eyeAssist

Team 40

Annamika Dua [ad8]
Sahil Kamesh [skamesh2]
Veda Menon [vedanta2]

ECE 445, Spring 2021

03/04/2021

TA: Xihang Wu

Design Document

Table of Contents

1. Introduction

1.1. Problem and Solution Overview

People with visual impairments represent a unique demographic who face challenges that may not be widely understood by the general population. According to the World Health Organization, there are an estimated 285 million people across the globe who are visually impaired. Furthermore, out of those 285 million, around 39 million are completely blind [1].

Mobility is an extremely crucial part of our lives; we depend on the ability to effectively navigate through any environment we are in every single day. The visually impaired often have trouble navigating through their environment without some kind of assistance. Reading any kind of text, whether that be a book or an important document, can also be burdensome. They may also have difficulty accessing audiobooks online, as those books must be pre-recorded before being sold to the public. This can be extremely frustrating and serve as a significant limitation in their lives. Developing tools to cater to these difficulties, especially through technology, can provide this demographic with a new range of abilities that may have been previously inaccessible.

Our solution is wearable multi-purpose “smart” glasses. These glasses aim to tackle both issues of mobility and reading, enhancing the quality of life of those with visual impairments. This product would allow the visually impaired to navigate their home with ease as well as give them the ability to read text in real-time.

Through the use of ultrasonic sensors mounted on the glasses, any obstacles close to the user within a certain field of view will be detected, and the user will be provided feedback regarding the direction and distance of the obstacles. This design improves upon traditional navigation canes by eliminating the need for constant physical effort on the user’s part. The glasses will also have a built-in camera that can detect and capture any text the user is looking at using OCR, and read it aloud to them with a text-to-speech converter.

1.2. Visual Aid

Figure 1: Visual aid for obstacle detection capabilities - Objects ahead of the user are reported while they are in motion

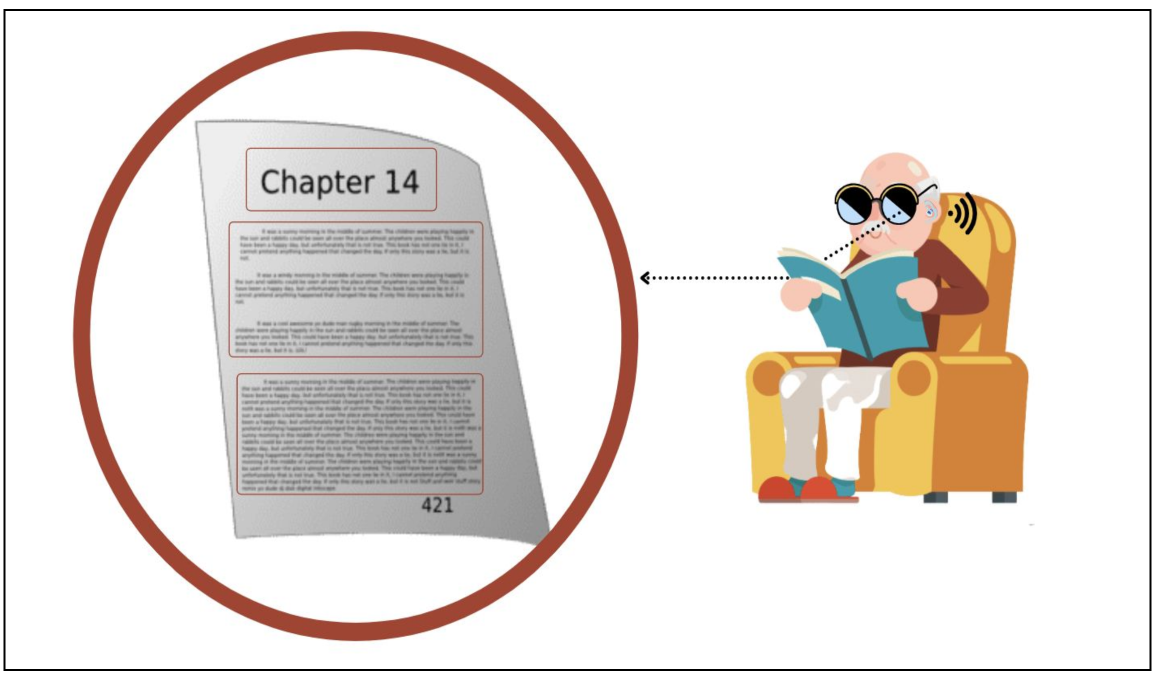

Figure 2: Visual aid for reading capabilities - A snapshot of text is converted to audio in real time for the user

1.3. High-Level Requirements

- The device should be able to detect and read an image of clear, unobstructed text within 30 cm (1 foot) of the user with an accuracy of at least 85%, within 3 minutes.
- The device should be able to detect obstacles at least 1.5 feet high in front of a mobile user at a distance of 1 meter and a field of view of 60 degrees, within an error range of 2 inches.
- The device should be able to operate without recharging for at least 4 hours.

2. Design

2.1. Block Diagram

Our design consists of four main modules: the power module, reading module, obstacle detection module, and a user interface module. The power module ensures the steady operation of the design, and will power all of the subsystems, meeting the high-level requirement of operation without recharging for up to four hours. The reading module will ensure the glasses can capture text with the camera, accurately extract it with computer vision, and perform the text-to-speech conversion, meeting the high-level requirement of reading text with an 85% accuracy. The obstacle detection module contains the accelerometer, which is used to detect whether the user is in motion, and activates the timing logic to send periodic signals to the ultrasonic sensors. The sensors then collect obstacle data ahead of the user and send it to the microcontroller, which will accordingly communicate with the audio system. The user interface module contains the push button, which is used by the user to activate the reading module, and the audio system, which is used to notify the user of the presence of any obstacles. All of this can further be seen below in Figure 3.

