# ASHESI UNIVERSITY

## DESIGNING WEARABLE OBSTACLE DETECTION AND AVOIDANCE TECHNOLOGY FOR THE VISUALLY-IMPAIRED

**CAPSTONE PROJECT**

B.Sc. Computer Engineering

**Tamisha Dzifa Segbefia**

**2021**

**ASHESI UNIVERSITY**

# DESIGNING WEARABLE OBSTACLE DETECTION AND AVOIDANCE TECHNOLOGY FOR THE VISUALLY-IMPAIRED

**CAPSTONE PROJECT**

Capstone Project submitted to the Department of Engineering, Ashesi University College in partial fulfilment of the requirements for the award of Bachelor of Science degree in Computer Engineering.

**Tamisha Dzifa Segbefia**

**2021**

## DECLARATION

I hereby declare that this capstone is the result of my own original work and that no part of it has been presented for another degree in this university or elsewhere.

Candidate's Signature:

………………………………………………………………………………………………

Candidate's Name: Tamisha Dzifa Segbefia

Date: Monday, April 26th, 2021

I hereby declare that preparation and presentation of this capstone were supervised in accordance with the guidelines on supervision of capstone laid down by Ashesi University College.

Supervisor's Signature:

………………………………………………………………………………………………

Supervisor's Name:

………………………………………………………………………………………………

Date:

………………………………………………………………………………………………

# Acknowledgements

I would, first and foremost, like to thank God for the ability to do. In Him, I live, move, and have my being.

I would also like to thank my supervisor, Dr. Siby Samuel, for his guidance and support throughout this period.

To Dr. Nathan, Dr. Rosca, and the other faculty who helped me through confusing moments of this project, I say, "thank you." To my family, who supported me throughout and gave me moral support when I wanted to give up, I appreciate you. To my friends, especially Issifu, Jean, Mngiesssss, Kristen, Shamo, Parker, Karen, and Stephen, I could not have done this without you.

Finally, to the Engineering Class of 2021—your energy is infectious, and the help you give to support your own is commendable. I am grateful.

# Abstract

Assistive technology for the visually impaired is developed regularly, but many of these solutions are either bulky, difficult to maneuver, or expensive. Visually impaired people need a way to avoid obstacles and move around in safety. Due to limited visibility, the person cannot tell with confidence if there is an obstacle approaching them. This project proposes a wearable obstacle and avoidance system for the visually impaired housed on an embedded microcontroller. The system consists of both RFID and sonar localization techniques for static and dynamic obstacles. Both approaches work well in terms of obstacle detection and avoidance, but further research is needed to integrate the processes to complement each other fully. A simultaneous localization and mapping (SLAM) approach, though computationally expensive, is also considered. The results show that it is possible to combine both RFID and sonar localization; however, this approach might be better suited for a less constrained system.



*Keywords*: RFID, ultrasound, localization, visually impaired

# Table of Contents

DECLARATION ........................................ i

Acknowledgements ........................................ ii

Abstract ........................................ iii

Chapter 1: Introduction ........................................ 1

1.1 Background ........................................ 1

1.2 Problem Statement ........................................ 2

1.3 Aims and Objectives ........................................ 2

1.4 Related Works ........................................ 2

1.5 Limitations ........................................ 4

Chapter 2: Literature Review ........................................ 5

2.1 RFID technology ........................................ 5

2.2 Sonar Technology ........................................ 7

2.3 Feedback Mechanisms ........................................ 7

Chapter 3: Design ........................................ 9

3.1 Design Requirements ........................................ 9

3.2 System Requirements ........................................ 9

3.3 System Constraints ........................................ 10

3.4 Design Decisions ........................................ 11

3.5 System Components ....... 11
3.5.1 Ideal System Design ....... 11
3.5.2 Alternative System Design ....... 14
3.6 Circuit Schematic ....... 16
3.6.1 Ideal System Schematic ....... 17
3.6.2 Alternative System Schematic ....... 18
3.7 System Flowcharts ....... 18
3.7.1 Ultrasonic Sensor Flowchart ....... 18
Chapter 4: Methodology ....... 20
4.1 Ultrasonic Sensor Obstacle Detection and Avoidance ....... 20
4.2 RFID Localization ....... 22
4.2.1 The Nearest Neighbors Algorithm ....... 23
4.2.2 Bayesian Interference Algorithm ....... 26
4.3 Feedback System ....... 26
4.4 SLAM Implementation ....... 26
Chapter 5: Results and Conclusion ....... 29
5.1 Results ....... 29
5.1.1 Ultrasonic Sensing Module ....... 29
5.1.2 RFID localization module ....... 30
5.1.3 SLAM Simulation ....... 30
5.2 Limitations ....... 31

5.2.1 Ultrasonic Sensor Obstacle Detection and Avoidance ............................31
5.2.2 RFID localization..........................................................................32
5.2.3 SLAM algorithms ..........................................................................32
5.3 Future Works ................................................................................32
5.4 Conclusion ....................................................................................33
References....................................................................................................34

# Chapter 1: Introduction

## 1.1 Background

Blindness is a lack of vision or a loss of vision that cannot be corrected by glasses or contact lenses [1]. According to [2], the worldwide incidence of blindness was pervasive to the extent that the World Health Organization set up an initiative known as “Vision 2020 – The Right to Sight." The initiative aimed to eliminate avoidable blindness around the world. Nearly one-fifth of the cases of blindness are caused by refractive error [2]. Visual impairments (VIs), on the other hand, affect the sense of vision. Types of VIs include loss of vision in the central field of view (macular degeneration) and loss of peripheral vision. Some common causes of VIs are blunt force trauma, hemianopia, cataract, and glaucoma.

One of the leading causes of blindness and visual impairment is blunt force trauma. Blunt force trauma refers to a physical injury that occurs due to forceful impact on the human body [3]. A study done in [4] presents evidence that delayed vision loss accompanies blunt force trauma. Another cause of blindness and VI is hemianopia. Hemianopia occurs after a person has experienced a stroke and can lead to visual disability [5]. The final cause of blindness and VI discussed is glaucoma. The term glaucoma refers to a group of optic nerve disorders that progressively cause vision loss, as explained in [6].

At least 2.2 billion people worldwide are either blind or have a form of visual impairment, as stated in [7]. As in [2], the prevalence of visual impairments and blindness is higher in sub-Saharan Africa than in other parts of the world. Blind or visually impaired people have difficulty performing activities for themselves. Examples of these activities are walking from one place to another and searching for an item. As such, it is difficult for these individuals to be integrated members of society. Due to this, the development and implementation of assistive technology have become critical.

Blindness and other visual impairments have been shown to increase a person's mortality. From a study conducted in [8], mortality is significantly increased in the blind. Also, [9] showed that blind people have higher mortality rates compared to sighted counterparts. Thus, blindness can lead to shortened life expectancy.

## 1.2 Problem Statement

Visually impaired people need a way to avoid obstacles and move around in safety. People with visual impairments lack individual mobility because there is a lack of real-time obstacle detection. Due to limited visibility, the person cannot tell with confidence if there is an obstacle approaching them.

## 1.3 Aims and Objectives

This project aims to design wearable obstacle detection and avoidance technology for the visually impaired. It also aims to make the system low-cost and affordable, specifically for places in Ghana and Africa.

