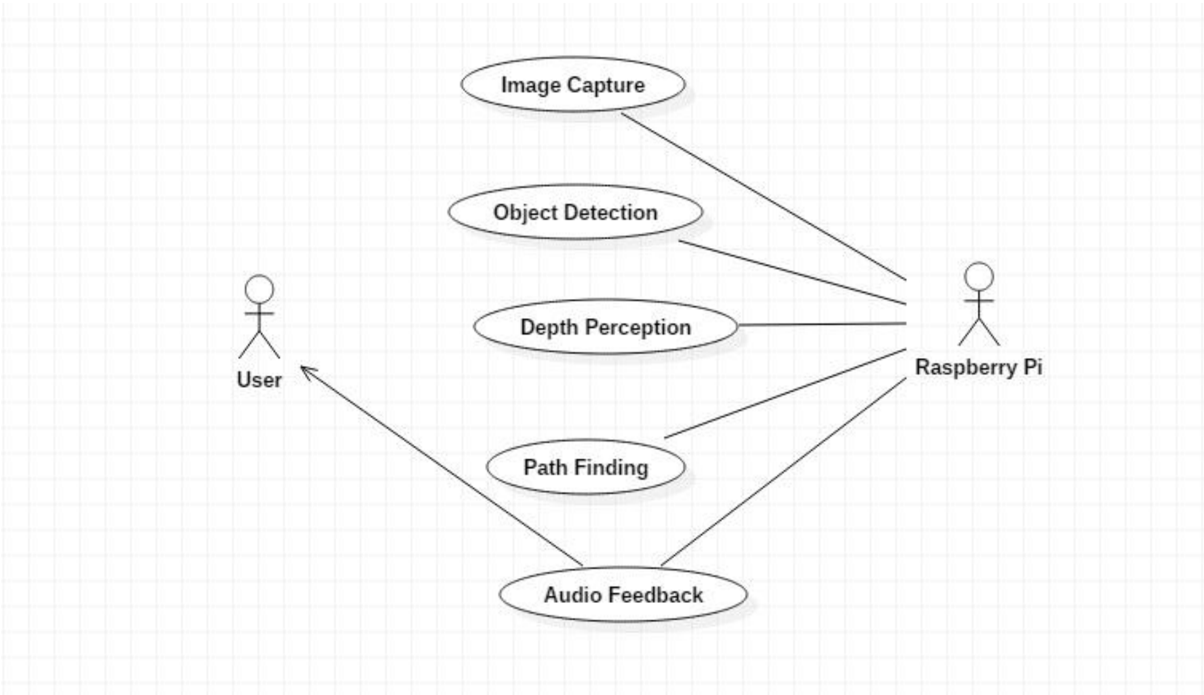
4.2.1.ENTIRE SYSTEM



4.2.2.PRIMARY CIRCUIT



4.2.3.SECONDARY CIRCUIT



4.3.CONSTRAINTS AND ALTERNATIVES

4.3.1.CONSTRAINTS

A lot of challenges were faced during the creation of this project. Some of those problems were rectified but others had to be given an alternative. Some of them are listed below.

- A single camera meant that we could not get the absolute depth of an object from the user. Such a problem could be solved by using stereo vision like humans. In this case a secondary camera had to be added this would have increased the amount of components the user will have to carry and would have increased the price. So an alternative of using disparity map to find relative distances was used where we could know which object was nearer but wouldn't get an exact distance.

- Using a stepper motor was better than using four different cameras in the four directions. Using a stepper motor decreased the cost significantly but a weight of one kilogram excess was added to the user as a result. This compromised user easiness but increased efficiency and lowered the cost.

4.3.2.ALTERNATIVES

The Arduino and the Raspberry Pi could in theory be replaced by embedded systems that can be tasked to do this specific operations instead of wasting a full fledged computer's potential. This is advisable as this will reduce the cost of production significantly. Also, integrating the phone with this system by using an Android/iOS app would further help as we would get a better camera, the drawback being that the user will have to manually point the camera in the right direction.