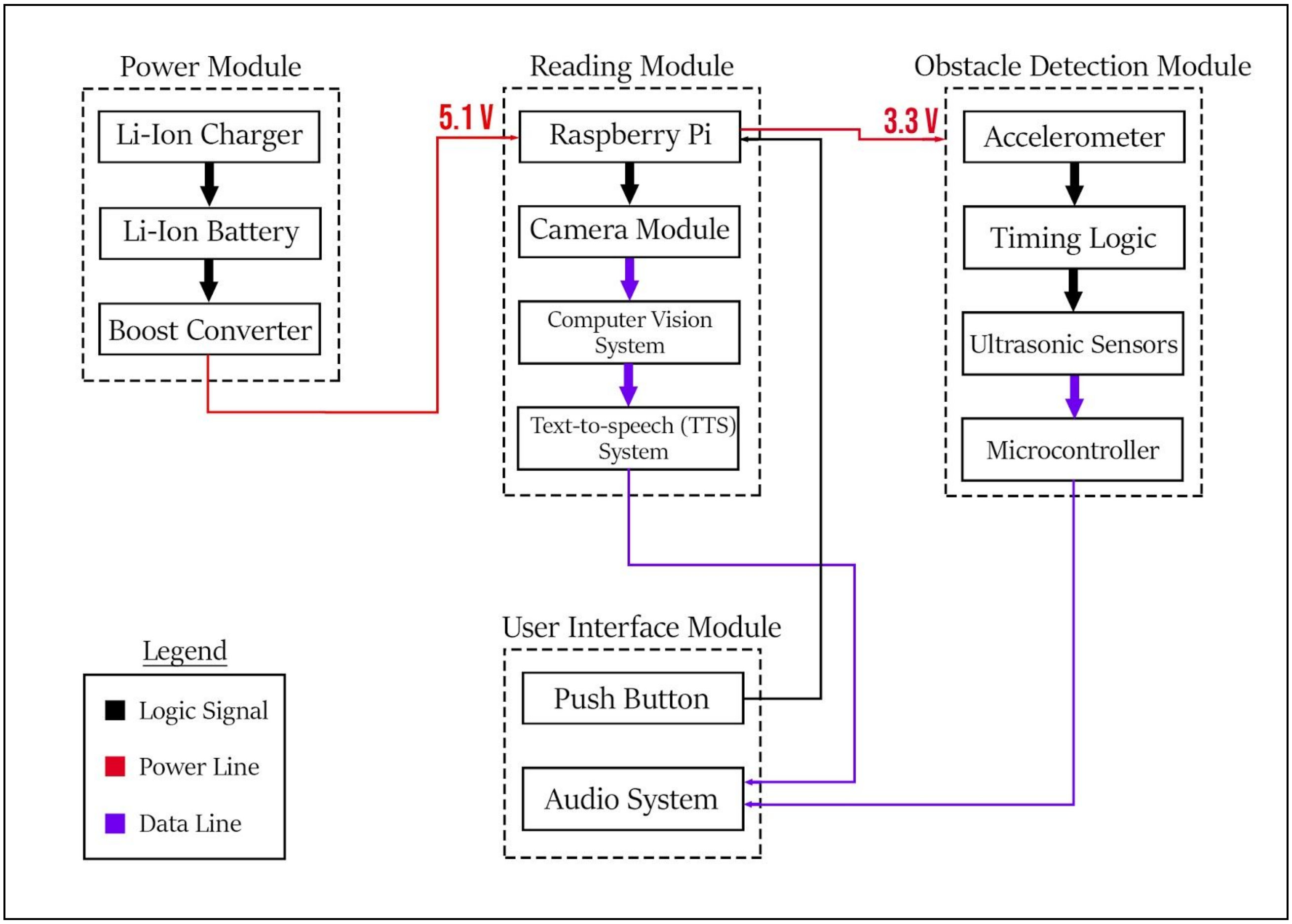

Figure 3: System block diagram

2.2. Physical Design

The project consists of several modules attached to a pair of glasses as shown in Figure 4. The first module is the obstacle detection module, which can be seen mounted on top of the glasses. It consists of some electronics, and two pairs of ultrasonic sensors that are visible in the physical design. The second module is the reading module, characterized by the camera mounted on one side of the glasses frame. All the other components that are needed for operation; the Raspberry Pi, the battery and the power booster, are placed in the box attached to the glasses as seen in Figure 4.