## 1.4 Related Works

Various forms of assistive technologies have been developed to aid the visually impaired in their day-to-day lives. One of such techniques is the use of stereoscopic sonar technology, as introduced in [10]. In this paper, the authors developed a system that uses a sonar system to send out a signal and report back to the user via vibrotactile feedback (vibration).

This technology's components are fitted onto a jacket with the shoulder pads' sensors to help with the sensing range. While this method was successful during the test run, one major disadvantage is its limitation in sensing a door. The door has poles on both sides, causing the shoulder pads to vibrate simultaneously and the user to believe he/she has approached a wall.

Another such technique is the use of RFID-based technology, as explained in [11]. The authors developed the system to help the user find an item inside a medicine cabinet. While this approach was practical on a small scale, it would become quite costly as the solution is scaled; this is because every item that needs to be detected must be tagged to be identified by the RFID reader.

Research in [12] presents a system aptly titled LANDMARC, which stands for "Location Identification Based On Dynamic Active RFID." LANDMARC makes use of fixed reference tags known as landmarks to provide location calibration. While LANDMARC has several advantages, the major drawback of the system is the need for active RFID tags. Active RFID tags are expensive to obtain and make the system expensive for low to middle-income countries. In addition, LANDMARC is not explicitly designed to be used by the visually impaired. Extra work would be needed to adapt the technology for use by the visually impaired.

The final existing technique discussed is an Android-based object recognition technique presented in [13]. In this paper, the authors used computer vision with a mobile phone's help to provide real-time feedback to the user concerning the type of object and whether it is in motion. While this technique is efficient, especially given the prevalence of Android phones on the market, one potential shortcoming is that the phone would have to be held in the user's hand everywhere they go as it scans the environment.

## 1.5 Limitations

One major limitation of this project is the “low-cost” constraint. As the project is designed to be low-cost, there is a limit to the types of materials and components used. However, care was taken to ensure that the low-cost components selected were not of inferior quality.

# Chapter 2: Literature Review

The obstacle avoidance and detection system designed in this paper rely on two leading technologies — radio frequency identification (RFID) technology and sonar technology. This section summarizes and evaluates other related works, literature, and articles that influenced the design and concept of the system.

## 2.1 RFID technology

An RFID system comprises two main components: the transponder and the reader (also known as the interrogator). The transponder is on the object to be tracked/identified, and the reader is with the person doing the tracking or identification. RFID transponders can contain or store data, and this data can be forwarded by the interrogator to another system for processing and analysis, as explained in [14]. Much of the literature and applications surrounding RFID technology uses RFID in tracking systems and not in obstacle avoidance systems. This is generally so because for an RFID reader to identify an obstacle, the obstacle needs to have been tagged first, making it not ideal for identifying dynamic objects.

Authors such as Sammouda and Alrjoub built a mobile blind navigation system in [15] for visually impaired persons within King Saud University (KSU). In this system, key landmarks and points along the KSU campus are already tagged with RFID tags. The user speaks his/her destination, and the mobile system calculates the shortest path to get there. The main advantage of this system is that the RFID-based technology is paired with GPS and Wi-Fi, making user navigation easier. However, as is the case with any RFID-based technology, the obstacle needs to be "pre-tagged" before the system can determine it.

The research in [11] introduces an RFID-based system that enables the user to find an object, specifically a medicine bottle in a medicine cabinet. In [15], for example, RFID was combined with GPS and Wi-Fi to provide improved localization and accuracy for the user. The system in [16], however, makes use of RFID and stereo vision (image processing). The system in [17] combines the RFID technology with an existing object in most visually impaired persons' routines— a walking stick. These authors assume that all the salient obstacles or landmarks have been "pre-tagged" and that there would be no dynamic obstacles that pop up during the user's navigation.

The significant point of agreement between these articles is that RFID alone is not enough to warrant an accurate obstacle detection and avoidance system for users, especially since obstacles need to be tagged before the reader can identify them. However, one technique proposed in [16] but not in the others is the inclusion of Bayes' rule in the system's programming. Bayes' rule is used to calculate the probability of an RFID tag's location after being identified by the reader to improve the accuracy of obstacle localization. Then, the stereo camera processes the region of interest (ROI) proposed and determined by Bayes' rule. The inclusion of Bayes rule makes the system more accurate for the visually impaired user and is a significant consideration for this project. However, [16] uses a stereo camera to process the region of interest for the presence of obstacles. While this works for their application, image processing is too computationally involving and requires, for the most part, a PC, thus rendering the system not wearable.

## 2.2 Sonar Technology

Cardin et al.'s [10] research uses a stereoscopic sonar system to detect potential obstacles and send vibrotactile feedback to the user based on the obstacle's positioning. The system's working principle is this: determination of the direction from which the obstacle comes from → determining (a) the height of the obstacle and (b) whether the floor is clear of obstacles → user positioning. It is important to note that the authors created an intuitive system that is simple and easy to understand (e.g., a vibration on the left shoulder means an obstacle on the left). Also, the sensors were fitted on the shoulders of the wearable jacket to help improve the sensing range of the device (i.e., eye-level range). Unlike in [15] and [16], sonar technology in [2] allows the user to pinpoint where the obstacle is without needing to be tagged. However, one of the gaps to note in [10] can best be visualized by imagining a user approaching a doorway. Since a doorway has doorposts on both sides, the user would receive vibrations from both shoulders. As such, the user might interpret that they are at a wall and might not be able to walk through the door successfully. Conversely, the user might approach a wall and believe that they have approached a door. Thus, the system would benefit from an update that better senses or determines the type of obstacle approached.

## 2.3 Feedback Mechanisms

This paper and project explore creating wearable obstacle detection and avoidance technology for the visually impaired. Many devices have been created for the visually impaired wearable technology space, with different technologies ranging from radio frequency identification (RFID) and near field communication (NFC), bio-sonar applications, and GPS technology. In addition to the obstacle detection and avoidance technology, several papers have utilized various feedback mechanisms such as haptic feedback, thermo-haptic, text to speech, and audio feedback.

The coin vibration motor (CVM) is a popular choice for a haptic feedback module. CVMs are small, can be affixed in place, and have a low cost compared to other motors such as stepper motors and servo motors. CVMs are Eccentric Rotating Mass (ERM) motors. This means that the motors rotate an unbalanced load to create the vibration effect. An external driving force, such as an operating voltage, causes the system to vibrate. As such, the coin vibration motor is an excellent choice for a haptic feedback system.

# Chapter 3: Design

The system described in this project has two subsystems—hardware and software. Additionally, the hardware subsystem can be broken down further into four modules—the processing module, the ultrasonic sensor module, the RFID sensing module, and the non-visual feedback module.

## 3.1 Design Requirements

*Table 3.1: Design requirements table*

| User Need | What will be measured | How to measure it (units) | Good value | Better value |
|---|---|---|---|---|
| **Affordable** | Cost of system | Ghana cedis | < 500 Ghana cedis | < 250 Ghana cedis |
| **Avoid obstacles** | Latency (time taken to detect and inform the user of obstacle) | Seconds | < 7 seconds | < 5 seconds |
| | Accuracy of non-visual feedback | Percentage (out of 100) | > 80 % | > 90 % |
| **Help user move** | Weight | Grams | < 200 grams | < 100 grams |
| **Easy to use** | Time taken to orient user on how to use | Time in days | < 7 days | < 4 days |

## 3.2 System Requirements

The system requirements table below outlines key functionalities the system must possess to be valuable to the user.