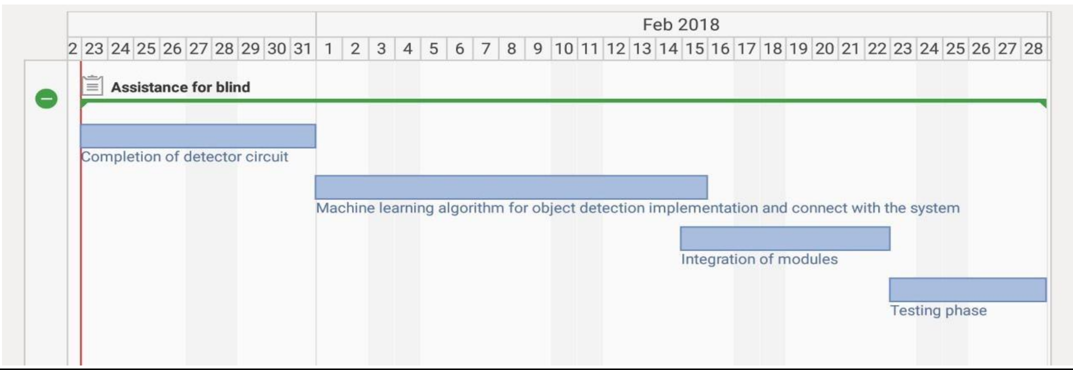
5.SCHEDULE, TASKS AND MILESTONES

There were three major tasks and several small ones for this project to be operational. The three major tasks were:

- Design and implementation of the detector circuit (Primary Circuit)
- Development of the Object detection and path finding algorithms
- integration and automation of the modules

The tasks were divided and given to each team member and was completed as per the Gantt Chart given below

5.1.GANTT CHART



6.DEVELOPMENT LIFECYCLE

6.1.REQUIREMENT ANALYSIS

One of the main steps involved in creating the system was to identify the requirements of the user and act accordingly. So the following requirements were found using various methods like asking the visually impaired on their expectation for a blind navigation system.

- It should be able to detect objects near them and identify them so that they can interact with them accordingly.

- It should be able to identify a path for them to traverse in a street by avoiding the obstacles and the path should dynamically change according to the changes in the environment.

- It should be able to detect the possibility of an accident and warn them in the fastest time possible so that they can take decisions accordingly.

- It should be able to communicate with them with ease and the responses shouldn't be limited and vague.

These requirements were then taken into account for deciding on which development model to use keeping in mind the time constraints and other challenges.

6.2.AGILE DEVELOPMENT METHODOLOGY

In order to create this project we used Agile development methodology this helped us in understanding the needs of the poor and develop the model. In Agile methodology we could create functionalities, develop them and then if not satisfactory add more. It is based upon the iterative and incremental models and focuses on the adaptability to the changing product requirements. Using Agile methodology we divided the entire project into smaller, easily developable features so that the product lifecycle would be more flexible. The following were the features that had to be developed.

- design and creation of the detector circuit or the primary circuit by taking into account the various possibilities and time requirements. Our first model was a logical circuit that however was scrapped and an Arduino was used. This made the circuit more flexible since whenever a change had to be made we didn't have to redesign the entire circuit, we just had to change the code on the Arduino