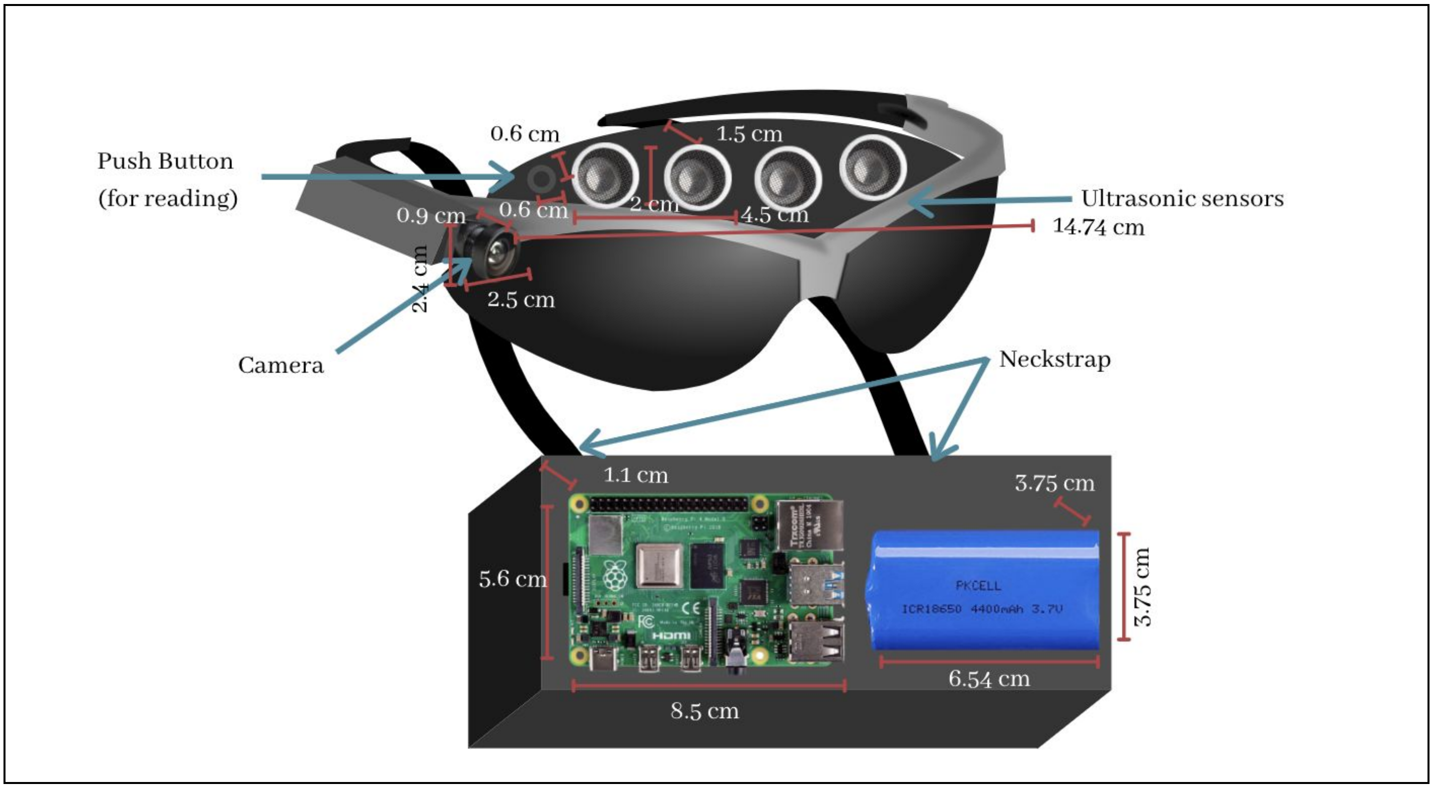

Figure 4: Physical Design of eyeAssist Glasses

2.3. Subsystems

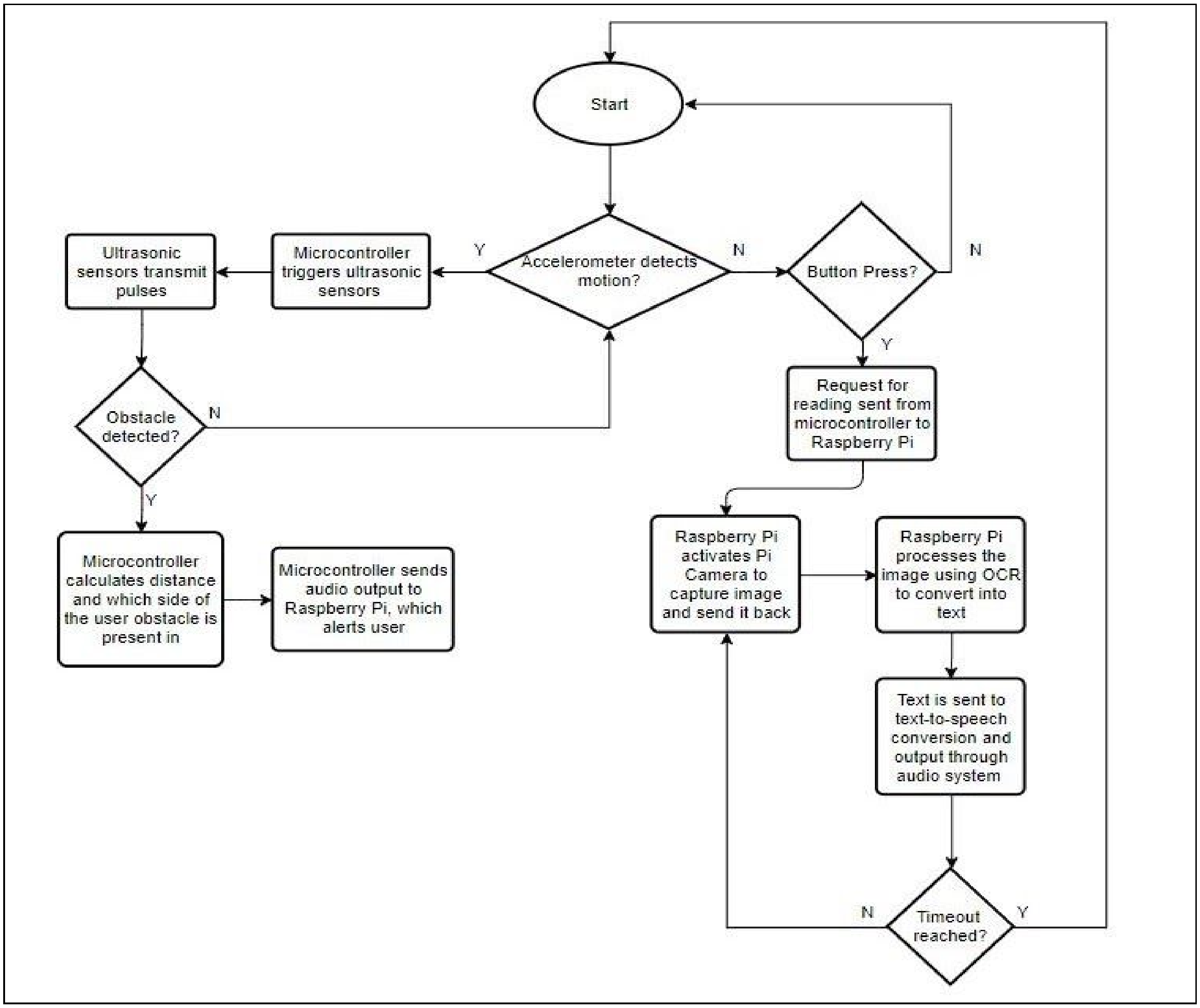

Figure 5: Flowchart of the design, depicting the device operation logic

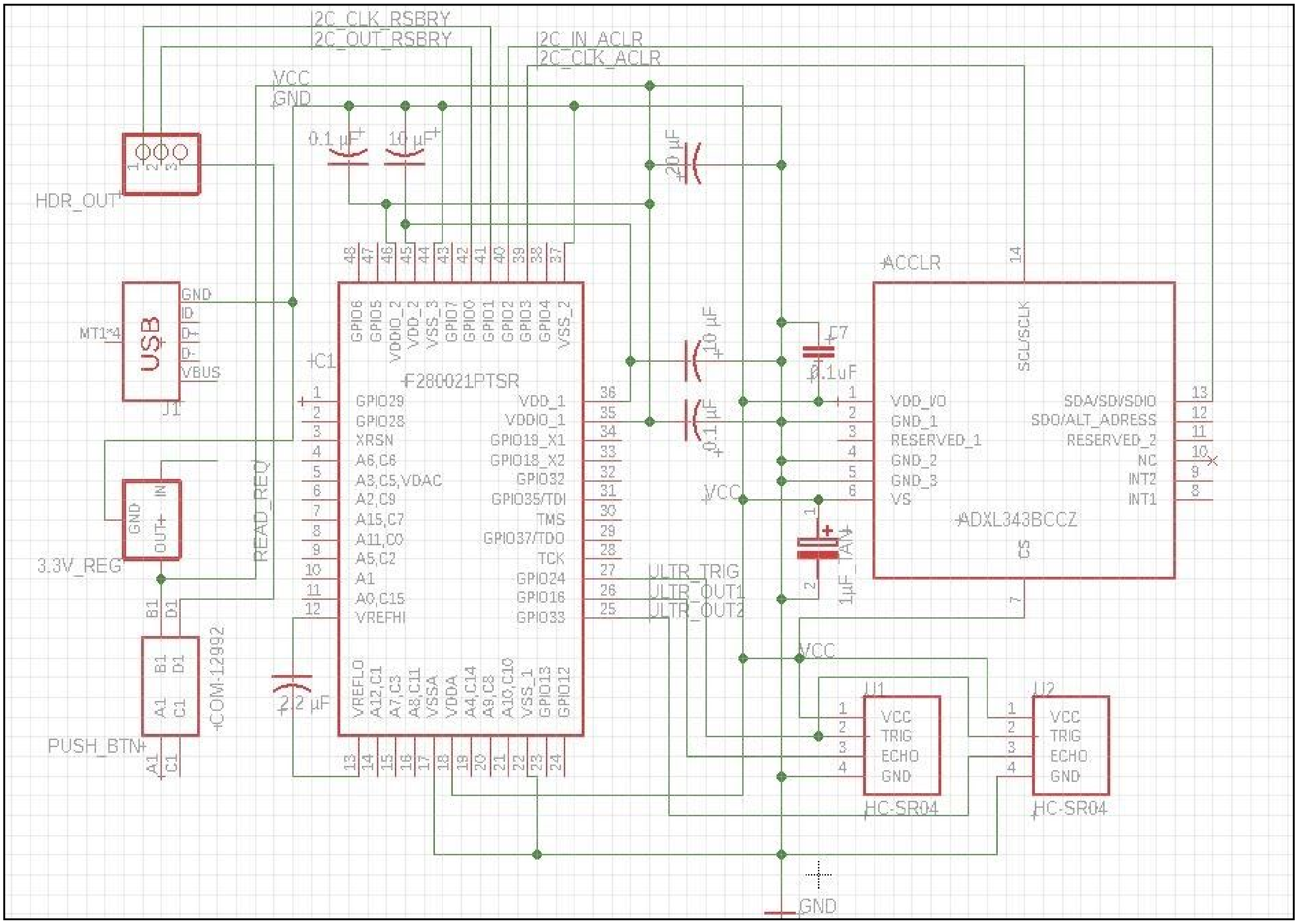

Figure 6: Circuit schematic of the obstacle detection module

2.3.1. Power Module

This module will allow for consistent and safe power distribution throughout the components as well as within the system. Specifically, it is used to power the Raspberry Pi and the ultrasonic module. To achieve this, we use a 3.7V Lithium Ion battery in order to power the Raspberry Pi, however we will also need a power booster in order to get the voltage upto the required 5.2V. We will then use the USB port on the Raspberry Pi to power the PCB, and with the required 3.3V. This module will also ensure that the product can operate for up to four hours without recharging, as described in the high-level requirements.

R&V table for Power Module

Requirements	Verification
Power Module 1. Design must operate for 4+ hours without charging.	**Power Module** 1. Verification process for Item 1: a. Fully charge the battery and connect it to the Raspberry Pi through the boost converter. b. Connect the USB output of the Raspberry Pi to the power input of the PCB. c. Start the OCR computation and the obstacle detection module, and record the time in the lab notebook. d. Wait for the battery to die out and record the time again. e. Compare the maximum duration to the requirement and present the result as a numeric value in the final report.