*Table 3.2: System requirements table*

| Type of Requirement | Name | Threshold / Type | Rationale |
|---|---|---|---|
| **Energy Consumption** | Number of hours it should last | 5 hours | The system should last for a reasonable number of hours |

| | | | |
|---|---|---|---|
| | | | before it needs to recharge so that it does not die on the user while in use. |
| | Type of battery | Lithium-ion battery | These are small and rechargeable and fit on wearable devices without taking up too much space. |
| **Feedback** | Efficient non-visual feedback mechanism | Haptic and audio feedback (i.e., vibration and sound) | Haptic feedback communicates the urgency of the closeness of the obstacle as it can vibrate harder to mean the obstacle is close; audio feedback is easily understandable by the user |
| **Cost** | Low-cost & affordable | -- | -- |

### 3.3 System Constraints

A significant constraint of the system is cost. One of the project's main aims was to design a low-cost and affordable system for places in Ghana and West Africa primarily. Care was taken to select relatively affordable components compared to existing solutions on the market.

As the system is designed to be implemented as wearable technology, the constraints of wearable technology affect the system. For example, wearables need to last for long hours while using little power. Due to this, the ATmega328P-PU microcontroller was selected because it has a low power consumption. The Raspberry Pi Zero and other similar devices were discarded because of their high-power consumption, even though they have faster processors than the ATmega. Also, wearable technology needs to be easily portable. This means that the overall system should not be large, bulky, or heavy. As such, the components to be selected needed to be small and lightweight enough for the user to carry.

## 3.4 Design Decisions

In selecting the brain of the system, the following Pugh chart was used. The value of each option's criterion is recorded below, with a (+) or (-) to indicate its performance relative to the baseline. The baseline, in this case, is an Arduino UNO. Table 3.3 below illustrates the selection process.

*Table 3.3: Pugh chart for the brain of the system*

| **Criteria** | **Weight** | **Arduino (baseline)** | **ATmega328P-PU** | **Raspberry Pi Zero** | **STM32F103C8T6** |
|---|---|---|---|---|---|
| **Cost (in GHS)** | 3 | 0 | GHS 15.10 (+) | GHS 28.70 (-) | GHS 27.90 (+) |
| **Power Consumption** | 2 | 0 | Low power (+) | High power (-) | Moderate power (+) |
| **Program Memory Size** | 1 | 0 | 32KB (-) | 512MB (+) | 64KB (+) |
| **Maximum CPU Frequency** | 1 | 0 | 20 MHz (-) | 1 GHz (+) | 72 MHz (+) |
| **PWM Peripherals** | 2 | 0 | 6 PWM pins (+) | Two pairs of PWM pins (+) | 1 PWM pin (-) |
| **Total** | -- | 0 | 5 | -1 | 5 |

## 3.5 System Components

### *3.5.1 Ideal System Design*

An ideal system for the project would be constructed using the components mentioned below. The ideal design takes into consideration the requirements mentioned earlier.

a) The ATmega328P-PU microcontroller was selected because of its low cost, relatively high storage, and adequate processing power.

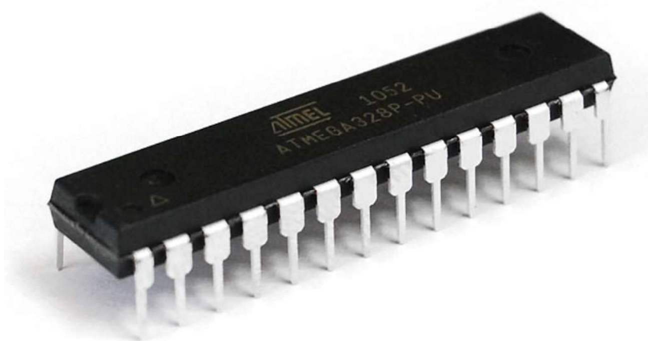


*Figure 3.1: An image of the ATmega328P-PU microcontroller*

b) The HC-SR04 ultrasonic sensor was selected because of its low cost, readability range (2 cm to 400 cm), and relatively high accuracy of 3 mm. This ultrasonic sensor is also supported with a variety of microcontrollers, including the ATmega328P-PU.


*Figure 3.2: An image of the HC-SR04 ultrasonic sensor*

c) The ID 12-LA RFID reader. This RFID reader was selected because it can read both active and passive tags, making it an ideal reader for RFID localization problems.


*Figure 3.3: An image of the ID 12-LA RFID reader module*

d) The Sparkfun coin vibration motor. This coin vibration motor is small enough to fit into wearable technology applications. It also has a low operating voltage that the ATmega328P-PU microcontroller can supply.

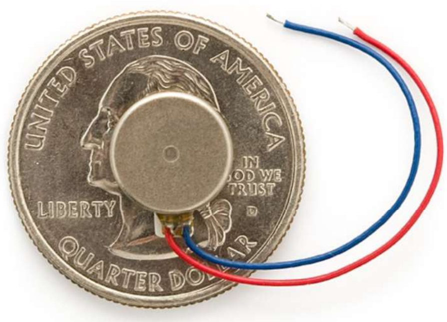


*Figure 3.4: An image of the Sparkfun coin vibration motor*

e) RFID tags (both active and passive)


*Figure 3.5: An image of an active RFID tag from Omni ID*

### 3.5.2 Alternative System Design

Due to time and cost constraints, an alternative system design was considered and designed. The following components make up the alternative system design, and this alternative design is what was implemented in Chapter 4.

a) The ATmega328P-PU microcontroller (Figure 3.1).

b) The HC-SR04 ultrasonic sensor (Figure 3.2).

c) The RC522 RFID reader was chosen for its low power mode, flexible interrupt mode, and fully programmable timers.

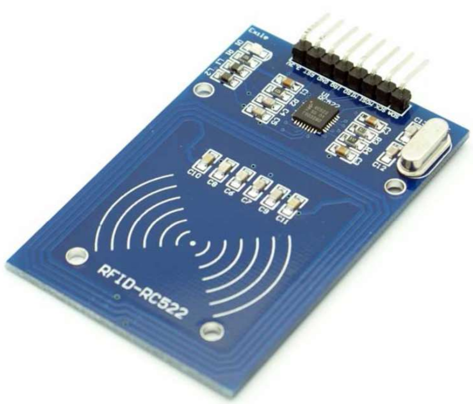


*Figure 3.6: An image of the RC522 RFID reader*

d) A piezoelectric buzzer. The piezoelectric buzzer can play various tones and sounds that can easily be distinguished from each other. This makes it an ideal component for audio feedback to the user.

*Figure 3.7: An image of a piezoelectric buzzer*

e) The 13.56 MHz RFID tags. These RFID tags are suitable for short to mid-range communication.

*Figure 3.8: An image of a 13.56 MHz RFID tag*

f) A Tower Pro MG996R servo motor


*Figure 3.9: An image of the Tower Pro MG998R servo motor*

## 3.6 Circuit Schematic

The circuit schematic diagrams were generated using the Fritzing software. Unused connections on the various microcontrollers and circuit components are grounded. The ideal and alternative system designs have a 5V power supply circuit using the LM 7805 voltage regulator, a 9V battery, capacitors, and resistors. The resultant 5V power supply is used to power the circuit.