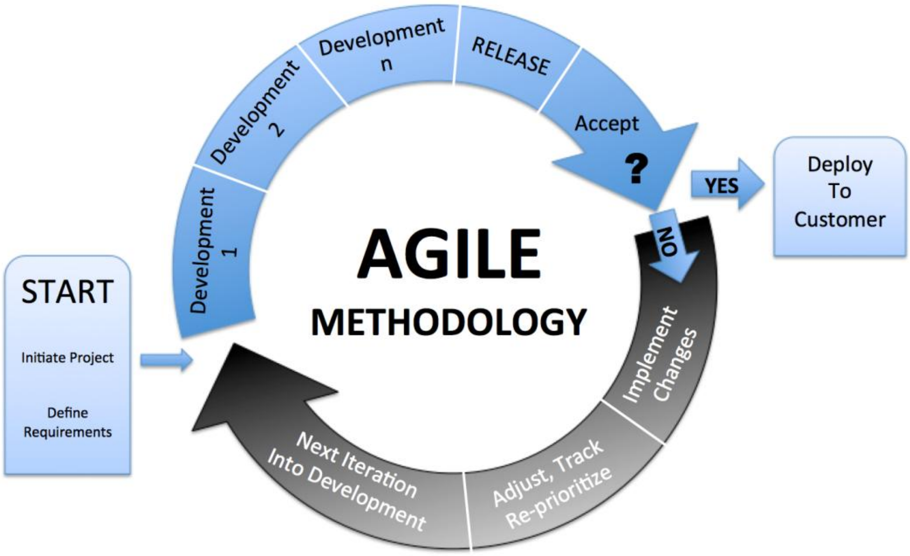
- Finding out the perfect algorithm and the training data so that the Machine Learning part could be trained to detect common objects that a blind person encounters in their everyday life.

- Creating an algorithm for finding the relative distances of the objects from the user since absolute distances required additional components.

- Creating an algorithm for using the position and relative distances of the objects to find an optimal path avoiding these objects.

- Providing real time audio output to the user regarding the surroundings and giving warnings in case of an impending accident.

An Agile methodology works like the figure given below



A lot of research and work had to be put in this project to mould it as it is today. Some of these wouldn't have been possible without the requirement analysis done. It told us what the people expects and what had to be done. Some of those are given below.

- The camera had to be taken on test runs to find out the optimal requirements for it to function properly. This was important as most of our system depends on getting a clear image from the camera. It also helped us understand the cameras limitations like at what speeds does the images start to blur etc. This has helped us in defining the limitations of the system.

- The machine learning algorithm had to be tested and timed so that the exact time required for the camera to capture the image and be free could be calculated. This data then had to be put into the stepper motor so that the motor would be at a standstill whenever the camera is capturing an image.
- The system had to be tested in actual conditions so that the actual response times and other important factors could be calculated and a conclusion could be made regarding the system's speed and accuracy. This was important as the person using it will have to trust the system with his life on the balance and so even a small error could mean catastrophe.

7.MARKETING AND COST ANALYSIS

7.1.MARKETING ANALYSIS

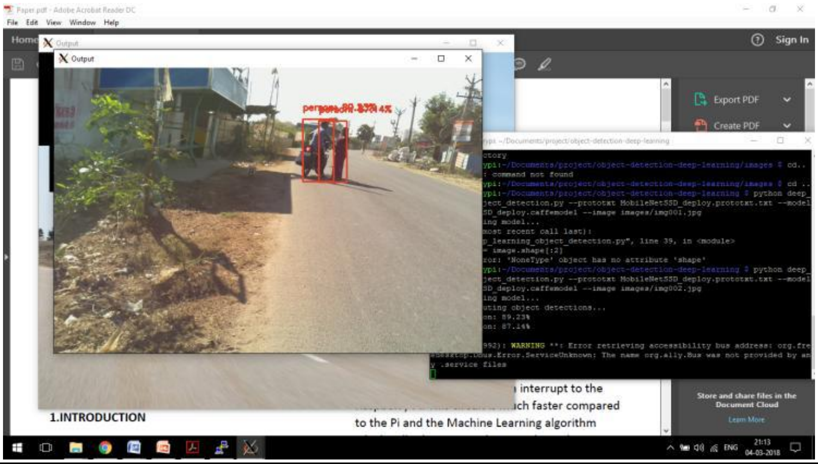
As mentioned earlier the existing systems in the market is flawed or too costly or has less features. Our system aims at providing top of the line product at a very low cost. Since there are no such devices in the market it is safe to say that once released it can definitely beat all other systems in existence.