3.7V DC Li-Ion Battery 1. Battery must output voltage in the range of 3.7V +/- 0.2V at 1A.	**3.7V DC Li-Ion Battery** 1. Verification process for Item 1: a. Fully charge the battery with the Li-ion charger. a. Test the output of the battery using an oscilloscope. b. Record the voltage obtained in the lab notebook. c. Compare the recorded voltage to the required range and present it as a numerical value in the final report.
Raspberry Pi 1. Raspberry Pi must output 3.3V +/- 0.2 V% to the obstacle detection module for a current load up to 100mA.	**Raspberry Pi** 1. Verification process for Item 1: a. Fully charge the battery and connect it to the Raspberry Pi through the boost converter. b. Connect the USB output of the Raspberry Pi to the power input of the PCB. c. Use a voltmeter to test the

	output of the voltage regulator of the obstacle detection module. d. Record the voltages obtained by the voltmeter in the lab notebook. e. Compare the recorded voltages to the required ranges and present the results as numerical values in the final report.

2.3.2. Reading Module

This module will be used to detect and process text the user is looking at using OCR, and then perform text-to-speech conversion on it. To do this, we will use the Pi Camera's 8-megapixel sensor to capture high quality images of the text. Then, we will run OpenCV libraries and Google Tesseract computer vision engine to process the image and convert it to text. This text is then used in the text-to-speech conversion component, and is then sent to the user feedback module to be read aloud to the user. These components together will all ensure that the text is read with an 85% accuracy, as laid out in the high-level requirements.

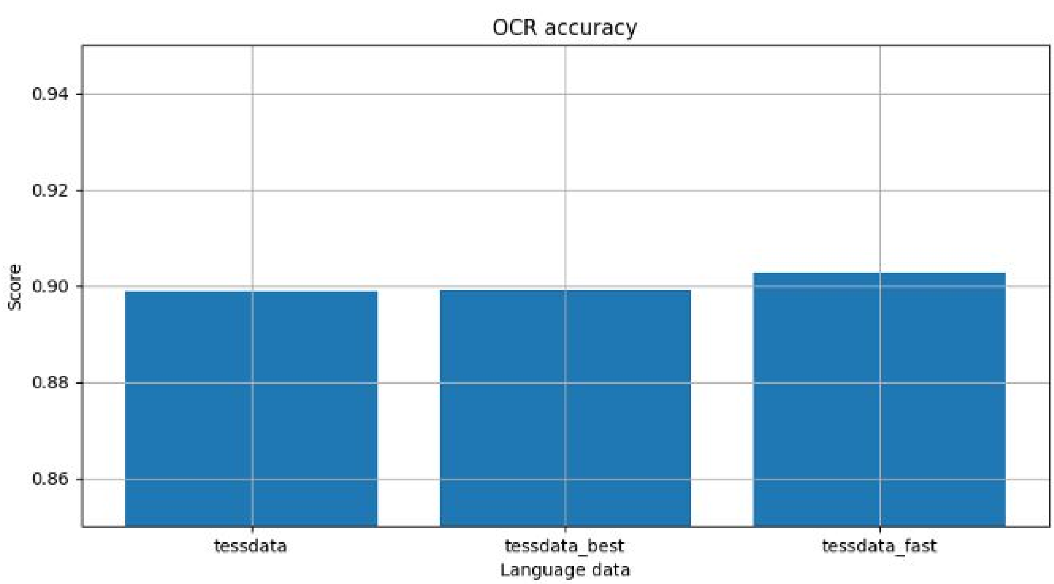

Figure 7: Accuracy of various Tesseract models [7]