### 3.6.1 Ideal System Schematic

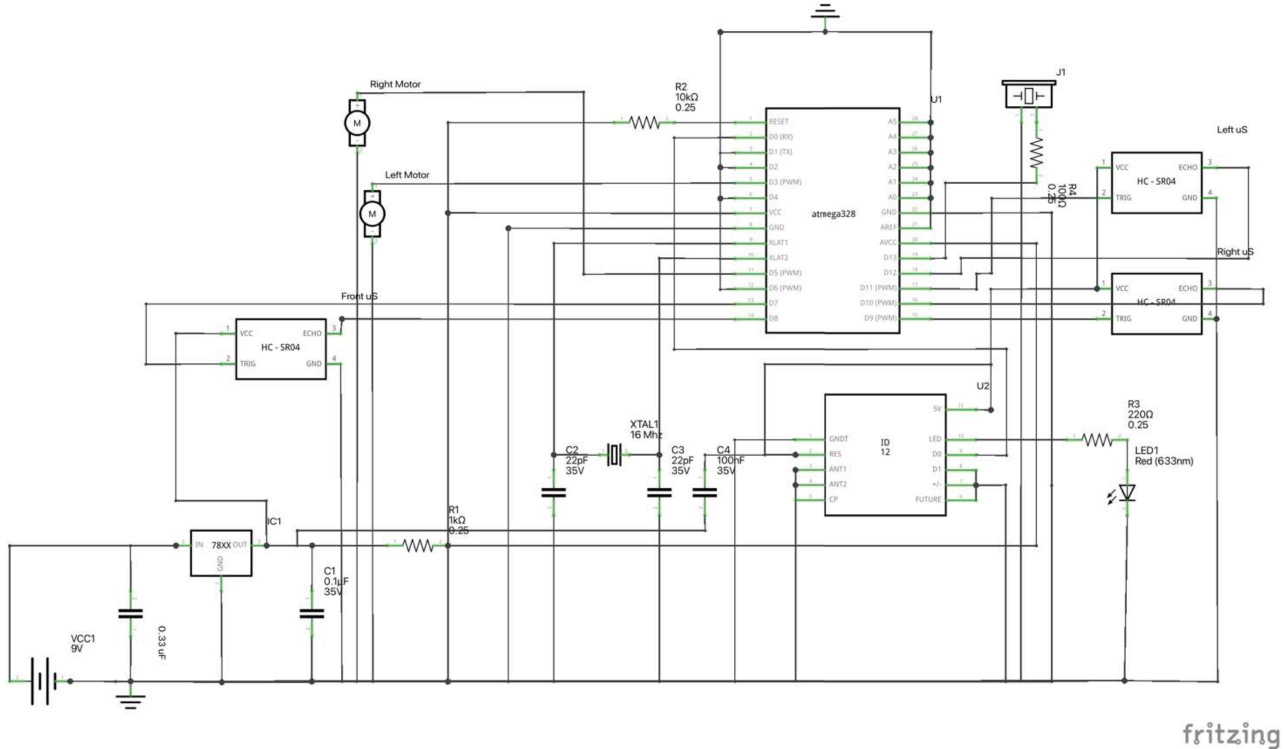


*Figure 3.10: Circuit schematic of the ideal system design*

Figure 3.10 consists of two coin vibration motors—one for the left and one for the right. These motors vibrate when there is an obstacle approaching the user. For example, an object approaching the user on the left would cause the left vibration motor to vibrate. For the basic control of the coin vibration motor, the leads are connected to a constant voltage DC source, and the motor will vibrate until the control is switched off. In the schematic above, the motor's constant voltage source will be connected to the digital output of the ATmega328P-PU pin.

There are also three ultrasonic sensor components— one for the forward direction and the other two for the left and right directions. In addition, the ID 12 LA RFID reader is connected to the ATmega328P-PU microcontroller to perform RFID localization, which is explained in the RFID Localization section.

### *3.6.2 Alternative System Schematic*

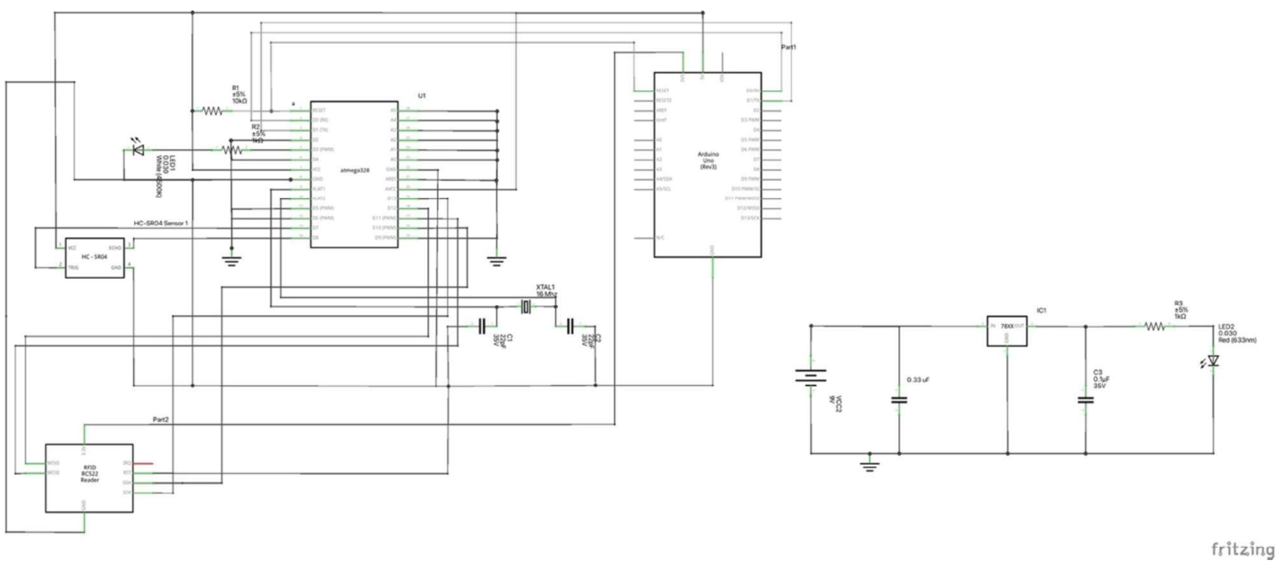


*Figure 3.11: Circuit schematic diagram of the alternative system design*

Figure 3.11 describes the alternative system design. In this system, there is one ultrasonic sensor instead of three. The ultrasonic sensor is mounted on a servo motor that rotates to the left and right. A piezoelectric buzzer sounds when the user approaches an obstacle and needs to change direction.

## 3.7 System Flowcharts

### *3.7.1 Ultrasonic Sensor Flowchart*

The following flowchart describes the system's behavior concerning the ultrasonic sensor module of the system. This module serves as both obstacle detection and obstacle avoidance.