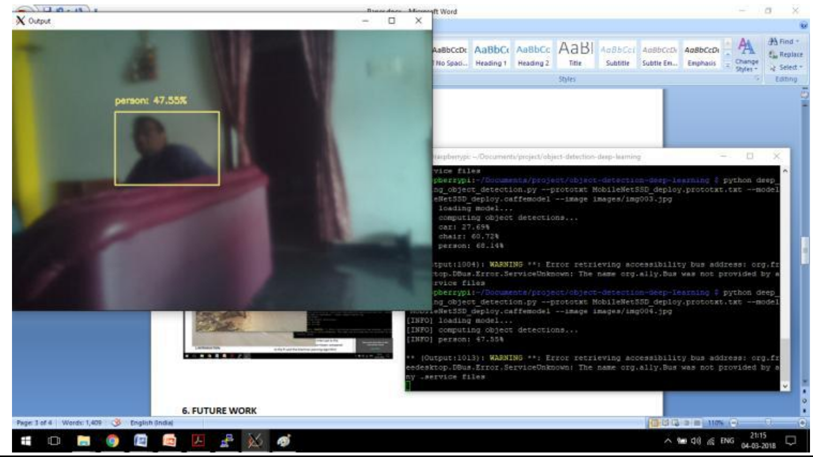
Our product will have a modular design which will make it more easier to repair and replace. Also since the software is loaded on a Raspberry Pi updates or changes can easily be incorporated this helps the system to be more flexible. Also, if mass produced the Pi and the Arduino can be replaced by embedded systems so that the cost can be reduced.

7.2.COST ANALYSIS

Item	**Cost**
Raspberry Pi Model B	Rs 2969
Raspberry Pi Camera Board	Rs 825
HC-SR04 Ultrasonic Distance Sensor	Rs 280*2 = 560
DRV 8825 Stepper Driver Motor	Rs 120
Stepper Motor SM57STH76-2804A	Rs 2140
Bread Board	Rs 139
TOTAL	**Rs 6753**

8.SCREENSHOTS





9.FUTURE WORK

This model can be extended to have more objects in the classifier so that the system will be able to detect more objects with accuracy. Moreover the system can be optimized by using low cost stepper motor or in the future use a 360 degree camera to have a better understanding of the surroundings. Furthermore, the system can be expanded to incorporate a GPS module and have navigation setup so that a person can speak his destination and the device will find dynamic routes to help him reach the destination. Also, while mass producing we can use embedded systems specifically built for this purpose rather than use an expensive Arduino. This will further reduce the cost and will benefit more people.

10.CONCLUSION

This system provides help to a visually impaired person. It relies on the person trusting the system and taking the instructions provided by the system seriously and obeying it without fail. Once the system is perfected it will aid the blind in a manner that wasn't possible before. It will be able to describe the surroundings as well as navigate him/her taking care no accidents takes place. This system ideally will be cheaper than the existing systems and will rely on dynamic processing to find paths rather than use an algorithm to find a path only to find out that the path is no longer viable. Also, if we use the phone camera to take images rather than use an external camera then we can get better quality images and use them instead of the menial quality photos. Also such integration with the phone will help reduce the costs as a lot of things can be obtained from it like GPS data. This will further reduce the cost and durability. This is a feasible alternative as a Smartphone is available with everyone these days.

This device can further be perfected to do a lot more functionalities and also have a more accurate description of the surroundings so that it will be a be a better system than the one presented by us.

11.REFERENCES

1.Wachaja, A., Agarwal, P., Zink, M., Adame, M. R., Möller, K., & Burgard, W. (2017). Navigating blind people with walking impairments using a smart walker. Autonomous Robots, 41(3), 555-573.

