Final Report: Project Proposal

Outdoor street navigation for people with limited vision

A Report Presented to
The Department of Electrical & Computer Engineering
Concordia University

In Partial Fulfillment of the Requirements of ELEC/COEN 490

by Group 13

Ossama Sabsoob	**ID#: 26395139**
Samih Makhams	**ID#: 26583067**
Steven Peters	**ID#: 26590519**
Sun He Li	**ID#: 26624847**

Project Supervisor
Dr. Yousef R. Shayan

Concordia University
January 2017

Abstract

In recent decades, many technological advancements have made it possible to find solution for complex issues. Finding a solution for people with limited vision is a breakthrough. Many people cannot afford buying a walking dog which is considered to be expensive in the market. Also, extensive training is needed before using the white cane. Our product is simple, affordable, and affective. It alerts of user of obstacles, notifies the user of the traffic light color and provides navigation though voice output. This replaces walking sticks which do not help people with limited vision to cross the street or to see whether the traffic light is green or red. This product has big future in the market as it can be extended later to include more scenarios and cases. It's impressive simple and inexpensive product that can resolve such a major issue.

Contents

List of Tables

List of Figures

1. Introduction

1.1 Objectives

This project will help a person with limited vision navigate his way through the streets of a city. Since the person is unable to see clearly, he should be alerted about the route to follow, when obstacles are present and when to safely cross a street. The user should be able to clearly understand the instructions given to him and should be guided correctly and efficiently to his desired destination.

1.2 Outlines of the project

This project helps a blind person navigate the streets of a city. It needs to detect obstacles, steps, traffic light colors, and navigates the user to his desired destination. It therefore has an obstacle detector that can detect pedestrians, dogs, pigeons, light poles, the edge of sidewalks, garbage bins, bus shelters. It also has a traffic light detector that is able to identify green and red lights. Lastly, it has a navigator that tells the user the directions to follow to get to his desired destination. These three systems are placed on a medical walker. A battery powers everything up (except for the phone).

1.3 function of the system from users' point of view

The user starts with choosing his destination using a customized app which will provide turn by turn voice navigation from the system when he needs to go to a place. While he's walking on the street, the walker with obstacle detector will avoid him hitting an obstacle or people with different types of signals. During his path, if the user reaches an intersection, he will be asked to click the shutter button to take a picture with pre-installed phone on the walker as the system analyses and tells the user the traffic light information, and at the end he will successfully reach his destination.

1.4 Potential applications

Our product is originally design to navigate blind people outdoors. Even it's only a prototype for now, we can imagine it has huge potentials. There will be extensive applications in a daily life. Since it can detect and signal user to avoid an obstacle, which could be used for indoors too. Walkers with baskets make it convenient for blind people shopping in a grocery store. There is also a potential application of robotic delivery. Stores put the package in walker and it can deliver the package to a customer's place. With our obstacle detection and traffic light detection, traffic makes no concerns.

1.5 Gantt chart (Task breakdown)

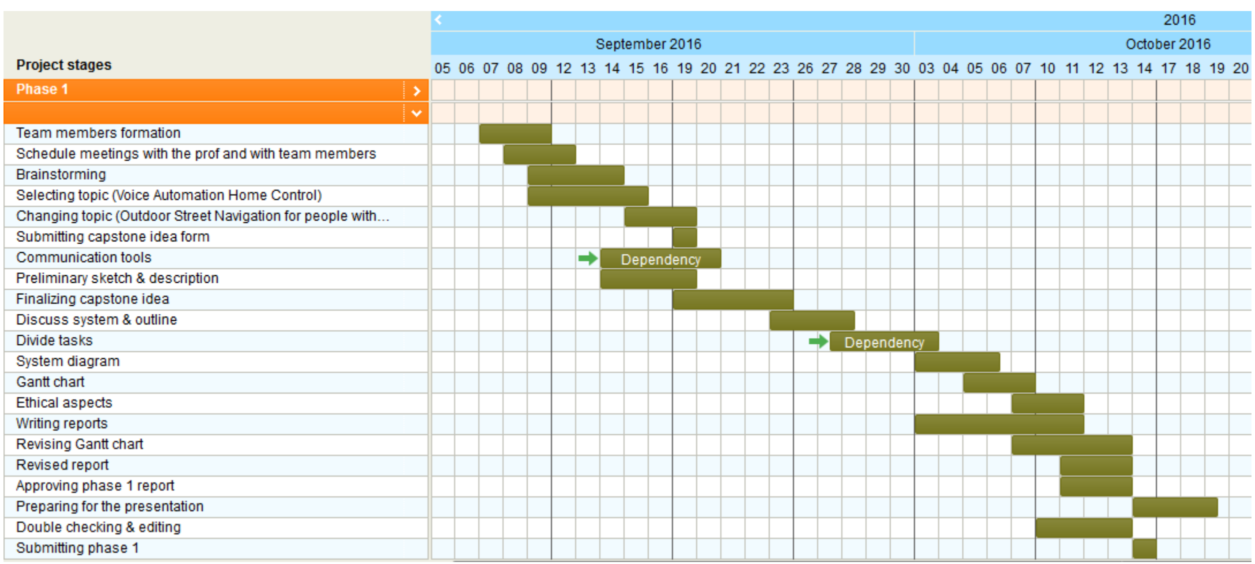


Figure 1 Gantt chart for Phase 1

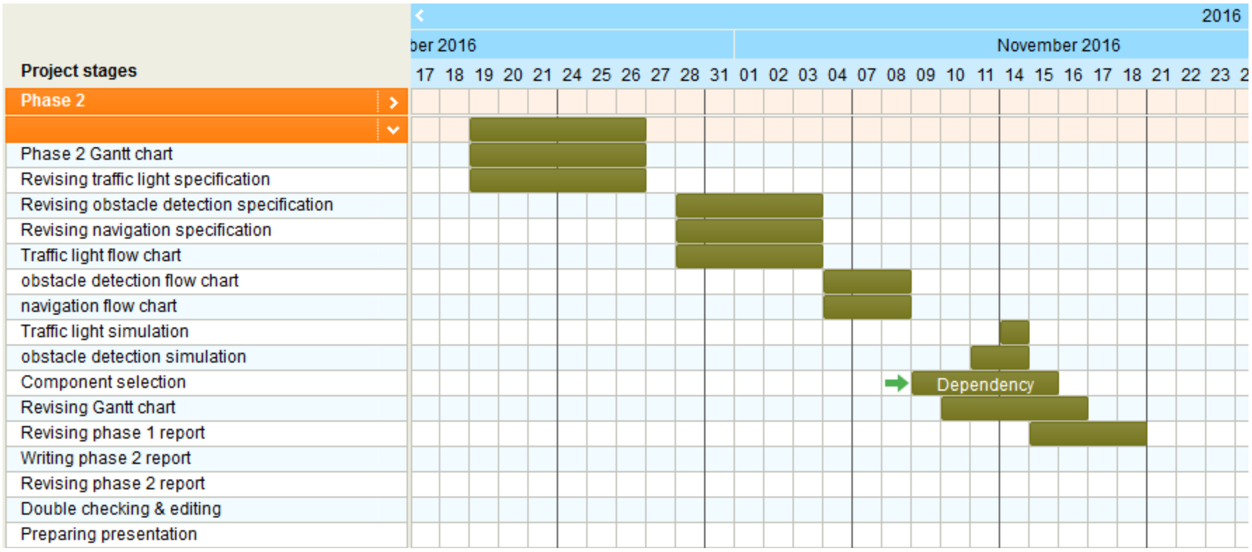


Figure 2 Gantt chart for Phase 2

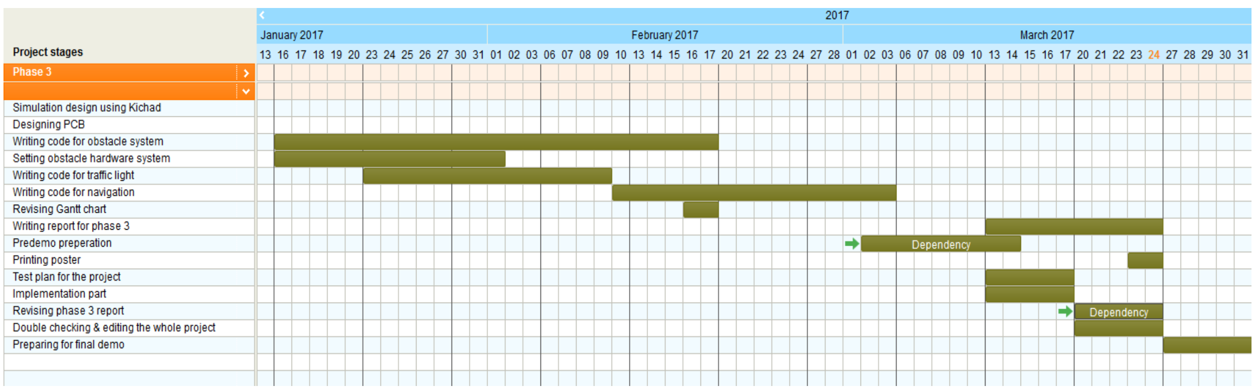


Figure 3 Gantt chart for Phase 3

For phase 1 of the project, we teamed up and scheduled meetings with the professor and with the team members. Then, we brainstormed the ideas and selected the topic which was (Voice Automation Home Control). Next, we changed the topic to (outdoor street navigation for people with limited vision). After that, we set our communication tools and divided the main tasks and then did Gantt chart and we wrote about ethical aspects and revised our report to submit it on the 18th of October. We set two dependencies for phase 1, one at preliminary sketch & description, and the other one at dividing tasks. We set the dependency on dividing tasks.

For phase 2 of the project, we started this phase by revising the traffic light specifications that were done in phase 1 part of the project. This took place on the 19th of October, 2016. Then we revised the specifications of the obstacle system and then navigation system. After that, we revised the flow diagrams and the functional diagrams of the traffic light, obstacle, and navigation systems. Next, we did simulation of the traffic light and obstacle detection system. After we finished the simulation, we were able to select our components properly. Then, we started writing the report for phase 2 and started to prepare the presentation for this phase. We also, included Gantt chart and managed to edit & double checked the report. For this phase, we set dependency at the component selection as this part cannot be done except when everything else before it is done.

We started phase 3 from 16, January, 2017. We started this phase by simulating the design using Kichad, then we started making the PCB design. After that, we started writing the codes for all the sub systems including traffic light, obstacle, and navigation systems' codes. We revised Gantt chart then we started preparing for the pre demo. After that, we started preparing the poster and writing about the test plan and the implementation parts of each sub system. Finally, we started writing the whole report and double checked it and edited the whole project. And we prepared for the final demo after we submitted the report on the 24th, March, 2017. In this phase, we set the dependences at revising phase 3 report and another one at pre demo presentation because we cannot do pre demo presentation except when the previous parts are already done.

2. Descriptions

2.1 - Structure

Outdoor Street Navigation for people with limited-vision system is built and integrated in a four-wheel walker. It consists of three subsystems: Navigation, obstacle detection and traffic light detection. The navigation and traffic light detection systems are developed based on an android app. A cell place on the walker with our app installed is performing navigation and traffic light detection. The obstacle detection system consists of a microcontroller, two ultrasound sonar, one IR sensor, two vibration motors and a buzzer integrated with PCB(expected) in the basket See specific model of components in each subsystem. Bellow in Figure 4 is a photo of our product.

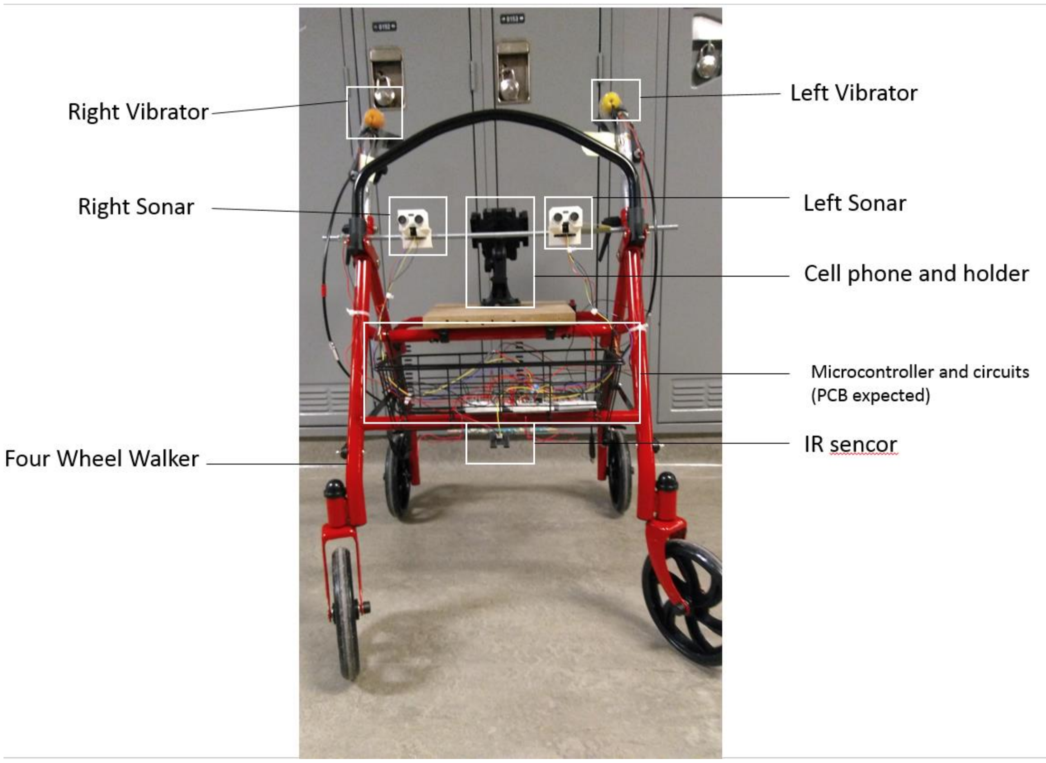


Figure 4 Walker Structure

2.1.1 Navigation and Traffic Light Detection

The navigation and traffic light detection system are develop base on an android application. Any android device with 4.1+system could run this application. Figure is the icon of the app. In our project, we place a Nexus 6p within a holder placed on the walker as part of the traffic light detection system.