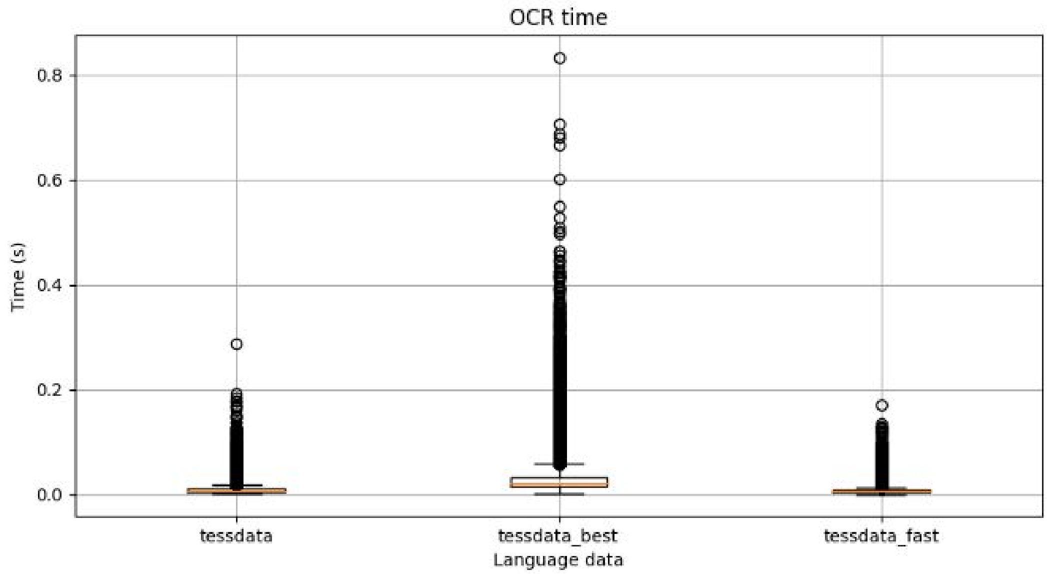

Figure 8: Time consumption for various Tesseract models [7]

R&V table for Reading Module

Requirements	Verification
Reading Module 1. The reading module must detect unobstructed text within a distance of <= 30 cm (1 foot).	**Reading Module** 1. Verification process for Item 1: a. Connect the power supply to the reading module. b. Place a page of printed text within 30 cm directly ahead of the camera. c. Plug in earphones into the audio jack of the Raspberry Pi and press the push button to activate the OCR. d. Record in the lab notebook whether the module detected text and output some audio attempting to read it out. e. Present the result as a True/False (boolean) check in the final report.

Reading Module 1. The reading module must read out unobstructed text within a distance of 30 cm (1 foot) with an accuracy of >= 85%.	**Reading Module** 1. Verification process for Item 1: a. Connect the power supply to the reading module. b. Place a page of printed text of 100 words within 30 cm directly ahead of the camera. c. Plug in earphones into the audio jack of the Raspberry Pi and press the push button to activate the OCR. d. Record in the lab notebook the original text, and the text read out by the algorithm. e. Compare the original and final text by calculating how many words were accurately read out and present the result as a percentage in the final report.
Reading Module 1. The reading module must be able to process a page of text and send the result to the audio system in <= 3 minutes.	**Reading Module** 1. Verification process for Item 1: a. Connect the power supply to the reading module. b. Place a page of a book within 30 cm of the camera. c. Plug in earphones into the audio jack of the Raspberry Pi, record the time in the lab notebook, and press the push button to activate the OCR. d. Wait for the OCR to finish and record the time in the lab notebook again. e. Calculate the time taken for the algorithm to complete and present the result as a numerical value in the final report.

2.3.3. Obstacle Detection Module

This module will be used to determine whether there is an obstacle in the way of the user's path. If the module detects any obstructions, it will also alert the user using voice feedback through the audio system, so that they can safely avoid the obstacle. When the accelerometer detects that the user is in motion by measuring the position, it will alert the microcontroller, which will activate an output pin to generate a high level signal of at least 10 µS. This is connected to the trigger pin of the ultrasonic sensors, which will transmit eight 40 kHz ultrasonic pulses and check if it receives any signals back, which would indicate an obstacle and the distance would then be calculated. This data from the ultrasonic sensors is then sent to the microcontroller, which checks how far any obstacles are from the user and communicates with the audio system accordingly. This module would allow the user to detect obstacles within a 2 meter range and an angle of 30 degrees, within an error range of 5 inches, as specified in the high-level requirement.

Since the HC-SR04 sensor outputs the time it takes for the signal to come back, we will have to calculate the distance ourselves. We can do this by using the following formula [4]:

$$Distance = \frac{Speed * Time}{2} \quad (1)$$

We must divide by 2 because we need to account for the signal travelling to the obstacle and then coming back to the sensor. Further, we will use the speed of sound, which is 0.34 cm/µs, as our speed because ultrasonic waves travel at the speed of sound.

For example, let's say we receive a pulse of 500 µs from the ultrasonic sensors. The we can calculate how far the obstacle is using the following calculation:

$$D = \frac{0.34 cm/\mu s * 500\ \mu s}{2} \quad (2)$$

$$D = 8.5\ cm$$

Thus, we can see that using this method, we are able to calculate the distance of the obstacle from the user.

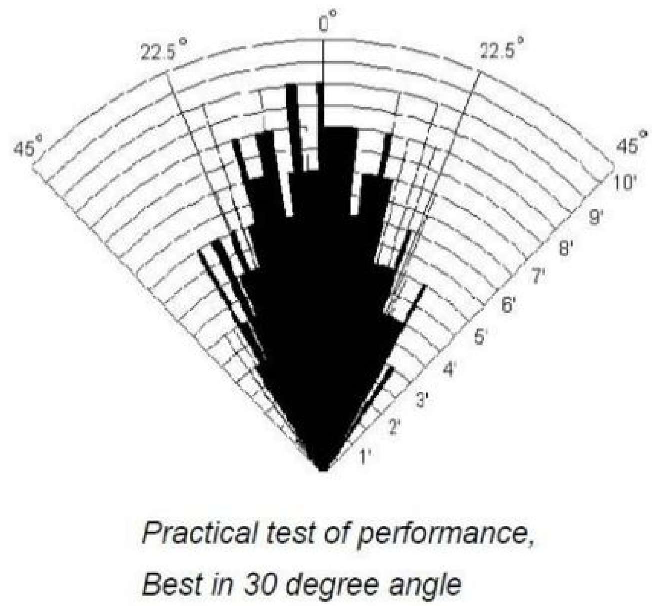

Figure 9: Best performance of the HC-SR04 sensor is within a field of view of 30 degrees. [5]