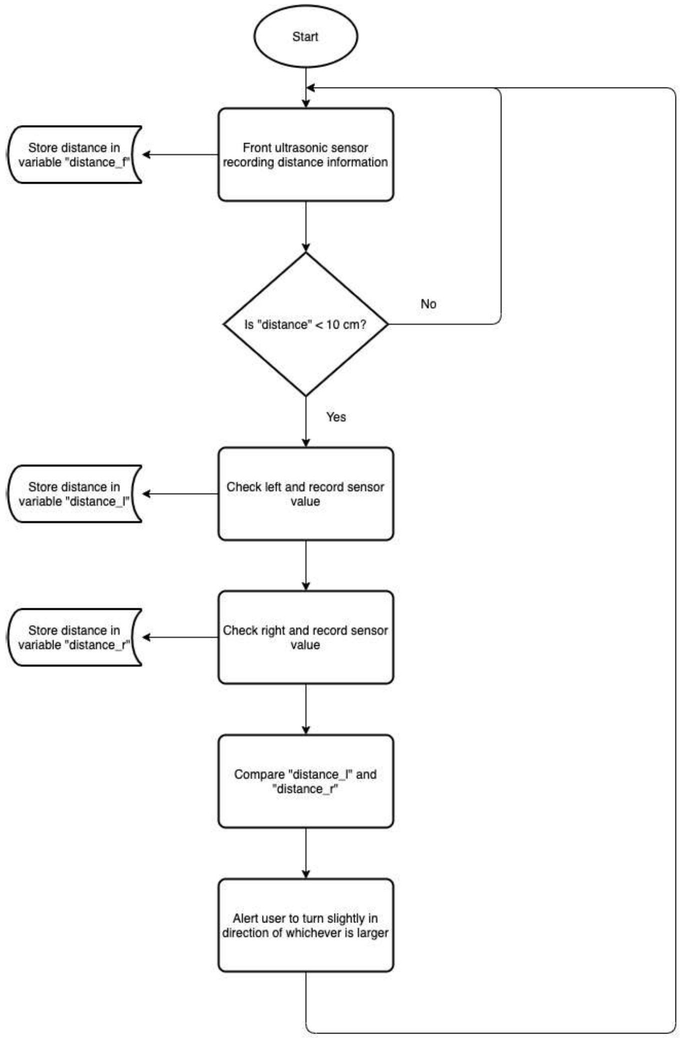


*Figure 3.12: Flowchart of the ultrasonic sensor subsystem*

# Chapter 4: Methodology

This chapter explains the steps taken to build, test, and implement the project, as described earlier. For the hardware aspect of the system, two breadboards housed the electronic components. For the software aspect of the system, code was written and uploaded to the ATmega328P-PU microcontroller via the Arduino IDE. An Arduino UNO was used as an in-system programmer for the microcontroller. MATLAB was also used to provide simulation capabilities. MATLAB was chosen as the application of choice for simulations because it can convert MATLAB code to C/C++ code to be uploaded on the resultant microcontroller.

## 4.1 Ultrasonic Sensor Obstacle Detection and Avoidance

The ultrasonic sensor subsystem is better suited for identifying dynamic obstacles in the user's path. It is also suitable for an outdoor environment that has multiple unknown and dynamic objects. Ultrasonic sensors operate on the principle of sending out a signal and waiting for an echo. The time interval between these two actions is calculated and converted to distance using the general formula $speed = distance / time$, where speed refers to the speed of sound (~343 m/s).

As shown in Figure 4.1 and Figure 4.2 below, three ultrasonic sensors were fixed on the breadboards – one pointing forward (uS1), one pointing to the left (uS2), and one pointing to the right (uS3). As the breadboard does not allow for diagonal alignment, the ultrasonic sensors were plugged into the breadboard horizontally and then tied to face the left and right, respectively. The threshold value for whether an obstacle was too close to the user was 10 cm for the test. The code written employed a line-follower robot algorithm technique and followed the general structure outlined in Figure 3.12. A fundamental assumption made in the code was that the three ultrasonic sensors would not have obstacles approaching all of them simultaneously. In addition to the audio output, textual output was printed to the serial monitor for debugging purposes.

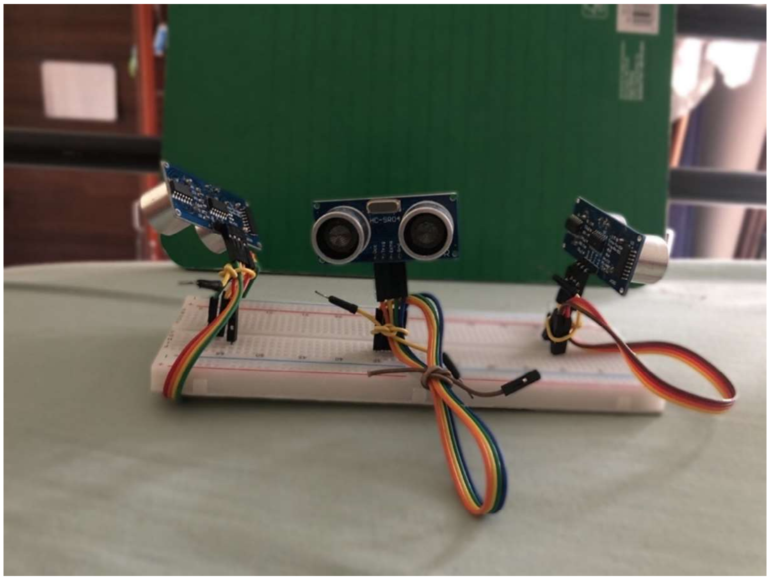

*Figure 4.1: An image of three ultrasonic sensors affixed to the breadboard*

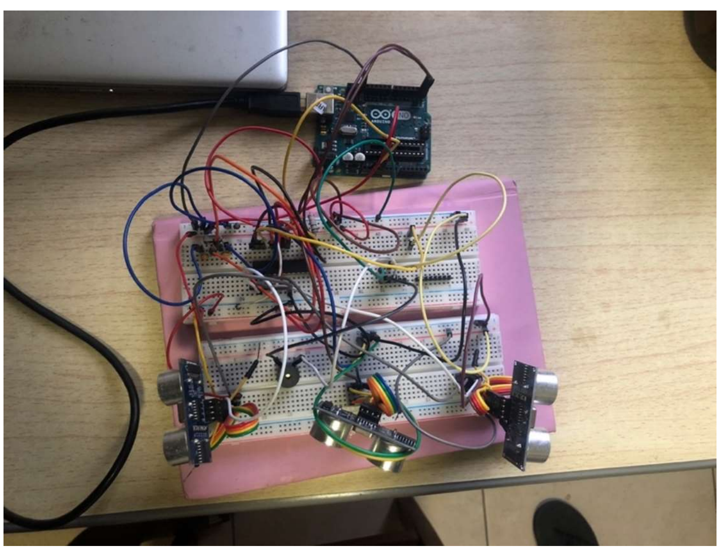

*Figure 4.2: The top view of the entire system with the affixed breadboards*

As described in Code Listing 4.1 below, no sound is emitted when the central view of the user is clear. However, once an obstacle blocks the user's central view, the tone for the left signal is sounded, which alerts the user to turn slightly to the left. The user knows when there is a clear path before them once the left signal stops sounding, and there is no other sound from the buzzer. The code is reminiscent of a closed-loop feedback system and line follower algorithm because it uses the ultrasonic data stream to correct its positioning.

```
void ultrasonic_sensors() {
    distance_l = sonar[0].ping_cm();
    distance_r = sonar[1].ping_cm();
    distance_f = sonar[2].ping_cm();

    if (distance_f >= OBSTACLE_THRESHOLD) {
        noNewTone(buzz_pin);
        Serial.println("Go forward");
    } else {  // if the threshold has been crossed
        // compare left and right and ask user to turn in the
direction of the larger one
        if (distance_l < OBSTACLE_THRESHOLD) { // assuming all three
cannot be less than the threshold at a time
            Serial.println("Go right");
            right_beep();
        }
        if (distance_r < OBSTACLE_THRESHOLD) {
            Serial.println("Go left");
            left_beep();
        }

        if (distance_l > distance_r) {
            Serial.println("Go left");
            left_beep();
        } else {
            Serial.println("Go right");
            right_beep();
        }
    }

    Serial.println();

}
```

*Code Listing 4.1: A code snippet of the ultrasonic sensor algorithm code*

## 4.2 RFID Localization

The RFID subsystem of the design is better suited for a more controlled indoor environment where landmarks and reference points have already been tagged. These reference tags contain location information which helps the reader determine its position based on various RFID localization algorithms. RFID localization is typically preferred to ultrasonic localization because the latter is easily susceptible to disturbance from environmental noise.