Figure 5 Outdoor Navigation

User interface is deign as Figure 6. This interface is customize to ease the use by people with limited vision. Icon size is large and easy to locate. User could perform single click a button to hear the destination name and long press to start navigation for the destination. Traffic light system is running in the background since the launch of the application. It keeps monitoring the current location and calls camera to perform traffic light analysis when user reaches the marked intersections in the database. See Figure 7 for screen of traffic light detection.

Figure 6 User Interface

Figure 7 Green Light Detected

2.1.2 Obstacle Detection System

The obstacle detection system exist to notify the visually impaired user of the presence of any obstacles or steps/drop-offs that the he may collide with. The component placements can be view in Figure 8. Two ultrasonic transducers are placed at the front of the walker at a height of 0.68 meters from the ground. The two sonars are separated by a distance of 0.33 meters.The IR sensor controls the buzzer, which isn’t visible in Figure 8 as it is placed directly on the circuit board.

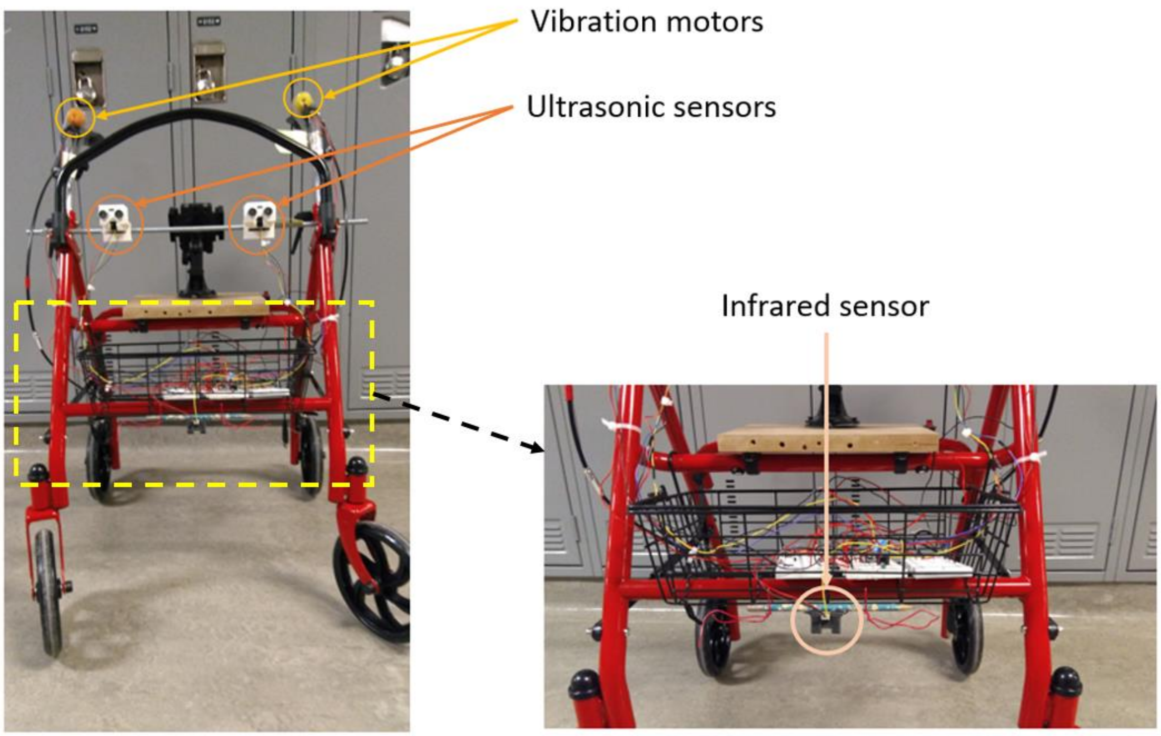


Figure 8 View of the walker and the components

2.2 Specifications

The system is designed to work under the environmental requirements mentioned below. It is also able to detect human sized obstacles at a distance of 1 m. The maximum obstacle detection speed is 10 km/h. Also, the traffic light detector identifies traffic light color at a distance of 4-10 m from intersections. The navigator pinpoints the user to within 5-10 m of his desired destination. Below are the detailed specifications about the system and its different components.

2.2.1 Environmental requirements

The device is to be operated outdoors and in urban cities only. It will operate on concrete and asphalt pavements. The device must be used outdoors only. A four-wheeled walker will be used by the visually impaired user.

General Features	**Description and/or specifications**
Weather condition	No heavy snow and no rain to avoid damage to the electronic components
Temperature	Min: -20 C Max: 40 C These temperatures are to preserve battery life and avoid damage to the battery
Time of day	Excluding 12:00- 14:00
Road condition	No snow covering the sidewalk. Only asphalt and concrete.
Location	In city streets only. No countryside. Also for outdoors only.
Humidity	Outdoor relative humidity ranges 0-95%

Table 1 Environmental Requirements

2.2.2 Obstacle detector

The device needs to detect obstacles at a distance of 1.50 meters in front of the user. Any obstacles that can obstruct the user needs to be detected. The device will notify the user of obstacles using vibrations.

The device needs to detect obstacles found on sidewalks

Non-moving obstacles:

poles, garbage cans, parked vehicles, buildings and trees

Moving obstacles:

vehicles, pedestrians, bicycles, dogs, cats

Requirement	Operating (testing) conditions	Values		Units
		Min	Max	
Detection range	A 0.6x1.7 m obstacle size	0.03	1.00	meters
Detection width	At a 1.5 meter range	0.61	0.685	meters
Obstacle size	At a 1.5 meter range	0.30x0.30	-	meters

Obstacle speed			10.00	Km/h
Power		5V-4 Ah	5V-4 Ah	

Table 2 Obstacle detector

2.2.3 Navigation specifications

The product is design to navigate person to a desired location with walking distance. It has an accuracy within 5-10 meters from target location and lasts maximum 30 mins long consider power consumption.

General Features	**Description and/or specifications**
Operating System	Android 5.0 and above
GPS	Fine and coarse location access
Internet	4G LTE connection available
Accuracy	Within 5-10 m from target location
Battery	5 V 1500 mAh lithium-ion battery

Table 3 Navigation Specifications

2.2.4 Traffic Light specifications

The traffic light detector determines the color of the traffic and returns an appropriate voice output as to whether it's green or red. The success rate of identifying green and red traffic lights is 85%. The rest of the time an output of "no light identified" is returned as a voice output.

General Features	**Description and/or specifications**
Distance from traffic light	4-10 m
Vertical angle of camera	85 degrees
Traffic Light tunnel visor shape	Standard circular visors
Operating time	Anytime except 12:00-14:00

Table 4 Traffic Light specifications

2.3 Approach to achieve the objective

In order to achieve the objectives of our project, we have the following design as approach. Navigation system and traffic light detection system will be android based application. User will get turn by turn voice navigation using the app. Voice output are used since the targeted customers are with limited vision. Obstacle Detection system will be able to detect certain size of obstacle which could exist on the sidewalk and also whether there is a step/hole. The whole system will be integrated in a four-wheel walker, the cell phone will be place on the walker too using a holder with adjustable angle to catch traffic light. Ultrasound sensor and IR sensor are used in obstacle detection system. The figure bellow is the functional diagram of our product.

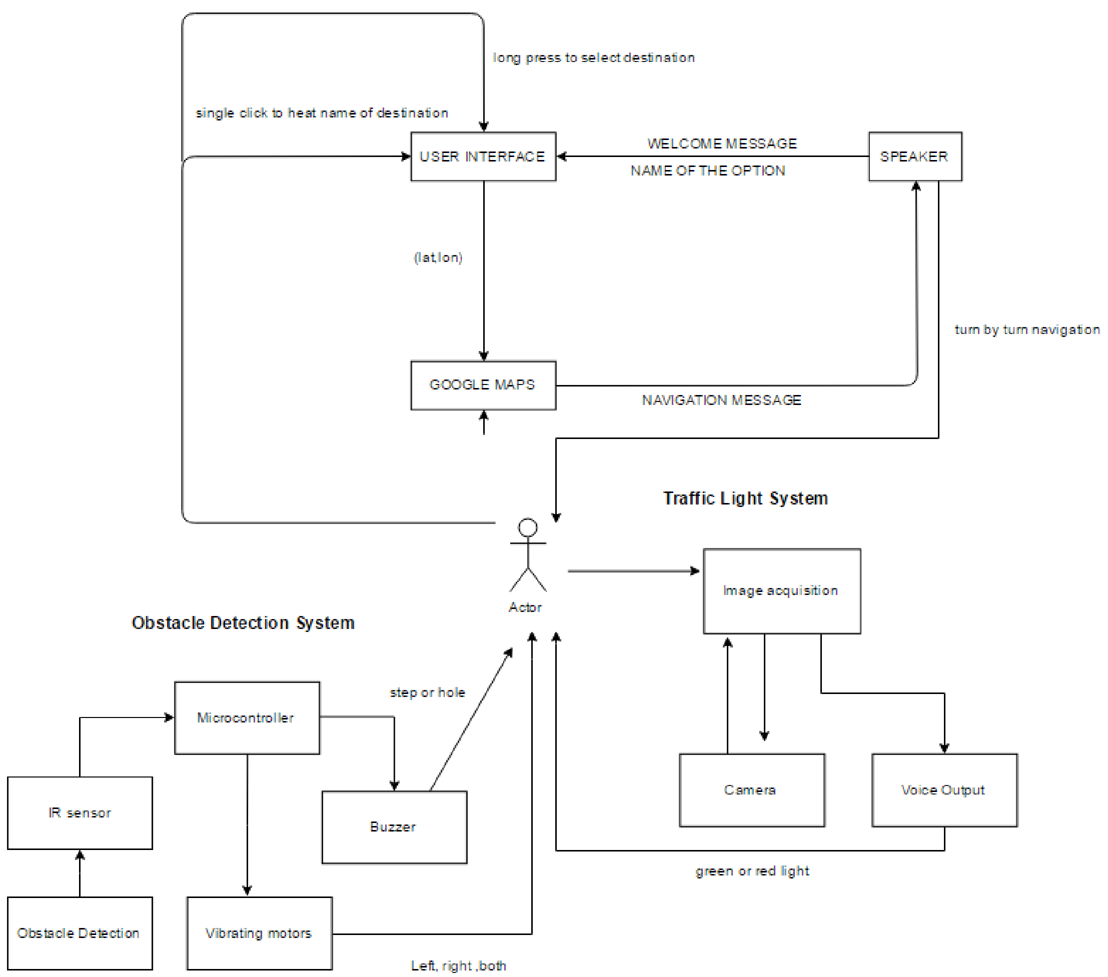


Figure 9 Approach to achieve the objective

2.3.1 Navigation

Our navigation is developed based on android environment. A cell phone is used to perform the navigation. The user will choose from the pre-defined destination. The device will then navigate the user to the destination using voice commands.

Considering of practical reason, instead of letting a user with limited vision to input an address which is barely impossible, we pre-define several destinations in our app and make them as buttons. According to researches about real life situations of people with limited-vision, they only visit certain locations for most of his/her time.

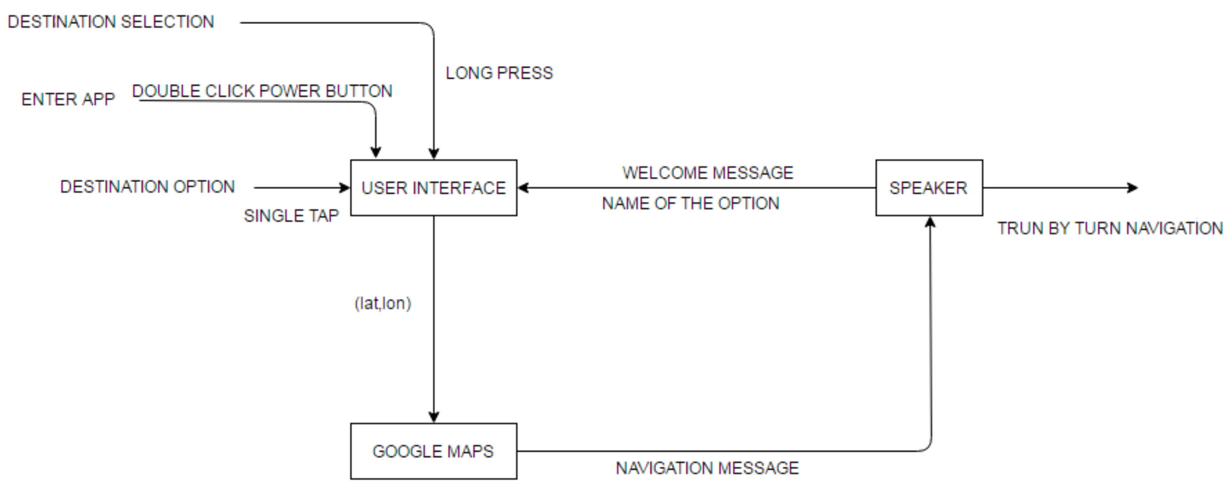


Figure 10 Navigation functional diagram

By the first time enter the app, a piece of voice message will start playing such as "Welcome to Navigation for Limited-Vision User! You have six options to choose. They are *Home*, *Hospital*, *Mall*, *Bank*, *Community Service and Metro.* "

By single tapping on the buttons, it will not enter the navigation right away but notify the user the choice by voice. For example, when a user tap on first icon on the screen, it will gives the user a voice message say "Option 1 of 6: Home". Then user could apply a long press on the button to enter navigation and start going home. Below is the use case diagram for navigation.