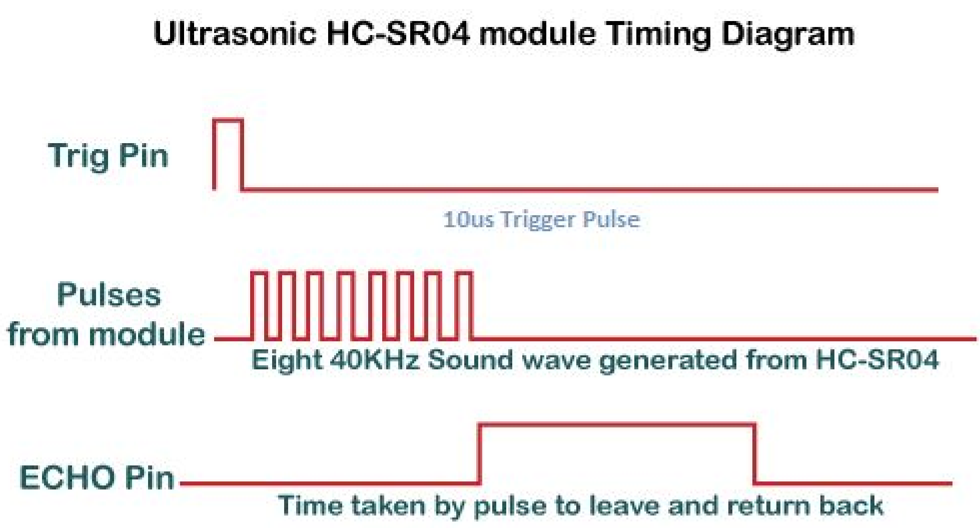

Figure 10: Timing Diagram of HC-SR04 ultrasonic sensor [6]

Furthermore, in order to detect objects at various heights, we plan on angling our sensors downwards at about a 45 degree angle for a better vertical field of view of objects on the ground. However, angling our sensors downwards would result in a false positive detection of obstacles due to the interference of the floor. To prevent our sensors from detecting the floor, we will calibrate our obstacle detection module according to the height of our users as shown in Figure 11.

As an example, we can use average body proportions. We can subtract the distance from the top of the users' head to their eyes as well as a buffer distance of 2 inches from the floor. For a user that is 5'10" (or 70 inches) in height, we can calculate the maximum detected distance using the formula below. This example will be further discussed in the tolerance analysis section.

$$Maximum\ detected\ distance \ = \ h - (h * \tfrac{1}{16}) \ - \ 2 \qquad (3)$$
$$= \ 70 - (70 * \tfrac{1}{16}) \ - \ 2) \ = \ 63.625\ inches$$

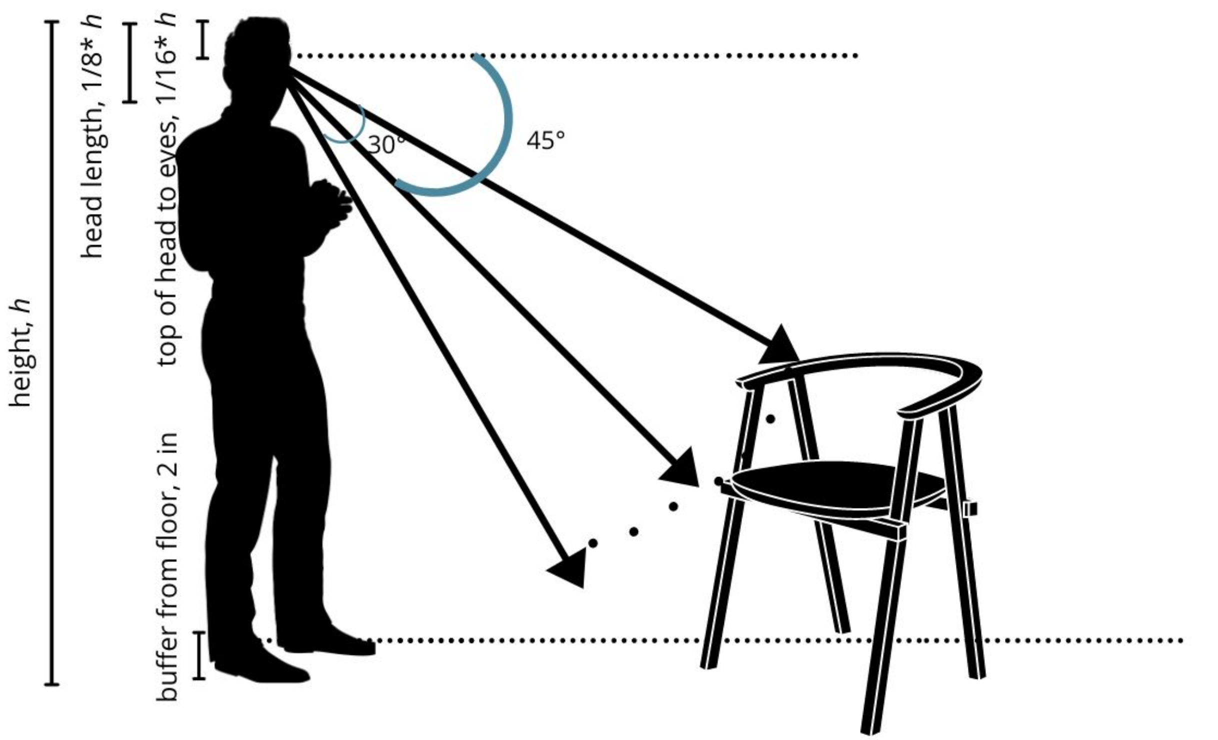

Figure 11: Elimination of floor detection through height calibration

R&V table for Obstacle Detection Module

Requirements	Verification
Obstacle Detection Module 1. Accelerometer must detect if the user is in motion, i.e. if the linear acceleration is greater than 1 m/s^2, and start the obstacle detection process.	**Obstacle Detection Module** 1. Verification Process for Item 1 a. Connect accelerometer to microcontroller and power supply. b. Begin walking with components. c. Measure change in acceleration to check if movement has been detected from a stationary position.
Obstacle Detection Module 1. Ultrasonic sensors should detect objects within <= 2 meters and a combined 50 - 60 degree field of view.	**Obstacle Detection Module** 2. Verification Process for Item 1 a. Use a compass to measure out a 30 degree angle. b. Use measuring tape to measure out 2 meters. c. Place an object at one corner of measured angle and distance. d. Start moving to trigger the obstacle detection module. e. Move the object around the detection range of the sensors and observe the output from the audio system. f. Record the maximum distance and angles that the sensors detected in the lab notebook. g. Present the results in the final report as the maximum range measured.