There are two main categories of RFID localization algorithms as explained in [18]: those that perform a calibration before the localization estimation is done and those that directly estimate location based on the received signal strength indicator (RSSI). Some examples of both types of algorithms are nearest neighbors, proximity, and Bayesian interference. The nearest neighbors algorithm was selected for the project because of its ease of implementation and low computational complexity.

*4.2.1 The Nearest Neighbors Algorithm*

The operating principle for the nearest neighbors algorithm is that the closer two points are, the smaller the difference between their received signal strengths. Based on this, an object can be localized by its neighbors. The equations used to estimate the RFID reader's position are shown below.

$$x = \sum_{i=1}^{k} w_i x_i$$

$$y = \sum_{i=1}^{k} w_i y_i$$

In the above equations, $k$ refers to the number of nearest neighbors used to localize the object. The coefficient $w_i$ is calculated based on the difference in the radio frequency signal strengths, and the equation is shown below. $m$ represents the number of anchor devices (for this paper, a value of 1 is assumed), $s_{ij}$ represents the RSSI at the $i^{th}$ reference point, and $s_j$ represents the RSSI of the localized point.

$$w_i = \frac{1 / \sum_{j=1}^{m} \left| s_{ij} - s_j \right|}{\sum_{i=1}^{k} \left( 1 / \sum_{j=1}^{m} \left| s_{ij} - s_j \right| \right)}$$

A challenge that occurred while implementing the nearest neighbors algorithm was that the RC522 RFID reader does not measure the RSSI value of the passive RFID tags. In addition, passive RFID tags do not typically retain RSSI values. Due to this challenge, the code was extended to manipulate the antenna gain of the RFID reader and return mock values for the RSSI. Although these were placeholder values, they served the purpose of helping the RFID reader localize the tag and, thus, localize itself. The code manipulating the antenna gain of the reader is shown in Code Listing 4.2 below.

```
int mock_rssi() {
    mfrc522.PCD_SetAntennaGain(0x01<<4);
    if (mfrc522.PICC_IsNewCardPresent() == 1) {
        // Serial.println("Level 1 ");
        return 1;
    } else
        mfrc522.PCD_SetAntennaGain(0x02<<4);
    if (mfrc522.PICC_IsNewCardPresent() == 1) {
        // Serial.println("Level 2 ");
        return 2;
    } else
        mfrc522.PCD_SetAntennaGain(0x03<<4);
    if (mfrc522.PICC_IsNewCardPresent() == 1) {
        // Serial.println("Level 3 ");
        return 3;
    } else
    mfrc522.PCD_SetAntennaGain(0x04<<4);
    if (mfrc522.PICC_IsNewCardPresent() == 1) {
        // Serial.println("Level 4 ");
        return 4;
    } else
    mfrc522.PCD_SetAntennaGain(0x05<<4);
    if (mfrc522.PICC_IsNewCardPresent() == 1) {
        // Serial.println("Level 5 ");
        return 5;
    } else
```

```
    mfrc522.PCD_SetAntennaGain(0x06<<4);
    if (mfrc522.PICC_IsNewCardPresent() == 1) {
        // Serial.println("Level 6 ");
        return 6;
    } else
    mfrc522.PCD_SetAntennaGain(0x07<<4);
    if (mfrc522.PICC_IsNewCardPresent() == 1) {
        // Serial.println("Level 7 ");
        return 7;
    } else
    // Serial.println("N/A ");
    return 0;
}
```

*Code Listing 4.2: Code snippet to produce mock RSSI values by manipulating the antenna gain*

The nearest neighbors method was implemented as shown in Code Listing 4.3 below.

```
void nearest_neighbor(int* ref_x, int* ref_y, int* ref_rssi, int k)
{
    // extending it would mean taking in arrays based on the points
around the reader
    // right now, I'm assuming that k = 2

    int s_j = mock_rssi();

    int w[] = {0, 0};

    for (int i = 0; i < k; i++) {
        int temp_num, temp_den = 0;
        temp_num = 1 / abs(ref_rssi[i] - s_j);

        for (int j = 0; j < k; j++) {
            temp_den = temp_den + (1 / (abs(ref_rssi[i] - s_j)));
        }

        int w_val = temp_num / temp_den;
        w[i] = w_val;
    }

    int x, y = 0;

    for (int j = 0; j < k; j++) {
        x = abs(x + w[j] * ref_x[j]);
        y = abs(y + w[j] * ref_y[j]);
    }

    Serial.print("x: ");
    Serial.println(x);

    Serial.print("y: ");
    Serial.println(y);
```

```
}
```

*Code Listing 4.3 : Implementation of the nearest neighbors algorithm*

#### *4.2.2 Bayesian Interference Algorithm*

While not implemented with the current system, the Bayesian interference technique is an important RFID localization approach. This technique is a reader localization technique, which means that it estimates the position of the RFID reader given the information of the tags within the reading range of the reader [19].

## 4.3 Feedback System

A piezoelectric buzzer was used to provide feedback to the user. If there was no obstacle blocking the central view of the user, no sound was played. The absence of a sound from the buzzer confirmed to the user that they could continue walking forward. However, if the user needed to turn left, code was written to produce sound with a frequency of 1 kHz that played every 100ms. If the user needed to turn right, code was written to produce sound with a frequency of 131 Hz (i.e., Note C3). This sound is also played every 100ms.

## 4.4 SLAM Implementation

Simultaneous localization and mapping (SLAM) algorithms are used in autonomous robotics and artificial/virtual reality games. SLAM algorithms are essential because they allow a system to create a map and localize itself when venturing into an unknown environment. There are two main categories of SLAM algorithms – visual SLAM (vSLAM) and lidar SLAM. vSLAM algorithms retrieve data from cameras and use this data to build the map for the robot or system. Key inputs to vSLAM algorithms are the position and orientation of the camera. Lidar SLAM algorithms receive inputs from laser or distance sensors and use this data to build the map for the system. For this project, a lidar SLAM technique was simulated. This is because the system's ultrasonic sensors can provide distance data that can be used in the localization and mapping of the environment. In addition, vSLAM algorithms are more susceptible to noise, and the data from camera sensors can be less accurate than data from lidar and ultrasonic sensors.

A simultaneous localization and mapping (SLAM) algorithm was not implemented on the ATmega328P-PU due to its limited RAM and relatively low processing capabilities. However, a simulation of this was done in MATLAB with emphasis on the lidar SLAM technique. The first lidar SLAM simulation in MATLAB was done using loaded lidar scans of a parking garage. The following parameters were set – range, map resolution, loop closure threshold, and search radius.