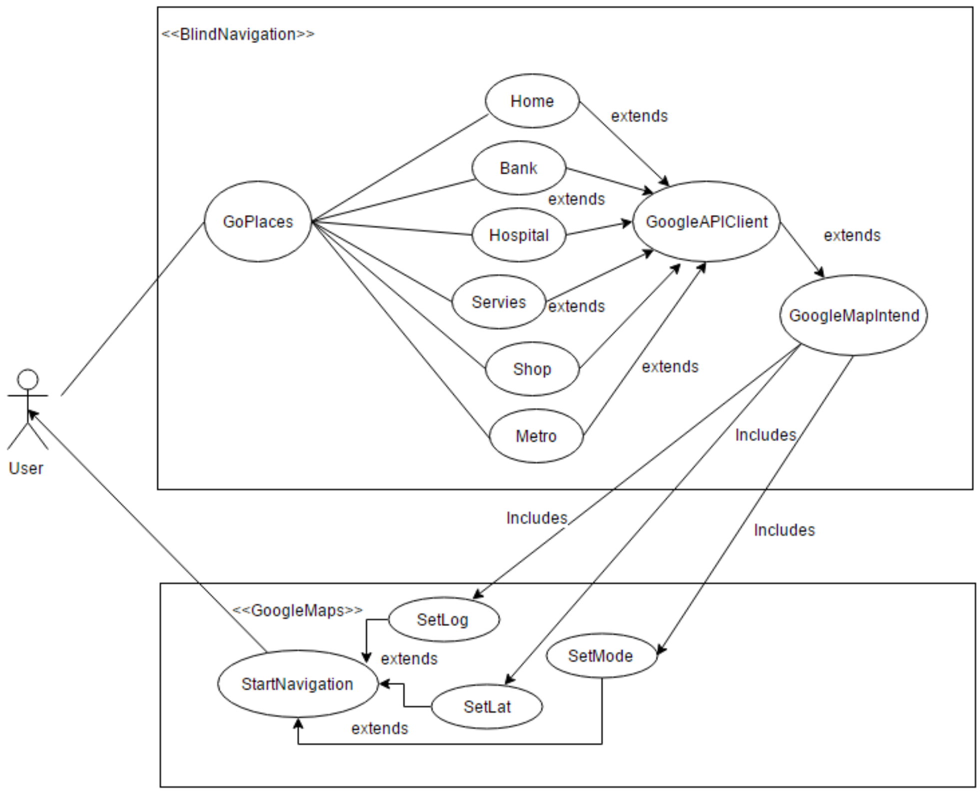


Figure 11 Use Case Diagram for navigation

The device will notify the user when he arrives to his destination and tell the user whether it's on his left of right.

2.3.2 Traffic Light Detection

The traffic light detector has to analyze and identify the color of the traffic light (green or red). Then a voice output lets the user know the color that the algorithm identified. The program will work in the steps shown in the flowchart below.

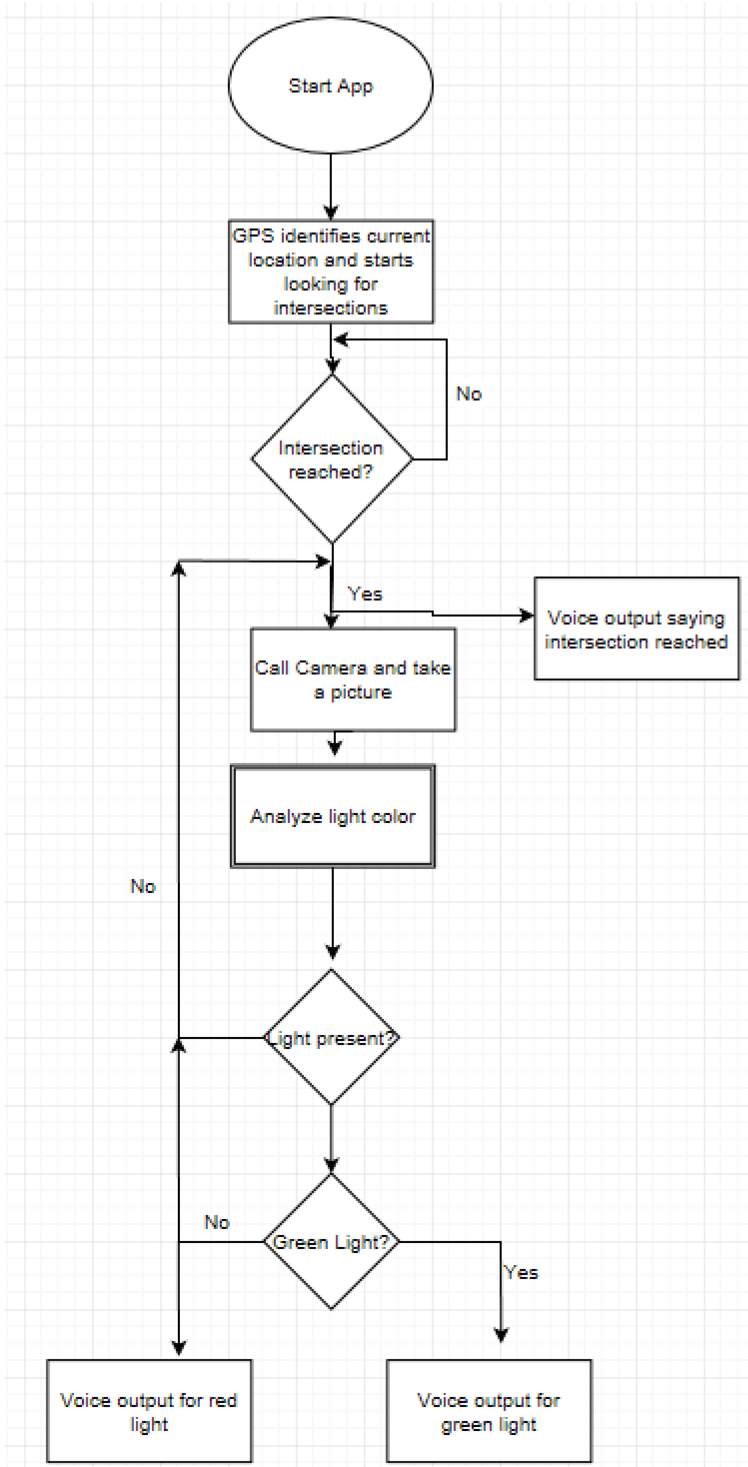


Figure 12 Flowchart for the traffic light detector

First the app is started. This app is run in a background service initialized by the Navigation app. This will let both apps be merged into one phone. Once the traffic light app is initialized, it will automatically start looking for the current GPS location. Once the current GPS location is identified, the algorithm will look through preset intersection markers. If the GPS finds a marker that is close enough, the algorithm will trigger a voice output and open the camera. The user positions himself towards the traffic light intersection that will be crossed. A picture will be manually taken by a Bluetooth shutter. The picture is analyzed right away. Three possible outcomes exist; green light, red light, and no light. A voice output will notify the user for each case. If no light or red light are identified, the camera will be recalled and the user will take another picture to be reanalyzed. If green light is identified, the camera will be closed and the program goes back to the GPS. No new intersections will be verified till the user moves away from the current intersection.

2.3.3 Obstacle Detection

To detect any potential obstacle could be on the sidewalks, we need to test and specify our range of detection.

The detection zone has a range of 1.00 meter and a width of 0.59 meters. A range of 1.00 meter is chosen based on the average waking speed of the user and his reaction time. The width if the detection zone needs to be at least at wide as the walker. The detection zone is divided into 3 regions: left, right and center. In order for the visually impaired user to determine in which region the detected obstacle is in, two ultrasonic sensors will be used. The sonar sensors will be separated by a distance that will cover the width of the walker and will be able to locate an obstacle within one of the three regions. The right sonar covers the right region and the left sonar will covers the left region. If an obstacle is located in the center region, the both sonars will detect it. The calculations are shown below.

1. $*a = Range\ \times \tan(beam\ angle/2) = 1.00 \times \tan(15°/2) = 0.132\, meters$

2. $Separation = Width - 2a = 0.59 - 2 \times 0.132 = 0.33\, meters$

**a is defined as half of the detection width of one sonar at a distance of 1.00 meter. Refer to Figure 4.*

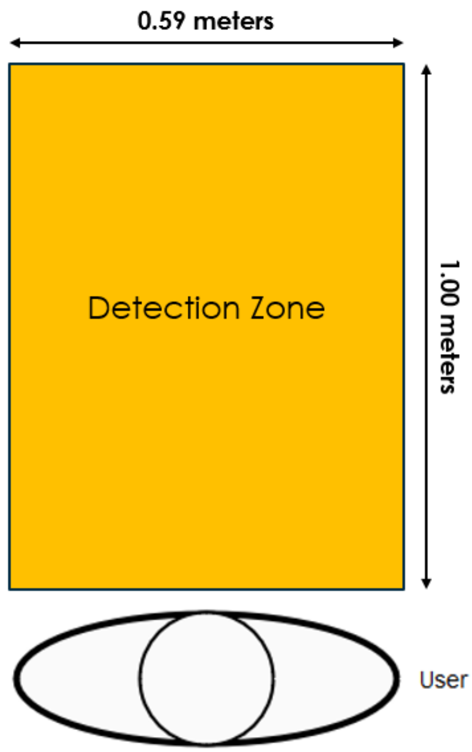


Figure 13

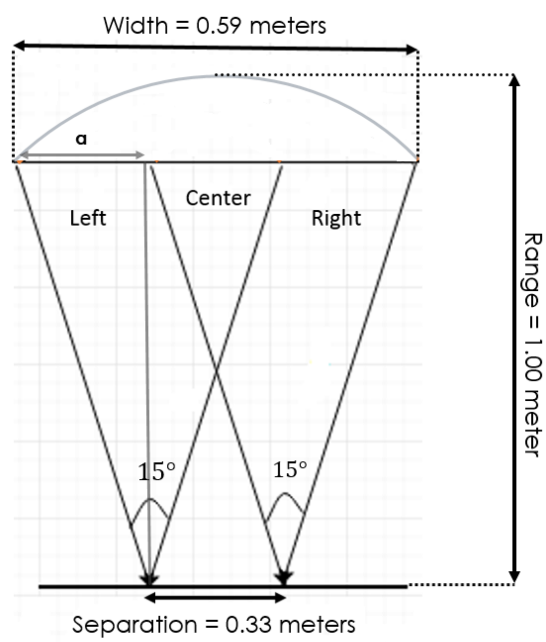


Figure 14

The infrared sensor emits infrared light and measures the intensity of the reflected light. This allows the sensor determine the distance to the detected object. When the infrared sensor detects a step or a drop-off, the buzzer will be activated.

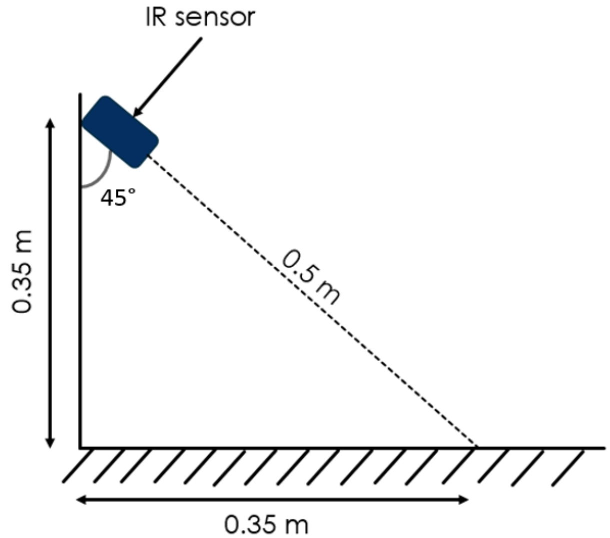


Figure 15

Once the detection range been decided, the follow functional diagram become very clear since we know we we need to approach.

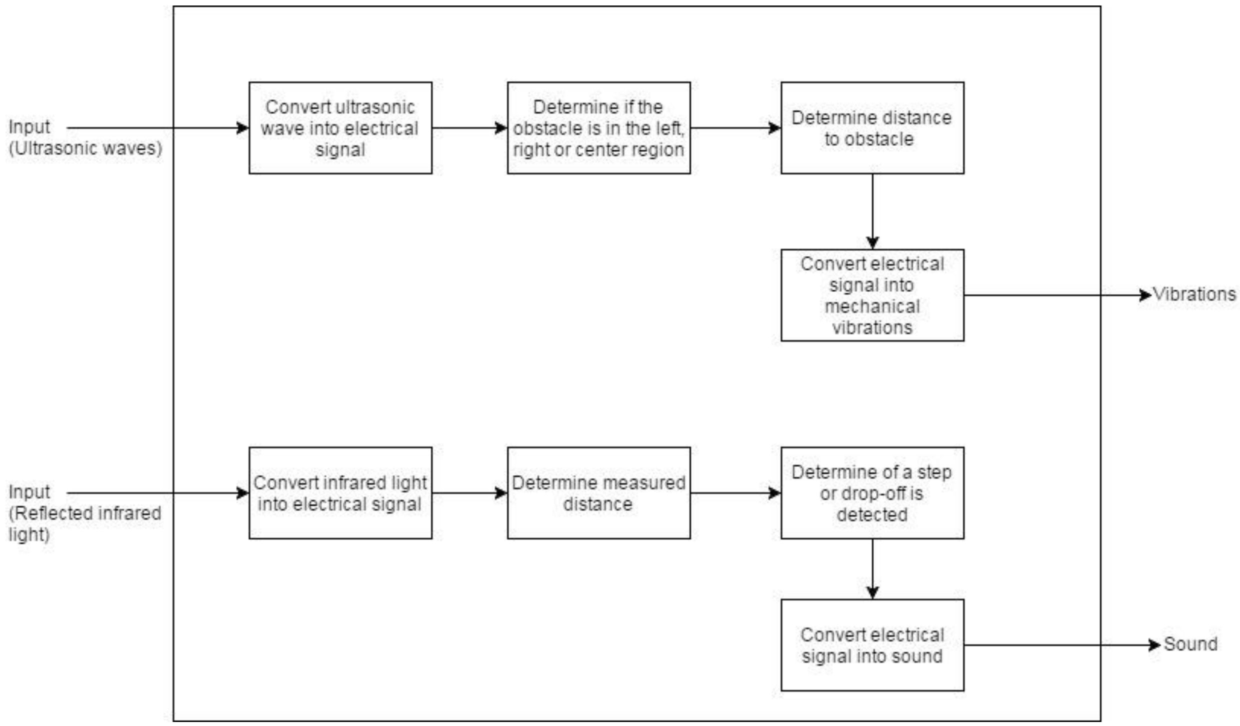


Figure 16 Obstacle detection functional diagram

The two ultrasonic transducers, or sonars, are placed at the front of the walker which produce ultrasonic sound waves. These sound waves bounce of the surface of any object placed on front of the sonar. The sonar then receives these reflected sound waves and can measure the distance to the obstacle by measuring the time it takes the sound waves to be transmitted and received by the sonars. Therefore, ultrasonic sensors can not only detect the presence of an object, but also how far the object is from the sensor. Two sonar sensors are used to identify the presence of obstacles on the right, left and center regions of the detection zone as shown in Figure 16. The left sonar detects obstacles on the left, and the right sonar detects the obstacles on the right. If both sonars detect an obstacle, this means the obstacle is placed in the center region. In addition, the sonars measure the distance to the obstacles. The strength of the vibration motors are proportional to the distance of the detected obstacle.