2.Martins, M. M., Santos, C. P., Frizera-Neto, A., & Ceres, R. (2012). Assistive mobility devices focusing on smart walkers: Classification and review. Robotics and Autonomous Systems, 60(4), 548-562.

3.MacNamara, S., & Lacey, G. (2000). A smart walker for the frail visually impaired. In Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on (Vol. 2, pp. 1354-1359). IEEE

4.Chaitali K Lakde., Dr. Prakash S. Prasad (2015).Review paper on Navigation system for visually impaired people. International Journal of Advanced research in computer and communication engineering.2278-1021

5.Kanchan Varpe.,M.P.Wankhade. Survey of Visually impaired assistive system(2013).International journal of Engineering and innovative technology(IJEIT).ISSN:2277-3754

6.Van-Nam Hoang.,Thanh-Huong Nguyen.,Thi-Lan Le.,Thanh-Hai Tran., Tan-Phu Vuong., Nicholas Vuillerme.,Obstacle detection and warning system for visually impaired people based on electrode matrix and mobile kinect.(2016)ISSN:2196-8896.

7.A.Helal, S. Moore and B. Ramachandran, Drishti:An integrated Navigation System for visually impaired and disabled(2001),5th International symposium on Wearable Computer, Zurich, Switzerland.

8.Johnson L A, Higgins C M.,A Navigation aid for the blind using tactile -visual sensory substitution.28th Annual Conference of the IEEE engineering in medicine and biology society.pp.6289-6292.(2006)

9.Rodrguez S A, Yabes J J, Alcantrailla P F, Bergasa L M, Almazan J, Cela A., Assisting the visually impaired: Obstacle detection and warning system by acoustic feedback. 17476-17496(2012).

10.Sainarayanan G., Nagarajan R., Yaacob S., Fuzzy image processing scheme for autonomous navigation of human blind. Appl. Softw. Comput.257-264(2007).

12.APPENDIX

4.4.CODE SNIPPETS

4.4.1.OBJECT DETECTION

```
import inotify.adapters

import logging

from thread import start_new_thread



import numpy as np

import argparse

import cv2




def object_detect(img_name):

model = "/home/pi/Documents/project/object-detection-deep-learning/MobileNetSSD_deploy.caffemodel"

proto ="/home/pi/Documents/project/object-detection-deep-learning/MobileNetSSD_deploy.prototxt.txt"

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(proto,model)


# load the input image and construct an input blob for the image
```

```
# by resizing to a fixed 300x300 pixels and then normalizing it
# (note: normalization is done via the authors of the MobileNet SSD
# implementation)
image = cv2.imread("/home/pi/Documents/project/object-detection-deep-learning/images/"+img_name)
(h, w) = image.shape[:2]
blob = cv2.dnn.blobFromImage(cv2.resize(image, (300, 300)), 0.007843, (300, 300), 127.5)

# pass the blob through the network and obtain the detections and
# predictions
print("[INFO] computing object detections...")
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 > 0.2:
# 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")

# display the prediction
label = "{}: {:.2f}%".format(CLASSES[idx], confidence * 100)
print("[INFO] {}".format(label))
cv2.rectangle(image, (startX, startY), (endX, endY),
COLORS[idx], 2)
y = startY - 15 if startY - 15 > 15 else startY + 15
cv2.putText(image, label, (startX, y),
cv2.FONT_HERSHEY_SIMPLEX, 0.5, COLORS[idx], 2)

# show the output image
cv2.imshow("Output", image)
cv2.waitKey(0)
return new
i = inotify.adapters.Inotify()
i.add_watch('/home/pi/Documents/project/object-detection-deep-learning/images/')

for event in i.event_gen():
if event is not None:
(header, type_names, watch_path, filename) = event
if(event[0].mask == 8  ):
print(event[0].mask)
f = open("hello.logger","a")
f.write("hello\n")
f.close()
start_new_thread(object_detect , (event[3],) )
```