The loop closure threshold is an important parameter that helps the system determine whether it has visited a previously scanned region. In this simulation, the value selected was 360. The higher the loop closure threshold value is, the more accurate the mapping. However, this depends on the quality of the scans. The range parameter refers to the maximum range of the lidar sensor. In this case, the value selected was 19.2 meters.

After setting the various parameters, the resultant lidar SLAM object was used to build an occupancy map with the data that had been collected. This map was visualized using the built-in MATLAB function, and the results were shown in a figure window. The code for the lidar SLAM implementation is shown in Code Listing 4.4 below.

```
load garage_fl1_southend.mat scans
scans = scans(1:40:end); % select every 40th scan

maxRange = 19.2; % meters
resolution = 10; % cells per meter

slamObj = lidarSLAM(resolution, maxRange);
slamObj.LoopClosureThreshold = 360;
slamObj.LoopClosureSearchRadius = 8;
```

```
for i = 1:numel(scans)

    addScan(slamObj,scans{i});

    if rem(i,10) == 0
        show(slamObj);
    end
end

[scansSLAM,poses] = scansAndPoses(slamObj);
occMap = buildMap(scansSLAM,poses,resolution,maxRange);
figure
show(occMap)
title('Occupancy Map of Garage')
```

*Code Listing 4.4: MATLAB code for lidar SLAM implementation*

# Chapter 5: Results and Conclusion

## 5.1 Results

### *5.1.1 Ultrasonic Sensing Module*

The first test for the ultrasonic sensing module involved placing the system in a stationary place and moving dynamic obstacles in front of the various ultrasonic sensors. The screenshot in Figure 5.1 showed the output of the serial monitor when various obstacles were placed in the path of the system. In this scenario, there were obstacles uS1's and uS3's path; thus, the user was instructed to turn left to avoid the obstacle.

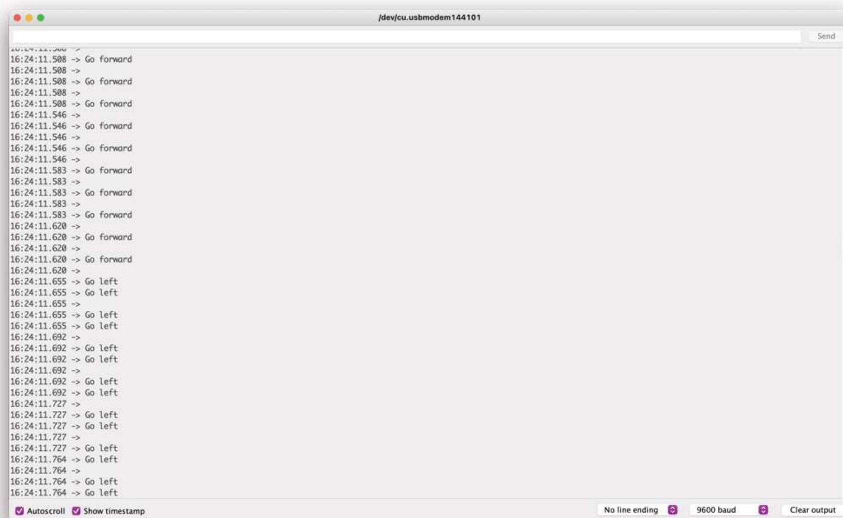


*Figure 5.1: Serial monitor output instructing the user to turn left*

The second test for the ultrasonic sensing module involved having a sighted user walking around with the system in hand. The system was held at waist level to mimic the wearable technology being designed for the user's waist. The third test for the ultrasonic sensing module involved having a blindfolded sighted user walking around with the system in hand. Once again, the system was held at waist level as in the second test.

In testing the system, it was realized that a series of dynamic obstacles would approach the user. These dynamic obstacles have different sizes, weights, and densities. Also, from the datasheet of the HC-SR04, it was noted that obstacles with smooth surfaces are more likely to be detected correctly than obstacles with rough or bumpy surfaces. Thus, in conducting the tests for the system, different obstacles were tested, including poles, doorways, chairs, and couches. The system was also tested with a human being as an obstacle. In each scenario, the system's response was recorded, and these results were used to estimate the accuracy of the system concerning ultrasonic obstacle detection and avoidance.

### *5.1.2 RFID localization module*

As mentioned in RFID Localization, the RFID localization module was tested with different RFID tags that had location information written as data on the tag. These RFID tags served as reference/landmark tags and were placed on stationary objects in the user's path. As these tags came into the reading range of the RFID reader, the location data was retrieved and used to find the approximate location of the user. Testing the system made it clear that the RFID module is better suited for user localization than obstacle detection.

### *5.1.3 SLAM Simulation*

From the first lidar SLAM simulation using lidar scans from the parking garage, the images can be found below. Figure 5.2 shows the SLAM map that was built with the lidar scan data. The output of the map makes it clear where the specific obstacles are located and how best the system can maneuver its way around them. Figure 5.3 shows the occupancy map that was built with the SLAM map output. In simple terms, the occupancy map shows the system which aspects of its path are occupied (the gray areas) and which aspects are free (the white areas).

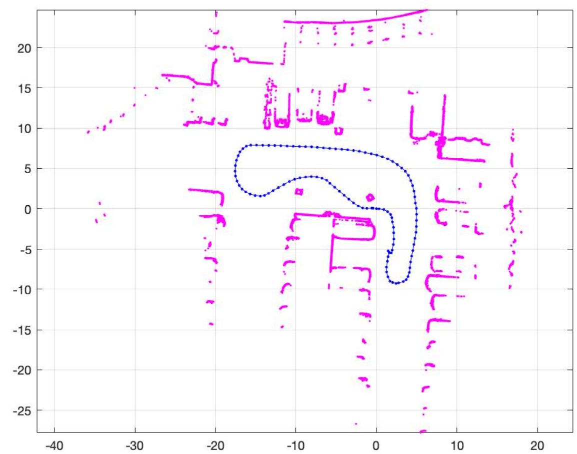


*Figure 5.2: Slam map built using lidar scan data*

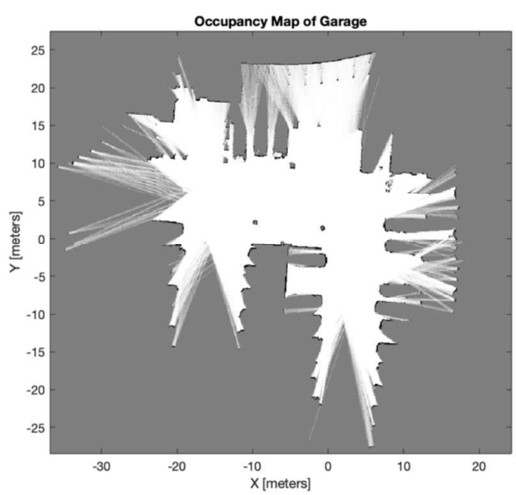


*Figure 5.3: Parking garage occupancy map built from the output of lidar SLAM object*

## 5.2 Limitations

### *5.2.1 Ultrasonic Sensor Obstacle Detection and Avoidance*

The ultrasonic sensing module can only detect whether an obstacle is approaching the user. It cannot detect the type of obstacle or tell the user the precise angle to turn to avoid a collision.