Vibro-tactile feedback is used to notify the user of the presence of obstacles. Each sonar controls a vibration motor placed on the handles of the walker. When the right or left vibration motors are activated, the user will know that an obstacle is found either on his left or right, respectively. When both vibration motors are activated, this means that an object is found in the center of the detection region. Also, the vibration motors will vibrate with an intensity proportional to the detected distance of an obstacle.

The infrared sensor emits infrared light and measures the intensity of the reflected light. This allows the sensor determine the distance to the detected object. When the infrared sensor detects a step or a drop-off, the buzzer will be activated.

3. Simulation

3.1 Traffic light simulation

The traffic light detector was simulated with an algorithm written in C++ on Visual Studio. The functions from the computer vision library “OpenCV” were used. This free library package allows the manipulation of pictures and pixel matrices. Traffic light pictures were taken at different times during a period of 12 hours with a HTC One V phone camera. The algorithm identifies the proper RGB and grayscale code for green traffic light in the loaded picture and then outputs the picture with the green light area darkened. Also, the algorithm looks only at the top of the picture to save computing time, as the traffic light should be in the top only. If a traffic light is lower down, then it is assumed that it’s from another street intersection, and shouldn`t be taken into consideration. Individual RGB pixel values are accessed and the algorithm determines whether the RGB and grayscale code corresponds to a predefined pattern that is identified as green traffic light. If so, then the identified pixel is replaced by a black pixel to show that the light was correctly identified. Then a message is outputted saying “Green Light Detected”. Here a few pictures with their results:

<u>Picture 1</u>

Figure 17 Traffic Light Simulation - day

In fig 17, the green light here is identified clearly and the center is entirely darkened. It is to be noted that apparent RGB values tend to change according to the position of the sun.

<u>Picture 2</u>

Figure 18 Traffic Light Simulation - night

In fig 18, the green light at night is clearly identified and is more visible than during the day. Much more pixels were identified as being green traffic light at night than during the day.

If the algorithm cannot discern a green traffic light from the picture, it will output a message saying "No Green Light Detected". This is the case when a red light is present. Below is an example of this.

<u>Picture 3</u>

Figure 19 Traffic Light Simulation

Now a few test cases were run into the algorithm. Here they are.

<u>Picture 4</u> **<u>Picture 5</u>** **<u>Picture 6</u>**

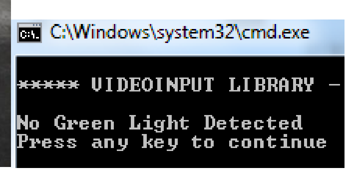


Figure 20 Test case 1 *Figure 21 Test case 2* *Figure 22 Test case 3*

In fig 20, the traffic light is too low in the picture, and so is not identified. The algorithm looks at only the first 400 rows of pixel in the picture. Everything beyond that is deemed too far. **In fig 21**, the green has been edited and replaced by a white circle. The green light is therefore not found.
In fig 22, the green light has been replaced by a white circle and also images of green leaves have been added to the picture. The green leaves do not have the same RGB code as green light and therefore no green light is identified.

4. Implementation

4.1 Implementation of the system

The project was implemented on a walker. This specific walker is commonly used by elderly people to move around. It features two handles with lever breaks. It has a basket that is 40 cm long, 30 cm wide and 20 cm high. We chose these dimensions for the basket because we will store the battery and PCB in the basket. The battery is 13 cm high and its base is 10x13 cm. The PCB is 20x20 cm. The dimensions of the basket therefore will only leave 7 cm of space that will be filled with foam. This will prevent sudden movements which could damage the obstacle system. The walker itself is 1.2 m high. This height is at a comfortable position for the user, as his arms will be parallel to the ground, and not angled. The handles are covered by a layer of solid rubber, which provides a firm grip. The distance from the left to right wheels is 70 cm. The average width of a sidewalk is 3.6 m, and so the walker would take up 1/5 of the sidewalk. The color chosen for the walker is red because it stands out better at night and in the snow. The basket is in the bottom front of the walker, at a height of 60 cm from the ground. This allows good clearance from the ground. The wheels can spin 360 degrees on their axle, and the bearings are smooth. This allows quick and effortless rotation in case of emergency. The weight of the walker is light, as it weighs 7 kilograms.

The frame material is hollow structural steel, which is a lightweight yet very durable construction material. The sidewalk will cause a lot of vibrational stress on the structural frame of the walker, and the chosen frame type has high strength rating. The PCB and battery are placed in the basket to be out of sight and also to not get in the way of the user. The basket also provides protection to the battery and PCB and will prevent damage and wear. The system has ultrasonic sensors, an IR sensor, and an Android phone. The ultrasonic sensors are used to detect obstacles down to a height of about 40 cm. Ultrasonic sensing is great at detecting obstacles at short ranges (within 3 meters). These sensors are very cheap ($6 each) and provide accurate obstacle detection. The sound waves they emit travel at the speed of sound, which is about 340 m/s. Therefore, if an obstacle is 1 m away, the feedback will be obtained in 5.88 ms. The time is nearly instant and the user feels no delay. The IR sensor is used to detect obstacles on the ground up to 20 cm high. This includes the edge of the sidewalk, pigeons, holes, steps. IR sensing is well suited to identify objects at very close proximity (60 cm away). Its beam is very narrow (3-4 degrees of spreading at 60 cm), whereas the ultrasonic sensor is much wider (20 degrees of spreading at 60 cm). Also, the light won’t bounce off the surface of the sidewalk (sound waves do however). The traffic light is analyzed by taking a picture and analyzing it through an algorithm. The detector is an Android phone. Android phones run on the Android operating system, which provides a great programming platform. The phone itself has many useful features which are used in the system, such as the built-in camera, GPS and Internet access. Also, the same phone has the navigation application, which runs in parallel with the traffic light detector. The ultrasonic sensors are placed on a rod on both sides of the walker. The rod is made of wood and has a screw on each side. Each screw has a bolt that is used to tighten the rod on the walker. In case of adjustment, the bolts can be unscrewed and the rod rotated in the desired direction (counter clockwise will orient the sensors downwards, clockwise will orient the sensors upwards). The casings for the ultrasonic sensors were 3D printed since no casing on the market fitted the sensors. The IR sensor is placed on a wooden rod that is attached to the basket, and the angle can be changed by rotating the rod either CCW (down) or clockwise (up). The phone is placed on a plastic phone holder that was purchased from Dollarama. It provides two degrees of freedom. It moves up and down with an angle of 260 degrees, left and right with an angle of 280 degrees. Tightening screws will ensure the holder will stay in that position indefinitely. This type of mounting provides for quick adjustment and testing of the traffic light detection. The plastic it is made of is very durable.

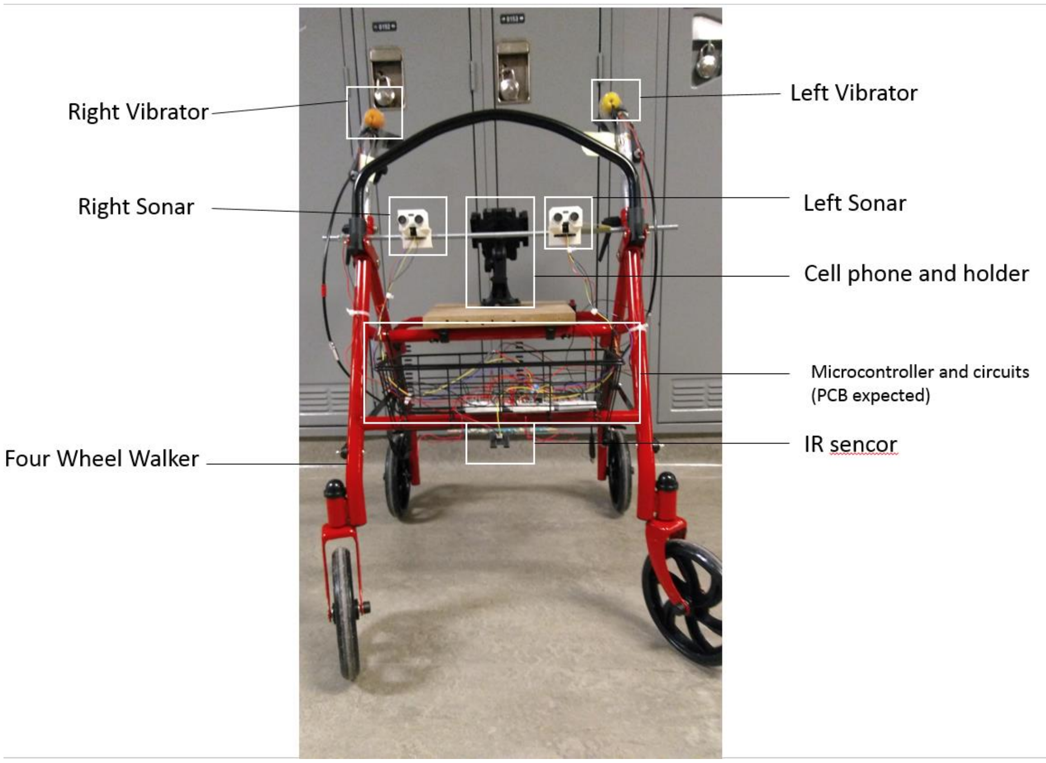


Figure 23 Walker Structure

4.2 Navigation Implementation

Based on the design of navigation system, we develop an android application. The follow diagrams shows the communications between the classes from software respective.

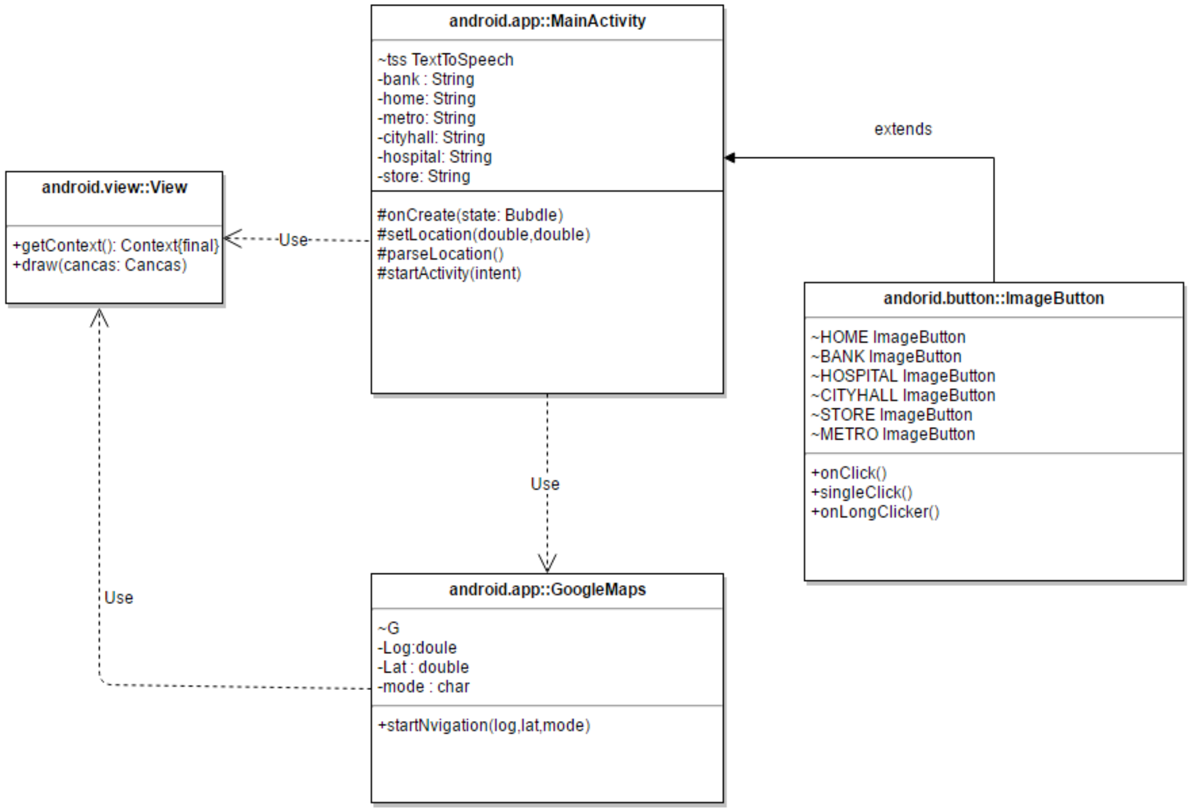


Figure 24 Use Case Diagrams for navigation

The MainActivity extends android view class and display the user interface. The Icons in the User Interface in figure5.4 are Imagebuttons.

Figure 25 User Interface

```
ImageButton Home_Button=(ImageButton)findViewById(R.id.button_home);
```

This is how the buttons are mapped with different locations. Each button has its own id, when the button is clicked, system check its id and give the permission to run for this id.

When there is single click, it checks the button id and play the name of the destination. Text to speech technology is used here. It converts a text message to Voice output and play it when a click initiated.

```
Store_Button.setOnClickListener(new View.OnClickListener() {

    @Override
    public void onClick(View v) {
```

```
if (v.getId() == R.id.button_store) {
        //mpButtonClickHome.start();
        tts.speak(store, TextToSpeech.QUEUE_FLUSH, null);
} } });
```

Whenever there's a long press on the button, it checks the button id and starts parse the coordinates and pass them as well as mode of navigation to the GoogleMaps, and launch the application with startActivity() method.

```
if (view.getId()==R.id.button_home){
    //mpButtonClickHome.start();
   // tts.speak(home,TextToSpeech.QUEUE_FLUSH,null);
    Uri gmmIntentUri = Uri.parse("google.navigation:q=45.497284, -73.578917&mode=w");
    Intent mapIntent = new Intent(Intent.ACTION_VIEW, gmmIntentUri);
    mapIntent.setPackage("com.google.android.apps.maps");
    startActivity(mapIntent);}
```

Once GoogleMaps receive the parsed longitude and latitude, it calculates the shortest route according to the traffic and start voice navigation right away with default mode as walking. See figure below the implementation.