2.3.4. User Interface Module

This module determines how the user interacts with the physical device. The push button will be used for triggering the text-to-speech feature, and the audio system will be used to activate voice feedback. The audio will be transmitted through the headphone jack on the Raspberry Pi.

R&V table for User Interface Module

Requirements	Verification
User Interface Module 1. The push button must activate the reading module in <= 3s of being pressed.	**User Interface Module** 1. Verification Process for Item 1 a. Program the Raspberry Pi to output a log of the current time as it is starting OCR computation. b. Record the time in the lab notebook as you are pressing the button. c. Listen to the output from the audio jack and record the time output by the Raspberry Pi in the lab notebook. d. Compare the two, record the duration in the lab notebook, and present it as a numerical value in the final report.
User Interface Module 1. Audio outputted from the headphone jack must be clear and audible i.e. >= 30 dB in intensity.	**User Interface Module** 1. Verification Process for Item 1 a. Start the reading module and obstacle detection module. b. Ensure audio feedback from both is loud enough to hear by the user. c. Record the result in the lab notebook and the final report as a boolean (True/False) value.

2.4. Tolerance Analysis

After speaking with our TA, we feel that the aspect of our project that will be critical to its success will be the obstacle detection module. Since mobility is such a critical task, it is important that our device has a reasonably high accuracy, and false positives/negatives are a concern. Unnecessarily alerting the user or even worse not alerting them in case of a harmful obstacle could cause serious injuries.

We further discussed that detecting obstacles of varying heights could pose a problem as well. As previously noted, we will be angling our sensors downwards at around a 45 degree angle to aid the detection of objects of varying heights as well as keeping a buffer distance of 2 inches from the floor to prevent false positives associated with the detection of the floor. Therefore, provided that the user is looking down at an angle of at least 45 degrees, the best case scenario would be that an object over 2 inches in height and is directly below the user can be detected, give or take 2 inches based on the user's height.

Using the maximum detected distance we calculated for a user that is 5'10, and given the angle between the sensors and the ground can calculate the height and distance with which an object would need to be to be detected.

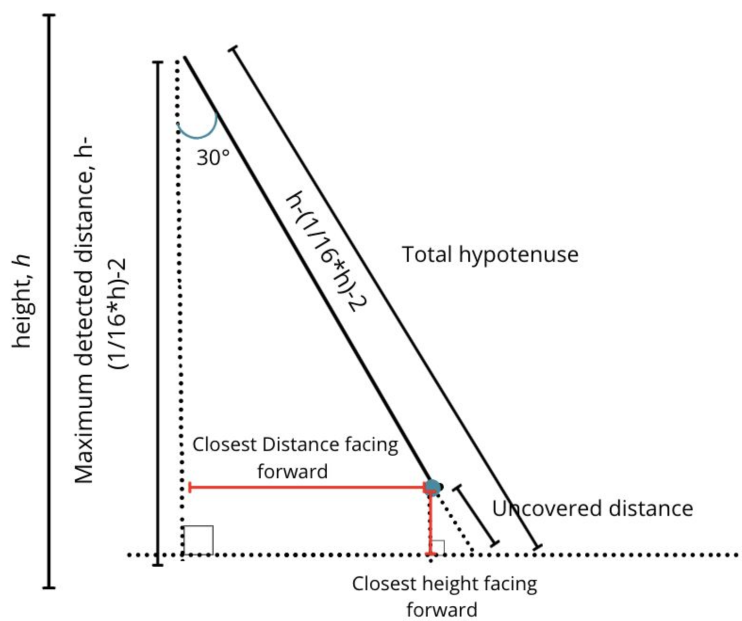

Figure 12: Measurements involving closest detected distance and height of an object while user is facing forward

$$\sin(30°) = \frac{\textit{Closest distance when facing forward}}{\textit{Maximum detected distance}} \quad (1)$$

$$\cos(30°) = \frac{\textit{Maximum detected distance}}{\textit{Total hypotenuse}} \quad (2)$$

Uncovered distance = Total hypotenuse- Maximum detected distance

$$\cos(30°) = \frac{\textit{Closest height when facing forward}}{\textit{Uncovered distance}} \quad (3)$$

Closest height when facing forward (from ground) =
Closest height when facing forward + 2 inch buffer

Therefore, in general terms, if a user is walking straight with their head facing forward, the closest object that can be detected would be around knee level and about 32 inches away.

Given the nature of our sensors, the further away an object is, the taller it would have to be in order to get detected.

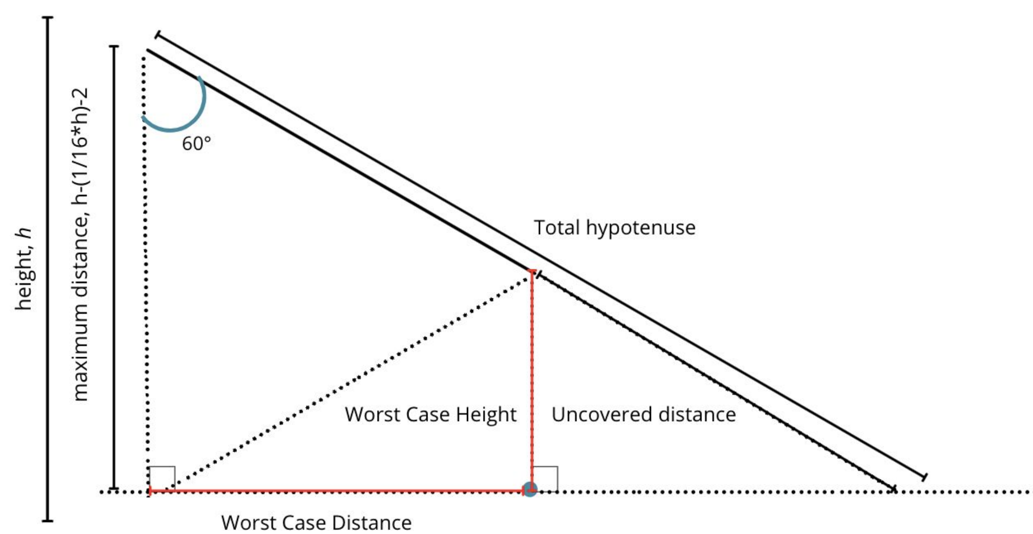

Figure 13: Measurements involving worst case distance and height

$$\cos(60°) = \frac{\textit{Maximum detected distance}}{\textit{Total hypotenuse}}$$

Uncovered distance = Total hypotenuse - *Maximum detected distance*

$$sin(30°) = \frac{Worst\ case\ height}{Uncovered\ distance}$$

$$cos(30°) = \frac{Worst\ case\ distance}{Uncovered\ distance}$$

$$Worst\ case\ height\ from\ ground = Worst\ case\ distance + 2\ inch\ buffer$$

Hence, our worst case scenario entails detecting an object 34 inches in height that is 55 inches away, give or take 8 inches. This issue is however negligible because our sensors will eventually detect the object as the user moves closer. With that in mind, we caution our users that the smaller the object the closer the object would have to be to the user to be detected by our sensors.