#### *5.2.2 RFID localization*

The received signal strength indicator (RSSI) is a measure that is typically obtained from active tags and not passive tags as used in the project. Thus, the localization algorithms would have benefited from actual RSSI values.

#### *5.2.3 SLAM algorithms*

SLAM algorithms are typically implemented in robot operating systems (ROS) and not with human beings. With a robot, the path can be planned and programmed ahead of time. With a human being, however, this is not quickly done. Also, when a robot needs to turn, the relevant command can be written, and the robot will turn. However, human beings cannot be forced to turn when needed. A feedback system must first communicate this to the user before the turning action is done.

Another limitation is that SLAM algorithms are computationally expensive and do not always provide real-time feedback. However, for the project presented in this paper, real-time detection is essential and could be the difference between life and death.

### 5.3 Future Works

Building on the SLAM algorithms, it would be possible to use the data collected from the ultrasonic sensors as an input to a lidar SLAM model. Since the processing for the SLAM model cannot be done on the constrained ATmega328P-PU microcontroller, the data could be transferred to the user’s phone. Then, the user's phone transmits this data to the cloud, where the relevant processing is done, and the output of this is displayed on the phone in the form of haptic vibration and audio output. In this case, the SLAM algorithm's strengths are incorporated into the system, making it more accurate.

Another area where the project can be extended is retrieving and sending the data to the user's mobile phone for relevant processing and feedback. This would help make the system fully wearable.

Finally, the system could be 3D modeled, making it easier to envision where the technology would be placed.

## 5.4 Conclusion

The project aimed to design a wearable obstacle detection and avoidance system for the visually impaired. This aim has been achieved with the building of the prototype described in the previous chapters. However, the implementation of the project does not work entirely as intended. This is, in part, due to the use of passive RFID tags and the use of a constrained microcontroller. Also, relying on ultrasonic sensors alone does not provide the necessary information about the obstacles for the user. However, the project has proved that it is possible to build an embedded system used as wearable technology for the visually impaired. Continually building upon the framework suggested in this project could potentially result in a low-cost, affordable system for blind and visually impaired persons in Ghana and West Africa.

## References


[1] "Blindness and vision loss," *MedlinePlus Medical Encyclopedia*. Accessed: Apr. 23, 2021. [Online]. Available: https://medlineplus.gov/ency/article/003040.htm.

[2] H. R. Taylor, "Global Blindness: The Progress We Are Making and Still Need to Make," *Asia-Pacific journal of ophthalmology (Philadelphia, Pa.)*, vol. 8, no. 6, Dec. 2019, doi: 10.1097/APO.0000000000000264.

[3] Nicole Cimino-Fiallos, "Hard Hits: Blunt Force Trauma," *Medscape*, May 28, 2020. https://reference.medscape.com/slideshow/blunt-force-trauma-6007991 (accessed Apr. 23, 2021).

[4] R. L. Anderson, W. R. Panje, and C. E. Gross, "Optic Nerve Blindness Following Blunt Forehead Trauma," *Ophthalmology*, vol. 89, no. 5, pp. 445–455, May 1982, doi: 10.1016/S0161-6420(82)34769-7.

[5] A. Frolov, J. Feuerstein, and P. S. Subramanian, "Homonymous Hemianopia and Vision Restoration Therapy," *Neurol Clin*, vol. 35, no. 1, pp. 29–43, Feb. 2017, doi: 10.1016/j.ncl.2016.08.010.

[6] A. K. Schuster, C. Erb, E. M. Hoffmann, T. Dietlein, and N. Pfeiffer, "The Diagnosis and Treatment of Glaucoma," *Dtsch Arztebl Int*, vol. 117, no. 13, pp. 225–234, Mar. 2020, doi: 10.3238/arztebl.2020.0225.

[7] World Health Organization, "Vision impairment and blindness," 2019. https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment (accessed Sep. 21, 2020).

[8] H. G. Krumpaszky, K. Dietz, A. Mickler, and H. K. Selbmann, "Mortality in blind subjects. A population-based study on social security files from Baden-Württemberg.," *Ophthalmologica*, vol. 213, no. 1, pp. 48–53, 1999, doi: 10.1159/000027393.

[9] H. T. Taylor, S. Katala, B. Muñoz, and V. Turner, "Increase in mortality associated with blindness in rural Africa," *Bulletin of the World Health Organization*, vol. 69, no. 3, 1991, Accessed: Apr. 18, 2021. [Online]. Available: https://pubmed.ncbi.nlm.nih.gov/1893509/.

[10] S. Cardin, F. Vexo, and D. Thalmann, "Wearable Obstacle Detection System for visually impaired People," Jan. 2005.

[11] A. Dionisi, E. Sardini, and M. Serpelloni, "Wearable object detection system for the blind," in *2012 IEEE International Instrumentation and Measurement Technology Conference Proceedings*, Graz, Austria, 2012, pp. 1255–1258, doi: https://doi.org/10.1109/I2MTC.2012.6229180.

[12] L. M. Ni, Y. Liu, Y. C. Lau, and A. P. Patil, "LANDMARC: indoor location sensing using active RFID," in *Proceedings of the First IEEE International Conference on Pervasive Computing and Communications, 2003. (PerCom 2003).*, Mar. 2003, pp. 407–415, doi: 10.1109/PERCOM.2003.1192765.

[13] N. Bari, N. Kamble, and P. Tamhankar, "Android Based Object Recognition and Motion Detection to Aid Visually Impaired," *International Journal of Advances in Computer Science and Technology*, vol. 3, no. 10, pp. 462–466, Oct. 2014.

[14] K. Finkenzeller, *RFID Handbook: Fundamentals and Applications in Contactless Smart Cards, Radio Frequency Identification and Near-Field Communication*, 3rd edition. West Sussex, UK: John Wiley & Sons, Ltd., 2010.

[15] R. Sammouda and A. Alrjoub, “Mobile blind navigation system using RFID,” in *2015 Global Summit on Computer Information Technology (GSCIT)*, Jun. 2015, pp. 1–4, doi: 10.1109/GSCIT.2015.7353325.

[16] S. Jia, J. Sheng, D. Chugo, and K. Takase, “Obstacle Recognition for a Mobile Robot in Indoor Environments using RFID and Stereo Vision,” in *2007 International Conference on Mechatronics and Automation*, Aug. 2007, pp. 2789–2794, doi: 10.1109/ICMA.2007.4304001.

[17] M. Nassih, I. Cherradi, Y. Maghous, B. Ouriaghli, and Y. Salih-Alj, “Obstacles Recognition System for the Blind People Using RFID,” in *2012 Sixth International Conference on Next Generation Mobile Applications, Services and Technologies*, Sep. 2012, pp. 60–63, doi: 10.1109/NGMAST.2012.28.

[18] J. Zhou and J. Shi, “RFID localization algorithms and applications—a review,” *Journal of Intelligent Manufacturing*, vol. 20, no. 6, p. 695, Aug. 2008, doi: 10.1007/s10845-008-0158-5.

[19] B. Xu and W. Gang, “Random sampling algorithm in RFID indoor location system,” in *Third IEEE International Workshop on Electronic Design, Test and Applications (DELTA’06)*, Jan. 2006, p. 6 pp. – 176, doi: 10.1109/DELTA.2006.73.