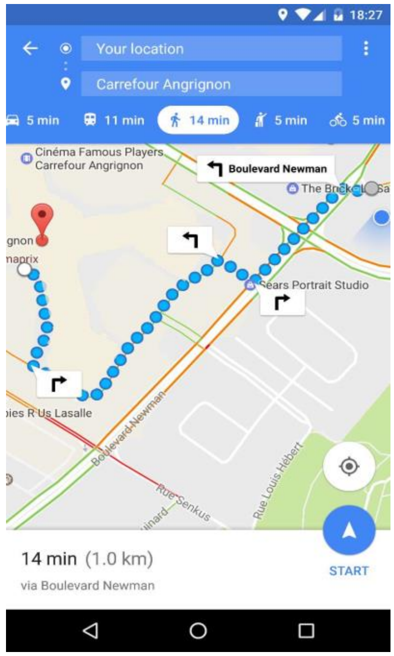


Figure 26 Navigation Page

4.3 Implementation of the traffic light detector

The traffic light detector was implemented on an Android phone that runs Android Ice Cream Sandwich 4.0 OS. The entire software was designed from scratch on the Android Studio platform. The Android programming language uses both Java and Android XML. The program for the traffic light detector consists of four classes, these are: MainActivity, CameraMaster, CameraPreview and ZeroIt. MainActivity contains the GPS algorithm that finds the distance from the current location to designated markers that represent traffic light intersections. The markers have both latitude and longitude values. An entire city could be mapped out in this way (code below).

```
latitude[0]= 45.49669;
latitude[1]= 45.495710;
latitude[2]= 45.497416;
longitude[0]= -73.578768;
longitude[1]= -73.579359;
longitude[2]= -73.578143;
```

If the distance is less than 20 m, a voice output will warn that an intersection will be reached soon, and the camera is called (code below).

```
if (distance < 20) {
    showStatus.setText("Intersection found");
    control2 = 1;
    intent.putExtra("Begin",1);
    startActivity(intent);
    break;
}
```

CameraMaster and CameraPreview classes initialize the camera in this app. They have been completely designed and implemented from scratch. The built-in camera app in the phone is not used. The reason is that using the built-in camera app will require the user to manually press on many buttons to save the picture and go back to the app. However, defining and implementing a camera view in the app allows the automation of the system (code below).

```
camera = checkDeviceCamera();
mImageSurfaceView= new CameraPreview(CameraMaster.this, camera);
cameraPreviewLayout.addView(mImageSurfaceView);
```

Once the camera is called, the user will have to manually take a picture with a Bluetooth shutter. It is preferred that the user takes the picture himself since it may take him time to adjust his position and face the correct traffic light. If the algorithm were to take a picture automatically, the wrong traffic light may be photographed. Also, due to GPS accuracy issues, the person may be more than 20 m away from the traffic light even though the algorithm warns the person that an intersection has been reached. For these reasons, the user will take a picture manually. All the other operations, however, are handled automatically by the program.

Once a picture is taken, it is saved in the Bitmap format in the temporary memory of the app. This means that every new picture will overwrite the preceding picture, and once the app is closed, the last picture taken will be completely destroyed. The code snippet below shows this.

```
bitmap = BitmapFactory.decodeByteArray(data, 0, data.length);
```

The picture is analyzed by a function that was designed for finding green and red traffic lights in different shapes (arrow, circle, square, and triangle). Yellow light was omitted because by the time the algorithm identifies yellow light, the traffic light would have turned red, and the person wouldn't have time to cross the street. Instead, a yellow traffic light will be the same as no light present. To analyze a green or red traffic light, the function looks at the pixels of the top of the picture only (the first 200 pixels). Anything below that line will be ignored. Firstly, this increases computing time massively, as only 200 pixels out of a height of 2500 pixels are analyzed (code below).

```
for (int i = 0; i < 200; i++)
```

The app tends to crash after a few pictures if the whole pixel height is analyzed. This is due to internal temporary memory stacks completely saturating and heating up. Also, by only looking at the top of the image other lower light sources that may cause interference are ignored. These interfering light sources are car break lights and store lights. The rgb code and brightness of each pixel is analyzed and a counter adds up if the pixel corresponds to either the green or red of a traffic light (two different counters). If the counter for green or red in a picture is over a threshold value (20 in the day and 1000 after sunset), the function outputs a voice message and returns true (code below).

```
if (red<0.1&&difference>0&&difference<0.1) {
    return 1; //green color
}
else if (red>0.7&&green<0.165&&blue<0.36&&difference<0) {
    return 2; //red color
```

```
}
else return 0; //Not green nor red

if (checkRGB(red, green, blue) == 1 && checkGrayscale(red, green, blue) == 1) {
    greenCounter = greenCounter + 1;
}
else if (checkRGB(red, green, blue) == 2 && checkGrayscale(red, green, blue) == 1) {
    redCounter = redCounter + 1;
}
```

The reason for the threshold to be 20 pixels during the day is because daylight masks most of the traffic light and not many pixels become visible. At nighttime, traffic light shines intensely, and so the concentration of green/red pixels is much higher. Other light sources also tend to interfere. White street light, for example, has a greenish halo that could be mistaken for green light. So increasing the threshold at sunset decreases the amount of false positives (code below).

```
if (times>18.33) {
    greenThreshold = 1000;
    redThreshold=2500;
}
```

The ZeroIt class recalls the CameraMaster class when a traffic light is identified as red or in the case of no traffic light found. This is because if the light is red, then the user needs to know when it turns green. The markers in the GPS algorithm map out intersections with traffic lights. So if no traffic light is detected at an intersection, then the user will have to take another picture to identify the traffic light color (code below).

```
if(control==1) {
    ZeroIt.this.finish();
    startActivity(intent);
}
```

```
if(control2==1){
    count= new CountDownTimer(5000,1000) {
        @Override
        public void onTick(long millisUntilFinished) {

        }
        @Override
        public void onFinish() {
            ZeroIt.this.finish();
            startActivity(intent);
        }
    }.start();
}
else if(control2==2){
    ZeroIt.this.finish();
}
```

Also, in the code above, the camera is called again only after a countdown timer counts 5 seconds. This is done for the following two reasons. Firstly, the phone's processor and internal memory need a few milliseconds to reinitialize the camera API. If the camera is called with no time delay, the app will automatically crash due to the processor's limited speed. Secondly, a delay before calling the camera again is good since the traffic light may have time to turn green if it was identified as red before. Once the traffic light is identified as green, the program goes back to the GPS algorithm (code above). The GPS algorithm locks on the marker that was identified previously and no new markers will be analyzed until the user moves to at least 20 m away from the previous marker. This means that if the user stays in the same position (next to the traffic light) after taking a picture, the algorithm will wait till the user moves away before starting to identify other intersections (code below).

```
if(control2==1){
    showDistance.setText(""+location.distanceTo(dest)+" m");
  if (location.distanceTo(dest)>20){
      control2=0;
      showStatus.setText("Waiting for new intersection");}}
```

Below is the UML diagram of the camera API

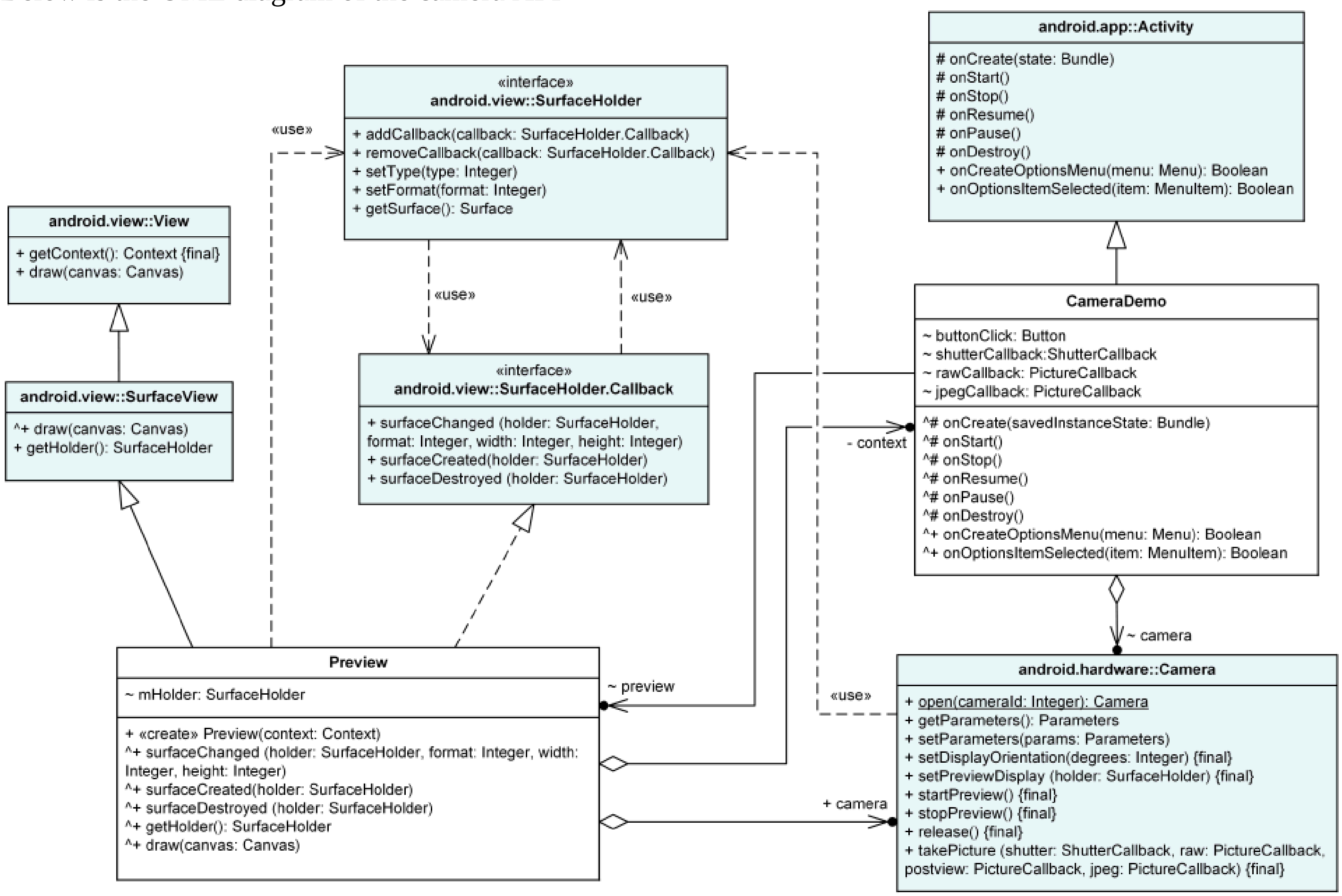


Figure 27 UML diagram of the traffic light detector

In the above figure, the camera is created with a CameraDemo object. This object is initialized by calling a new Camera object (the Camera class returns a CameraDemo object). In the Camera class, the basic layout for the camera is initialized. The orientation of the image can be flipped with the function "setDisplayOrientation". The display of the images is set with a SurfaceHolder object. Then the startPreview() function is called to open the camera and begin displaying image frames. Once the button is clicked, the function SurfaceHolder.callback is called and the image taken is returned as a Bitmap object. Once the picture is taken, the camera preview is destroyed with stopPreview().

4.4 Implementation of Obstacle Detection System

4.4.1 Schematic Diagram

In the Figure below, the schematic diagram is presented. The schematic contains the components on the printed circuit board. Connectors are placed on the PCB as to allow the components to be connected easily. The ultrasonic sensors and the infrared sensor connect directly to the ports of the microcontroller. The vibration motors are being driven by a discrete n-channel MOSFET driver circuit which supplies the necessary amount of current to the motors.

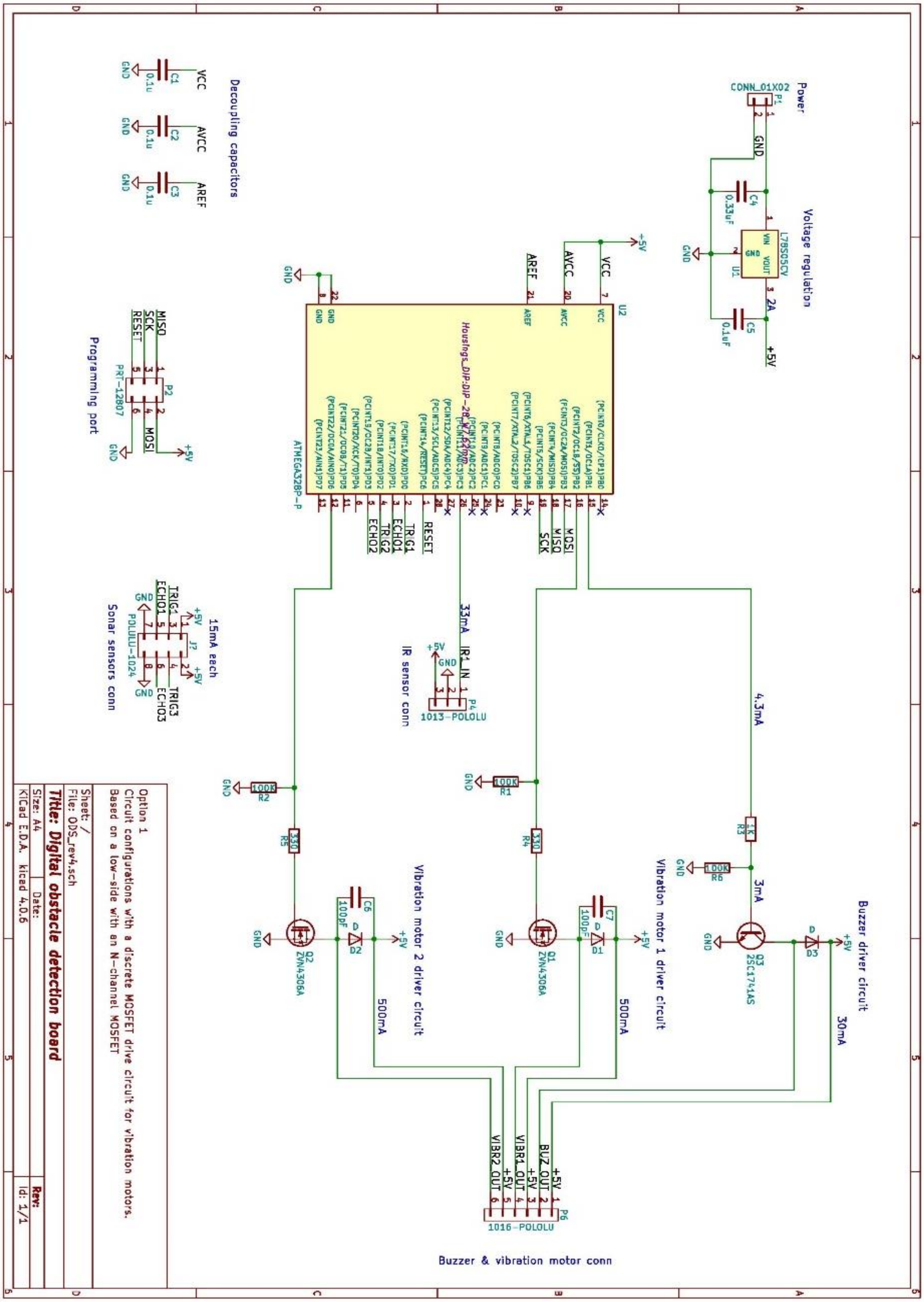


Figure 28 Schematic Design

The HC-SR04 converts reflected ultrasonic waves into an electrical digital signal. The sonar sensor produces a pulse-width proportional to the distance to the obstacle. An atmega238p microcontroller receives the signal from the sonars and processes it to determine the distance in meters (Refer to Figure 6 for the schematic diagram of the obstacle detection system). The microcontroller is programmed to ignore all obstacles that is farther than 1.00 meter. The microcontroller is then programmed to send a pulse-width-modulated (PWM) signal to the vibration motors based on the recorded distance. If the obstacle is located within 1 and 0.5 meters of the sonars, a PWM with a duty-cycle of 70 % is sent to the motors. If the obstacle is located within 0.5 meters and 0.03 meters of the sonars, a PWM with a duty-cycle of 100 % is sent to the motors. This allows the user to know how far the obstacle is. Also, since each sonar controls one of the vibration motors, the user can easily determine whether the obstacle is located on his left, right or center.