To minimize uncovered areas of detection we will instruct our users to scan their surroundings before proceeding, particularly encouraging them to angle their head downwards to detect any objects immediately in front of them. With careful and methodical usage, our uncovered distance remains small but we urge our users with severe visual impairments to use other assistive navigational devices in conjunction with ours.

3. Cost and Schedule

3.1. Cost Analysis

3.1.1. Labor

Total = $45,000

Name	Hourly Rate	Hours Invested	Total
Annamika	$40	150	$15,000
Sahil	$40	150	$15,000
Veda	$40	150	$15,000

3.1.2. Parts

Total = $143.79

Name	Description	Manufacturer	Part #	Quantity	Cost
Li-ion Charger	Needed to charge the lithium ion battery	Adafruit	MCP73831	1	$5.95
Li-on Battery	Needed to supply power to the device	Adafruit	ICR18650 4400mAh 3.7V	1	$19.95
Boost Converter	Needed to boost the 3.7V battery output to 5.1V	Adafruit	PowerBoost 1000C	1	$19.95
Raspberry Pi 4 B, 4GB	Needed to run the OCR system and audio output	Raspberry Pi	Raspberry Pi Model 4 B	1	$55.00
Pi Camera	Needed to capture images for text-to-speech	Raspberry Pi	Pi Camera	1	$24.99

	conversion				
Ultrasonic Sensors	Needed to detect the user's surroundings	Sparkfun	HC-SR04	2	$3.95
Microcontroller	Needed to process the sensor's outputs	Mouser Electronics	F280021PTSR	1	$3.55
Accelerometer	Needed to determine if the user is in motion	Adafruit	ADXL343	1	$5.95
Push Button	Needed to determine when to begin text-to-speech	Sparkfun	COM-12992	1	$0.55

3.1.3. **Total Sum:** $45,000 + $143.79 = $45,143.79

3.2. Schedule

Week	Annamika	Sahil	Veda
3/1	• Finalize design document • Prepare for design review	• Order parts • Finalize design document • Prepare for design review	• Make physical design mockup • Finalize design document • Prepare for design review
3/8	• Design PCB for obstacle detection module	• Design PCB for obstacle detection module	• Finalize physical design/components for machine shop
3/15	• Put in PCB order for first round of PCBway • Begin testing Raspberry Pi functionality	• Assemble reading module (Raspberry Pi and Pi Camera) • Begin testing Raspberry Pi functionality	• Review PCB design • Begin testing Raspberry Pi functionality
3/22	• Continue working with Raspberry Pi, Tesseract, OpenCV • Begin testing text-to-speech general functionality	• Start assembly of obstacle detection module (PCB)	• Start assembly of obstacle detection module (PCB) • Test PCB design and adjust if needed
3/29	• Finalize and correct any errors with reading module	• Start testing obstacle detection functionality • Refine reading module	• Finalize and correct any errors with reading module

		prototype	
4/5	• Integrate modules to reach a cohesive design	• Integrate modules to reach a cohesive design	• Integrate modules to reach a cohesive design
4/12	• Ensure both modules are combined seamlessly • Thoroughly test final design for any errors or missing requirements	• Refine prototype of design • Thoroughly test final design for any errors or missing requirements	• Update documentation on final design • Thoroughly test final design for any errors or missing requirements
4/19	• Mock demo	• Mock demo	• Mock demo
4/26	• Final demo • Work on final paper	• Final demo • Work on final paper	• Final demo • Work on final paper

4. Discussion of Ethics and Safety

The ethics behind our product design and the safe usage of our final product are of the utmost importance to us. We are determined to uphold and apply IEEE's Code of Ethics and address all possible ethical or safety concerns. As outlined in IEEE's fifth Code of Ethics, it is our responsibility to be transparent and realistic in terms of the capabilities of our product [3]. Especially since our product is used as an assistive device for the visually impaired, it becomes crucial that we conduct rigorous testing and provide complete disclosure of the accuracy of our product to ensure the safety of our consumers.

IEEE's first Code of Ethics emphasizes the paramount importance of the safety, health, and welfare of the public [3].To eliminate potential hazards resulting from close contact with electrical components, we plan to enclose our hardware components and power supply in a separate compartment that attaches to the user's glasses with a strap. This isolates the potentially dangerous components like the power supply away from the user's body, increasing their safety. The use of lithium batteries in our product also poses risks such as overheating. Our batteries will have a protection circuitry that prevents overcharging keeping the voltage from going too high as well as too low so the battery will cut-out entirely at 3.0 V.

Having a camera continuously running certainly constitutes privacy concerns so it is important to us that our users have complete control over when their camera is turned on. We intend to use a push button to enable reading capabilities which would therein trigger the camera module. We affirm to protect the privacy of our users and assure to collect no other data besides text related data through our camera.

Ultimately, we strive to provide the best possible solution for our consumers. Seeking criticism as well as acknowledging and correcting errors as noted in IEEE's fifth Code of Ethics, would allow us to continuously improve upon our product [3].

References

[1] World Health Organization, Global Data on Visual Impairment, [Online], Available: https://www.who.int/blindness/publications/globaldata/en/

[2] Sparkfun, HC-SR04 Ultrasonic Sensor Datasheet, [Online], Available: https://cdn.sparkfun.com/datasheets/Sensors/Proximity/HCSR04.pdf

[3] ieee.org, IEEE Code of Ethics, [Online], Available: https://www.ieee.org/about/corporate/governance/p7-8.html

[4] Last Minute Engineers, How HC-SR04 Sensor Works, [Online], Available: https://lastminuteengineers.com/arduino-sr04-ultrasonic-sensor-tutorial/

[5] University of Washington, HC-SR04 Ultrasonic Rangefinder, [Online], Available: https://dxarts.washington.edu/wiki/hc-sr04-ultrasonic-rangefinder

[6] ElectronicWings, Ultrasonic Module HC-SR04, [Online], Available: https://www.electronicwings.com/sensors-modules/ultrasonic-module-hc-sr04

[7] Towards Data Science, Google's Tesseract OCR:, [Online], Available: https://towardsdatascience.com/googles-tesseract-ocr-how-good-is-it-on-documents-d71d4bf7640