Since, the maximum output current of an atmega328p isn't sufficient to drive the vibration motors. A motor driver circuit will be used to drive the motor. The vibration motors need 500 mA of current to start spinning and around 180 mA of current at rated load, or maximum rpm. A discrete N-channel MOSFET drive circuit is used to drive the vibration motors. The discrete motor driver is shown in Figure 5.

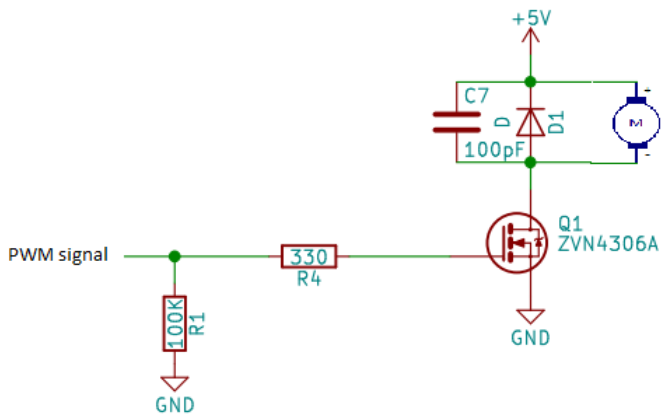


Figure 29 Discrete n-channel MOSFET motor driver

The Q1 N-MOSFET can provide the necessary amount of current to the vibration motors and also be switch on or off with a voltage at the gate of the transistor. The diode D1 prevents any current from damaging the MOSFET when the motors are turned off. The 100 pF capacitor serves to reduce the high frequency electro-magnetic wideband noise generated by the motor. The 100 kΩ pull-down resistor ensures that the MOSFET is switched off when no active signal is present.

The step/drop-off detection system uses a Sharp GP2Y0A02YK with a detection range between 20 and 150 cm. The IR sensor sends an analog signal to the microcontroller which is then converted into measured distance. The algorithm in the microcontroller code is able to identify the presence of a step or a drop-off based on the measured distance to the IR sensor. The sensor will normally measure a distance of 0.5 meters as it points diagonally towards the ground (Refer to Figure 2). If the IR sensor measures a distance that is less than 0.4 meters, then a step is detected. If the IR sensor measures a distance that is more than 0.7 meters, then a drop-off is detection. Upon detection of a step or a drop-off, the microcontroller will send a signal to the buzzer, emitting a buzzing noise to alert the user.

In order to power the obstacle detection system, the power rating of the sensors motors and buzzers as well as the average operation time by the user are all taken into consideration. The average user time of operation is around 3 hours. This requires at least 3Ah of charge from the battery. Also, the battery will provide a maximum of 2A of current at any given time. The battery used in the system is a 12 volt, 4.5 Ah valve-regulated lead-acid battery. Lead-acid batteries are safer, and easier to maintain compared to other types of battery. A low-dropout voltage regulator is used to step-down the voltage from 12V to 5V. Capacitors are connected between the input, output and ground of the regulator in order to suppress voltage ripples.

4. 4. 2 PCB

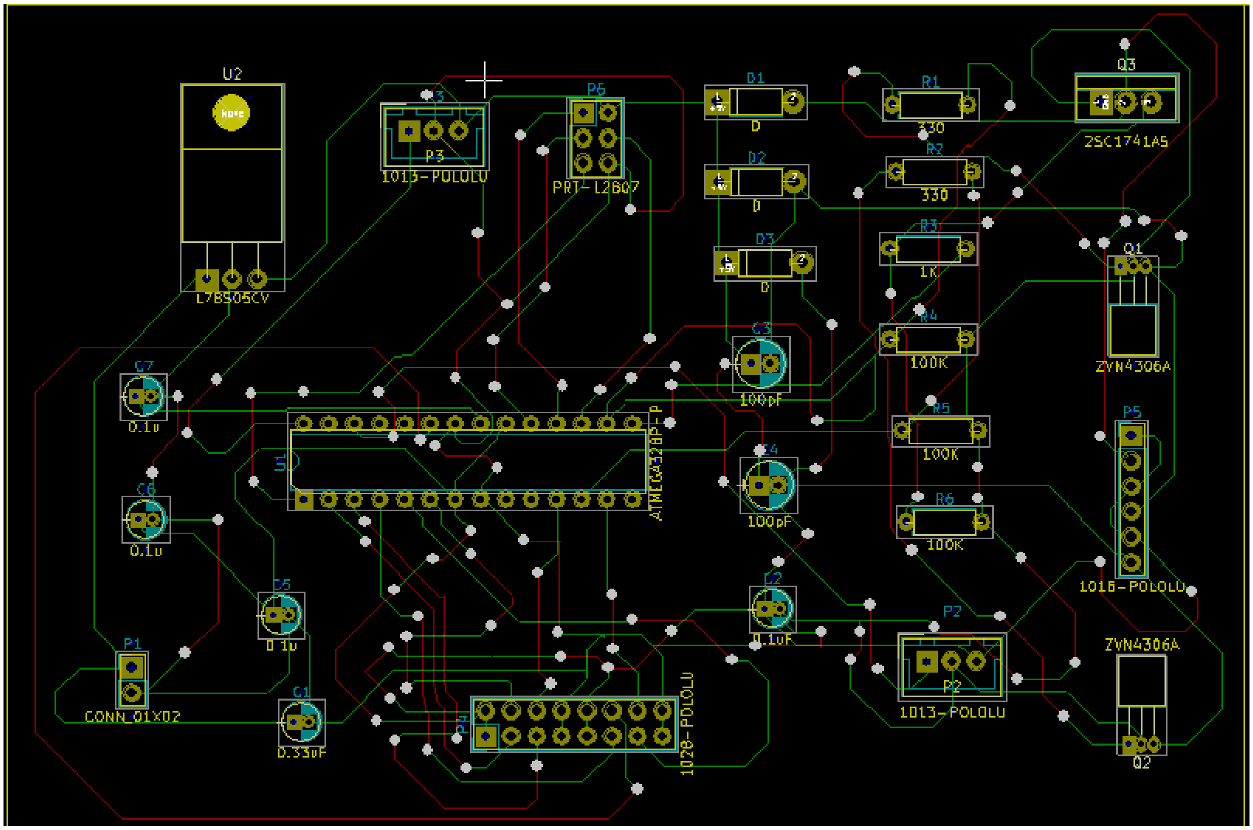


Figure 30 PCB Design

For designing the PCB, we first needed to do the schematic and choose the components. We had to select whether we want these components to be through whole or SMD (surface mounted). For the resistors and capacitors, we need to know whether we are using axial types or radial. Then, regarding the footprints, we need to know each footprint for each component we use in our design. In total, we have 25 components that are used for the obstacle detection system. We tested these components and they all worked. Specifically, we used 4 resistors, 9 capacitors, 5 connectors, and microcontroller. For the connectors, we used 1x3 for the only IR sensor we are using and 1x6 connector for the MOSFETS. Doing the PCB, we chose to make the design small so it'd be more professional. And we made sure to follow all the instructions regarding the voltages and currents that are going through the components. Thus, we chose the thickness of the wire carefully and followed closely the instructions that Mr. Jeff told us to follow. We also managed to use two layered (front and back), the back are all the connections in and out of the pins and the front are the connections when the vias are created. We checked the design with Mr. Jeff and we are still working on optimizing the layout and minimizing the number of the vias. This should be the last attempt and the PCB should be ready to be approved and printed to be used in the final demo.

4.4.3 Bill of materials

Item number	Design Reference	Description	Quantity	Unit	Manufacturer	Manufacturer part number	Supplier	Supplier P/N	Unit cost (CAD)	Extended cost (CAD)
1	IRsensor1-IRsensor2	IR sensor, 4-30 cm, 33 mA, 4.5-5.5 V	2	pcs	Parallax Sharp	GP2Y0A41SK0F	ABRA Electronics Corp.		$22.33	$44.66 (pick up at store)
2	P1	1x2 pins straight female header	1	pcs	POLOLU	1012	POLOLU	1012-POLOLU	$0.49	$0.49
3	P5	1x6 pins straight female header	1	pcs	POLOLU	1016	ABRA Electronics Corp.	1016-POLOLU		ABRA Electronics Corp. Cost: $0.76
4	P4	2x8 pins straight female header	1	pcs	POLOLU	1028	ABRA Electronics Corp.	1028-POLOLU	$1.53	$1.53
5	P2-P3	3-pin connector, (5-pack)	1	pcs	JST Sales America Inc.	JST XH-style	ABRA Electronics Corp.	BAT-CON-2S	$5.59	$5.59
6	P6	2x3 (6 pin - 0.1 spaced) IDC Breadboard Helper	1	pcs	Adafruit	2105	ABRA Electronics Corp.	2102-ADA	$1.61	$1.61
7	JST wire	3-Pin, 6" length, 26 AWG	2	pcs	POLOLU	1799-POLOLU	ABRA Electronics Corp.	PRT-09915	$2.10	$4.20
8	SonarSensor1-SonarSensor4	2.5-5.5V, 3.1mA	4	pcs		HC - SR04	ABRA Electronics Corp.	HC-SR04	$6.99	$27.96
9	Buzzer1	4-8V, 30mA	1	pcs		BUZ-120	ABRA-Electronics	BUZ-120	$1.48	$1.48
10	VibrationMotor 1-VibrationMotor 2	1.8-6.0V	2	pcs	Jinlong Machinery & Electronics, Inc.	ZJQ24-35F580C	Digi-Key	1670-1026-ND	$8.74	$17.48 + $8.00 shipping
11	U1	8-bit AVR, 1.8-5V, 1.1mA	1	pcs	Atmel	ATmega328p	ABRA-Electronics	ATMEGA 328P-PU	$6.93	$6.93
12	Battery	4.5Ah, 12V	1	pcs	B B Battery	BP4.5-12-T1	Digi-Key	BP4.5-12-T1-ND	$40.27	$40.27
13	U2	5V, 2A	1	pcs	STMicroelectronics	L78S05CV	Digi-Key	497-1468-5-ND	$1.03	$1.03
14	Walker	Walker with basket and 4 wheels	1	pcs	Drive Medical	10257RD-1	Amazon Canada	10257RD-1	$103.56	103.56 +$31.70 shipping = **$135.26**
15	Q1-Q2	N-Channel 60V 1.1A (Ta) 850mW	2	pcs	Diodes Incorporated	ZVN4306A	Digi-Key	ZVN4306A-ND	$2.13	$4.26
TOTAL	**$304.77 CAD**	**$350.49** (with tax)								

Table 5 Bill of Materials

5. Measurements and system Validation

In order to check the validation of the implementation of this product, we created the detailed test plan as follow for all system respect to the specifications.The following procedure sequences are to verify the functionality of the system as specified in the specifications section of this report.

Part 1 - Navigation

The test condition: the tester will get to the destination by only listening and following the instructions of the navigator. He cannot follow another direction that he thinks is correct, even if he is lost. The testing tool is the person testing the app. The limitation of having someone follow the instructions of a navigator is disorientation, not hearing the instructions well, following another route he thinks is better or faster.

1. Start the app and ensure it doesn't crash
2. Short press on one of the buttons on the screen, listen to the voice instruction to locate "Bank"
3. Long press on the Bank button to start navigation
4. Follow the voice instructions, start walking and notice if it provides the correct turning signal at the intersection.
5. When it's reported to reach the destination, measure the distance from you and the desired location and compare it with our desire specification. (5-10m)
6. Check the battery usage from system.

Part 2 - Obstacle Detection

The test condition: the tester must feel the vibrations to assume the presence of an obstacle; without relying on visual cues. The testing tool used here is the person testing the detector. He may not feel the vibration motors well if he is wearing gloves or his cell phone vibrates at the same time as the vibration motors.

a) Testing the entire subsystem

1. Place a 7''x11'' sheet of paper 0.2m/0.5/1m/1.5 m away from the left sensor and note whether there are vibrations
2. Place a school bag instead and repeat step 1
3. Place a garbage can instead with a 40 cm diameter and 50 cm high and repeat step 1
4. Place a person and repeat step 1
5. Place a person 1.5 m away from the center of the walker and note the vibrations
6. Repeat step 5 at 1 m
7. Repeat step 5 at 70 cm
8. Place a person the left of the walker 90 cm away and note the vibrations
9. Repeat step 8 on the right sensor
10. Move the person to 20 cm away from the left sensor and note the vibrations
11. Repeat step 10 on the right sensor
12. Ask a person to run past the sensors and note the vibrations
13. Repeat step 13 at walking speed instead
14. Take the walker outside, place it 50 cm away from a wall and note the vibrations
15. Bring the walker to the edge of a sidewalk and note the buzzing sound
16. Bring the walker to the top of a staircase and note the buzzing sound
17. Place a school bag of the floor, advance the walker towards it and note the buzzing sound

b) Testing the circuitry

The tool used is a multimeter, and so accuracy issues could change the measurements.

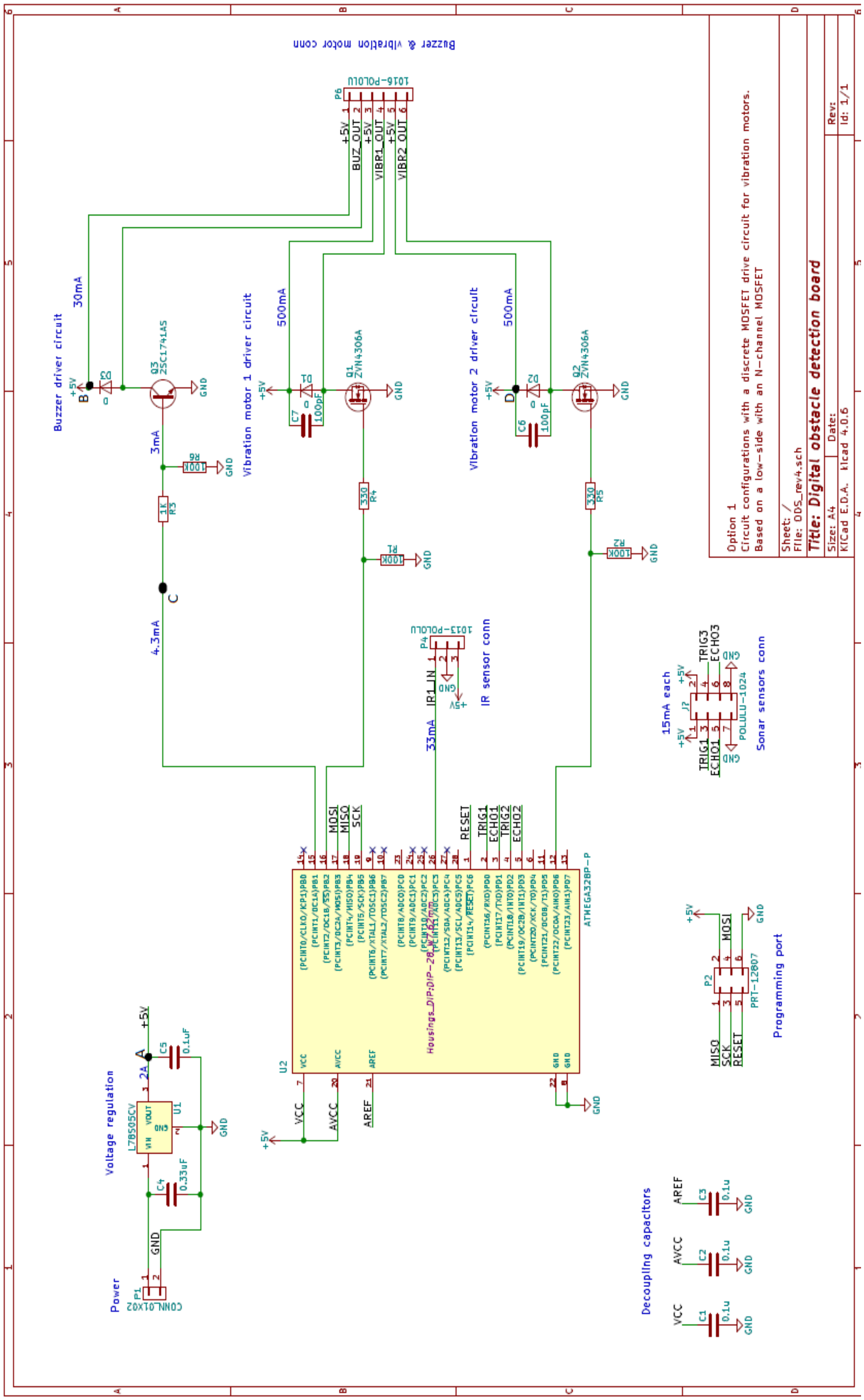


Figure 31 Schematic Design

1. With a multimeter, measure the voltage at Node A
2. With a multimeter, measure the voltage and current at Node B
3. With a multimeter, measure the current at Node C
4. With a multimeter, measure the voltage and current at Node D

Part 3 – Traffic Light Detection

The test condition: only stop when the app requires it, and only cross the street when the app determines that the traffic light is green. The tools used to test are traffic lights and the GPS satellites that are on a geocentric orbit around the earth. The limitations of the traffic light are incompatible shapes, long waiting times, different rgb codes. The limitation of the GPS satellites are misalignment, mismatched orbit speed, atomic clock malfunction.

1. Verify that the app displays current GPS coordinates
2. Walk towards a marked intersection and note whether the camera opens or not
3. Stand at 4 m from the traffic light and set the phone's vertical angle to 90 degrees (clockwise)
4. Note whether the traffic light appears in the top of the image
5. Repeat steps 3-4 for an angle of 88, 85, 82, 80, 75, 70
6. Stand at 8 m from the traffic light and set the phone's vertical angle to 90 degrees (clockwise)
7. Note whether the traffic light appears in the top of the image
8. Repeat steps 6-7 for an angle of 88, 85, 82, 80, 75, 70
9. Open the App "Pre-demo traffic light" and take pictures of traffic lights with the following shapes: arrow and circular
10. Note whether the algorithm identifies the green light
11. With the App "Pre-demo traffic light", take pictures of green traffic lights at 6 am, 8 am, 10 am, 3 pm, 5 pm, 7 pm, 9 pm, 12 am.
12. Note if green light is identified

Part 4-System Level

1. Take the walker outside when the temperature is -20 C
2. Note if any component malfunctions
3. Take the walker outside when the temperature is +25 C
4. Note if any component malfunctions
5. Take the walker outside when the relative humidity is 10%, 20%, 40%, 60%,80%, 90%, 95%
6. Note if any component malfunctions

6. Result

6.1 Navigation

It's able to find the desired destination, which is the "bank" according to the test plan. The buttons are very easy to be located since the voice output is clear. Navigation start when a long press is applied to the bank button. During the navigation, it gave correct signals of turning at each intersection, and at the end we reach the bank we wanted to go to and we were only 4 meters away from the location which meet the specifications of the navigation system.

6.2 Obstacle detection

6.2.1 Size of the obstacle

obstacle	size of obstacles(m*m)	distance(m)	result
paper	0.29*0.21	0.2	yes
school bag	0.50*0.25	0.5	yes
garbage bin	1.00*0.40	1.0	yes
human	1.75*0.65	1.5	no

Table 6 Obstacle sizes

The obstacle detection is able to detect obstacle size as small as a piece of paper (facing directly to sensors) within a range of 1.0 meter. When there's something 1.5 meter away from the system, it will not be detected. This is exactly what we expected.

Region testing and vibration results

region	distance	obstacle	result
left	0.2	human	left
right	0.2	human	right
center	1.5	human	no vibration
center	1.0	human	both
center	0.71	human	both

Table 7 Distance of obstacle

The voltages and currents at each node are presented in the table below.

Node	Voltage (V)	Current (mA)
A	5	NA
B	5	30
C	NA	4.3
D	5	500

Table 8

The results indicate that our obstacle detection system is able to distinguish obstacle in different regions within the range. The vibration which is the output are correctly being generated.

6.3 Traffic Light Detector

The current GPS coordinates were displayed by the app. The camera opens when a marked intersection is reached. Tried it at 9 different intersections, and at every single intersection the camera opened. The table below displays the results for steps 3-8 in the Test Plan section.

Distance from Traffic Light (m)	Vertical angle of phone (degrees)	Traffic Light visible?
4	90	No
4	88	No
4	85	Yes
4	82	Yes
4	80	No
4	75	No
4	70	No
8	90	No
8	88	Yes
8	85	Yes
8	82	Yes
8	80	No
8	75	No
8	70	No

Table 9 Results for vertical angle orientation of the phone

Arrow and circular shaped traffic lights were successfully identified.

The table below shows the results for steps 11-12

Time of day	Green Light identified?
6 am	Yes
8 am	Yes
10 am	Yes
3 pm	Yes
5 pm	Yes
7 pm	Yes
9 pm	Yes
12 am	Yes

Table 10 Results for identifying green light at different times in a day

6.4 System level

No system failed or malfunctioned at temperatures of -20 C. The temperature of 25 C could not be tried outside since we are still in the winter season. However, the system functions well indoors where the temperature is around 25 C. The table below shows the results for steps 5-6. Since the weather does not permit us to perform steps 5-6 outdoors, the testing was conducted inside a bathroom with the shower on to create humidity.

Relative humidity	Malfunctions?
10%	No
20%	No
40%	No
60%	No
80%	No
90%	The left ultrasonic sensor stopped working
95%	Both ultrasonic sensors stopped working

Table 11 Relative Humidity

Humidity affects the speed of sound; it increases its velocity. As the air gets more humid, it condenses with water droplets. The sound waves will propagate much faster and so bounce back at higher velocities to the sensors. Since the code for measuring distance relies on the speed of sound in humidity of 20% (common average humidity), faster sound waves cause inaccurate distance measurements (smaller than usual).

7. Conclusion

This capstone project was completed within the required deadline. All the testing was performed on the different subsystems, and they proved to be reliable and efficient. The navigation system was accurate to within 5-10 m of the target location. It is more than enough accuracy to pinpoint the user to his desired location. Also, the buttons function as they were designed. One short click gives the name of the button, and a long click will initialize Google Maps to the preset destination in the button. The obstacle detection functioned according to the specifications. It detected human sized obstacles at 1 m away and closer. It also detected other obstacles, such as paper, garbage bin and school bag. Also, the detector distinguished between center, left and right obstacles. The voltages and currents at the nodes were all correct and corresponded to the values written on the schematic. The vibration motors were strong and the vibrations were felt throughout the arms. The traffic light detector performed as designed and in accordance with the specifications. The camera was able to catch the traffic light in the top of the image at distances of 4-8 m. The optimum angle of the phone was 85 degrees going clockwise from the horizontal. The best time to use the traffic light would be anytime except between 12:00 and 2 pm where the sun is at its brightest level and causes a lot of interference, therefore undermining the accuracy of the algorithm. The entire project, with all three subsystems on-board, was tested in controlled environments to ensure that it functions in accordance with the specifications. At temperatures of -20 C and +25 C, it operated normally and no malfunctions occurred. The humidity test was conducted in a bathroom with the shower on. At 80% humidity, both sensors became inaccurate. They were identifying distances that were off the actual values. At 90% humidity, the left ultrasonic sensor stopped working. At 95% humidity, both sensors stopped working. However, all the other components were working quiet well. Humidity affects the speed of sound; it increases its velocity. As the air gets more humid, it condenses with water droplets. The sound waves will propagate much faster and so bounce back at higher velocities to the sensors. Since the code for measuring distance relies on the speed of sound in humidity of 20% (common average humidity), faster sound waves cause inaccurate distance measurements (smaller than usual). This project was performed and implemented in accordance with the design and the specifications for each subsystem were respected. The project remains a prototype and more development needs to be done in the future to turn it into a viable product. Other teams in the future could take over this project and enhance it.

Appendix

CEAB Case Study

Analogical case study

The Laser Cane is an electronic mobility aid developed by Bionic Instruments. The first model of the Laser Cane was produced in 1965 for the U.S. Veterans Administration. Over the decades, Bionic Instruments has continuously improved their product through numerous iterations. Similar to our product, the Laser Cane uses sonar sensors to determine the presence of obstacles. In addition to sonar technology, the Laser Cane uses Laser technology to further detect certain obstacles and to determine drop-offs. Bionic Instruments has performed extensive clinical evaluations with the Veteran Administration during the seventies in order to solve the issues that the visually impaired experience. This is the reason that the Laser Cane was chosen as a case study. The company made great efforts to conduct accurate studies on the performance of the product. Bionic Instruments also made sure that users received proper orientation and mobility training to ensure full understanding of the product. Today, users credit the Laser Cane for its reliability and safety.

One concern from users, however, is the inconsistency of the Laser Cane with how its laser technology detects obstacles of certain materials. For example, light from the laser will pass straight through transparent glass, and therefore the laser sensors will not detect it.
As Strobel, Fossa, Panchura, Beaver, and Westbrook (2003) stated, "Assistive technology can mean a difference between gainful employment and unemployment; between success in the educational system and failure; and between integration into the community and segregation." [3]. ETAs increase the user’s safety, mobility, and travel speed. They offer the user a greater level of independence and a readiness to travel in unknown or unfamiliar environments, and a reduction of stress and discomfort. This is the goal of our capstone project. We want the visually impaired user to be able to function within society despite his disability. In addition to giving the visually impaired greater independence, the safety and wellbeing of the user is very important. Bionic Instruments utilized the latest technology in their Laser Cane to allow the user to properly detect

pedestrians, obstacles, steps and drop-offs. Bionic Instruments had to design their product in accordance to safety regulations and requirements of the government in order to ensure the safety and wellbeing of the visually impaired. When it comes to our capstone project, we have a moral and legal obligation to ensure the safety of the user.

Contemporary Practice:

In the current market, there exist a variety of electronic travel aids (ETA) for the visually impaired, however, ETAs are not yet widespread. This is attributed to the fact that earlier devices did not meet the needs of visually impaired people. Most developers have failed to realize the importance of developing their product in conjunction with the user. In order for the product to fully meet the needs of the visually impaired, the user must be involved during research and development. For example, having a prototype that users can evaluate and give proper feedback on, can help developers know what changes need to be made for the next step. This is an approach that Bionic Instruments took when developing the Laser Cane. Proper orientation and mobility training should be given to the user to make sure the user knows how to use the product properly. After market release, it is important to determine how much of a positive impact the product has had on the user. Proper evaluation should be undertaken to properly answer this question.

Public perceptions of technology

Over the last 15 years, assisting technology for the visually impaired has allowed many users to complete everyday chores that previously wouldn’t be possible for them. Assisting technology has allowed people with low vision to write documents, browse the internet, and send and receive emails through the use of screen reading software, special talking and Braille devices and screen magnification software. Smartphones today, however, offer solutions to visually impaired people that are more accessible and cheaper than ever. “Cell phones and tablets have revolutionized the way people who are blind or visually impaired interact and use technology,” explains a blogger by the name of Sandy’sView who suffers from low vision, “[they] enable people with visual impairments to do things that were previously impossible, or – at the very least – challenging. It is now possible for the iPhone, for example, to describe the color, shape and size of objects to someone who is blind thanks to an app called TapTapSee. Furthermore, other apps, such as Be My

Eyes, connect blind or visually impaired individuals with a sighted person, who will then assist them by describing things.”

Technology assessment and choice

During the initial development of the GPS navigation, the obstacle and traffic light detection systems, we wanted to find a way to clearly relay the information to the visually impaired user. Since the information cannot be relayed visually, the only other way for the user to effectively receive information is through vibro-tactile and audio feedback. We also wanted to avoid confusing the user by overwhelming him with too much information. We decided to use vibrational feedback to notify the user of obstacles and voice commands to navigate the user as well as notify him of the presence and color of traffic lights. Since the voice commands will be coming from the smartphone, we realized the user will have difficulty hearing the commands properly. A first solution was to have the user wear earphones, but we realized this is dangerous as we are taking away the user’s sense of hearing, who already has no sight. A better solution was to simply use speakers to allow the user to hear the voice commands.

Another issue we came across was the placement of the vibration motors. Initially we wanted to place the vibration motors on the torso of the user, but we decided to place them on the handles of the walker. The reason for this decision was that we wanted to simplify the way the user is to setup the system and head outside. Having the vibration motors placed permanently on the walker as opposed to putting them on the user every time the he has to go outside greatly simplifies the use of the system.

Electronic Travel Aid for Visually Impaired

User Manual

User manual

Materials

- Drive Medical Four Wheeled Walker
- Two HC - SR04 ultrasonic sensors (3-400cm)
- One Sharp GP2Y0A02YK infrared sensors (20-150cm)
- Two Jinlong Machinery ZJP16-70E310 vibration motors
- One Bluetooth clicker
- One BB-Battery VRLA 12V 4.5Ah battery

Android Software

- Blind guide (available on Play Store)

User interfaces

Figure 32 - User interface of pre-defined location

The user interface of the smartphone will read out the predefined location as you swipe you finger along the screen. Simply long press the screen for 2 seconds to select desired destination.

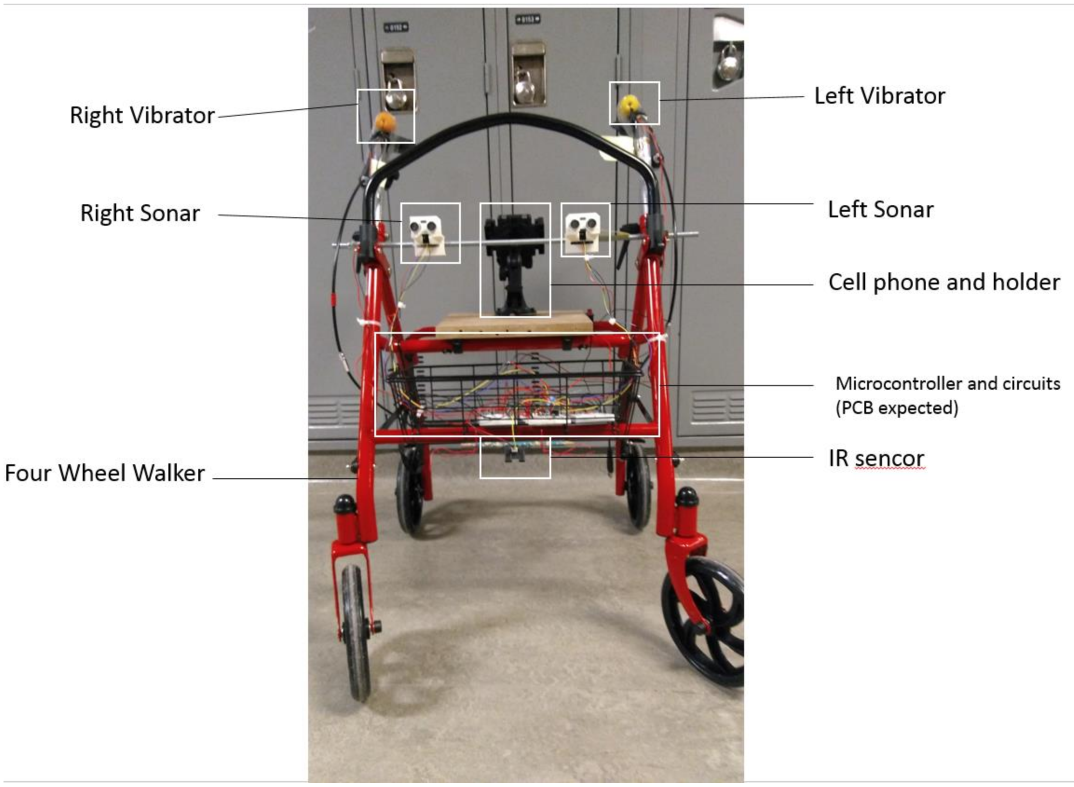


Figure 33 - Overall system walker

1. Right vibrator - Vibrates when an obstacle is detected within the right region
2. Left vibrator - Vibrates when an obstacle is detected within the left region
3. Right sonar - Scans the right region for obstacles
4. Left sonar - Scans the left region for obstacles
5. Cellphone holder - Used to hold the smartphone
6. Four wheeled walker - Platform of the system
7. Circuit board - Contains microcontroller and electronic components
8. IR sensor - Scans for steps and drop-offs
9. Bluetooth clicker (shown in Figure 3) - Turns on the camera when button is pressed

Figure 34 - Bluetooth clicker

<u>Before use</u>

1. Before operating the electronic travel aid (ETA), it is important that you receive orientation and mobility training to fully understand how to use the system.

2. When operating the electronic travel aid walker for the first time, make sure to charge the VRLA battery for at least 5 hours. A 3 Stage Sealed Lead Acid Battery Charger (CH-LA1210-UL) can be used to charge the battery. A constant voltage of 14.7V can also be applied to the terminals of the battery for charging (Refer to Figure 1).
3. Before using the ETA system, have an orientation and mobility expert input your predefined address locations into your smartphone.

Recommended actions

1. Make sure to charge the battery after every use.
2. Lock the brakes when walker is not in use.
3. Do not walk to fast as this increases the chance of collision and injury

How to charge battery with constant voltage applied at terminal

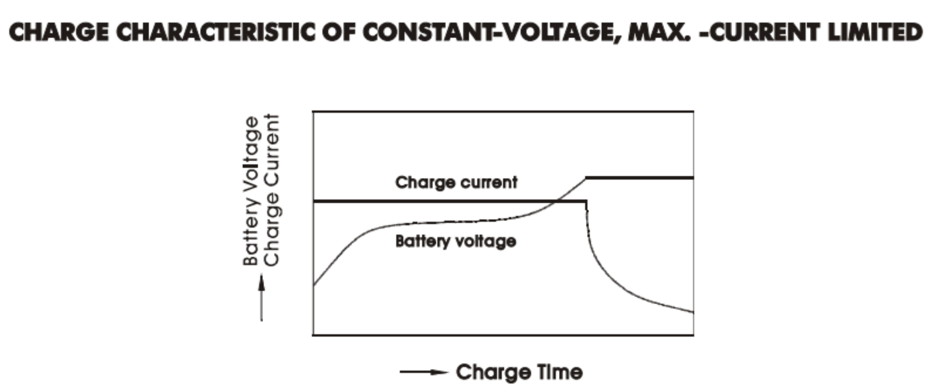


Figure 35 – Constant voltage battery charging characteristics

To charge the battery using a constant voltage, you will need a DC supply capable of outputting 150mA at 14.7V. Apply a voltage of 14.7V across the terminals of the battery. Make sure to limit the current to 150mA. Charging may take between 3 and 4 hours.

Procedure

1. Turn switch on. You will know the switch is on if vibration motors are active when an object is in right front of the walker.
2. When selecting desired destination, scan your finger across the screen of the smartphone to hear the names of the pre-defined location.
3. Long press for 2 seconds to select your destination.
4. Follow the spoken navigation directions.
5. As you walk, the vibration motors will be activated when obstacles appear in front of you.
6. The buzzer will notify you when a step or sidewalk curb is present.
7. When approaching an intersection, the smartphone will output a message notifying you that you are at an intersection.
8. If no traffic light is present, cross the street with caution.
9. You will know you are at your destination when the smartphone reads out the message “You have arrived at your destination”.
10. Return to step 2 when entering a new destination.

Specifications

Walker specifications

Handle height: 31” – 37”

Unit dimensions: 23.5” (W) x 25.5” (L)

Seat dimensions: 12” (L) x 12” (W)

Weight : 18.6 lbs.

Weight limit: 300 lbs.

Sonar sensor specifications

Beam angle: 15 degrees

Voltage rating: 2.5 – 5.5V

Current rating: 3.1 mA

Dimensions: 1.77”x0.79”x0.59”

Detection range: 3 – 400 cm

Infrared sensor specifications

Operating voltage: 4.5 – 5.5V

Operating temperature: -10 to +60°C

Detection range: 20 – 150 cm

Vibration motors

Operating voltage: 2.5 – 7.5V

Operating temperature: -20 to +70°C

Weights: ~14g

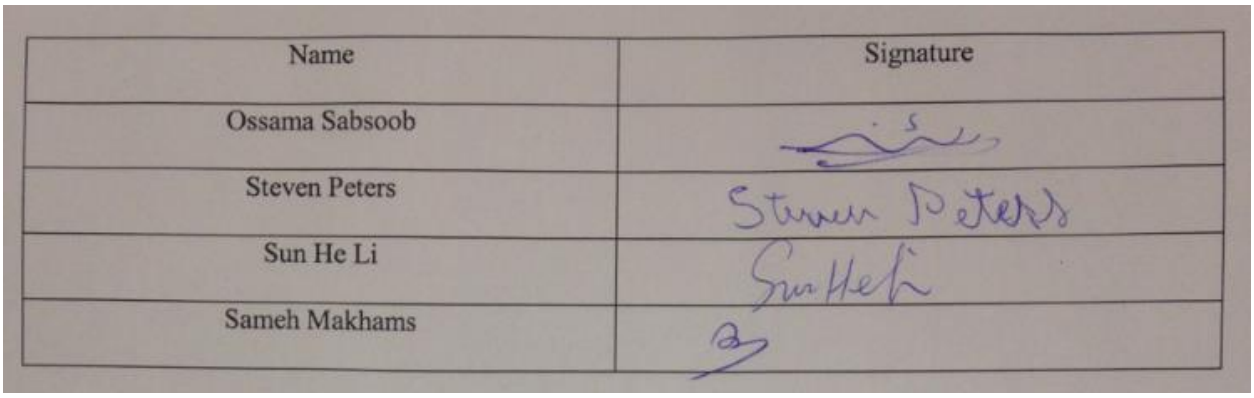
Topic Approval

This is the signatures of the team members of group 13.

Throughout writing the report titled "Outdoor Street Navigation for a blind person", we had distributed the tasks equally and fairly and got the report approved by the professor Dr. Yousef Shayan on March 24, 2017. The tasks assigned to a particular team member is not solely done by that member. Everyone also worked in other areas of the project, and assisted each other when a team member required help.

We certify that this submission is the original work of members of the group and meets the Faculty's expectations of Originality.

Team members' signatures



Name	Signature
Ossama Sabsoob	
Steven Peters	
Sun He Li	
Sameh Makhams	

References

[1] Jianwei Gong, JG, Yanhua Jiang, YHJ, Guangming Xiong, GMX, Chaohua Guan, CHG & Gang Tao, GT (2010). *The Recognition and Tracking of Traffic Lights Based on Color Segmentation and CAMSHIFT for Intelligent Vehicles*. (1st Ed.). San Diego: University of California.

[2] Guo Mu, GM, Zhang Xinyu, ZXY, Li Deyi, LDY, Zhang Tianlei, ZTL & An Lifeng, ALF (2015). *Traffic light detection and recognition for autonomous vehicles*. China: The Journal of China Universities of Posts and Telecommunications.

[3] Sylvain Cardin, S.C, Daniel Thalmann, D.T & Frederic Vexo , F.V (2012). *Wearable Obstacle Detection System for visually impaired People*.Switzerland: Ecole Polytechnique Fédérale de Lausanne.

[4] A.D, Android Developer (2016, April 8). *android.hardware.camera2*. Retrieved October 2, 2016, from androiddevelopers.com, https://developer.android.com/reference/android/hardware/camera2/package-summary.html

[5] A.D, Android Developer (2015, May 8). *android.hardware.vibrator*. Retrieved October 2, 2016, from androiddevelopers.com, https://developer.android.com/reference/android/hardware/camera2/package-summary.html

[6] Capi, G. (2012). *Development of a New Robotic System for Assisting and Guiding Visually Impaired People*.

Proceedings of the 2012 IEEEInternational Conference on Robotics and Biomimetics

[7] Chaccour, K., & Edit, J. (2015). *Multisensor guided walker for visually impaired*

elderly people.

2015 International Conference on Advances in Biomedical Engineering (ICABME)

[8] Bates, J. Boote, and C. Beverly, *"Psychosocial interventions for people with a dementing illness: A systematic review,"*

Journal of Advanced Nursing, vol. 45, no. 6, pp. 644–658, 2004.

[9] H. A. Yanco, "*Wheelesley, a robotic wheelchair system: Indoor navigation and user interface,"* in Lecture Notes in Artificial Intelligence: Assistive Technology and Artificial Intelligence, V. Mittal, H. Yanco, J. Aronis, and R. Simspon, Eds., 1998, pp. 256–268.

[10] *A preliminary evaluation of the bionic instruments--Veterans Administration C-4 laser cane: a final report*. (1973). Washington: National Academy of Sciences.

[11] Working Group on Mobility Aids for the Visually Impaired and Blind, Committee on Vision. (1986). *Electronic Travel Aids: New Directions for Research*. National Academies Press

[12] Roentgen, ,., & Ganert J, G. (2008). *Inventory of Electronic Mobility Aids for Persons with Visual Impairments: A Literature Review*. NY: Journal of Visual Impairment & Blindness.