

Integrating Object Detection with GPS for Enhanced Mobility Assistance to Visually Impaired Individuals

Arnav Mishra

Massachusetts Academy of Math and Science

December 12, 2021

Abstract

Numerous technological solutions are being created to alert others of the presence of visually impaired people, but there is still a lack of capable assistive technologies to enable individuals with visual impairments to navigate by themselves. Therefore, this project aims to design a solution that allows these individuals to walk outdoors safely and independently through object recognition and path determination. First, the accuracy and precision of the infrared and ultrasonic sensors were tested on aluminum objects placed several distances away. These aluminum objects would have eccentricities of infinity, one, and two to test how the curvature affects the accuracy and precision of the sensors. Then, an algorithm was tested in a virtual environment with four, five, and six blocks randomly placed in a simulated area to create and modify an obstacle avoidance system. Finally, the algorithm was connected to the sensors and tested in the same environment with tangible blocks. The overall accuracy and precision of both sensors decreased as the object's eccentricity increased. The system was remarkably accurate overall for the obstacle avoidance algorithm as the program was successful at least 94.0% of the time and 92.0% when using the sensors. The solution bridges a significant knowledge gap in the community regarding visually impaired technology and aims to help the visually impaired navigate outdoors more comfortably.

Introduction

According to the World Health Organization, more than one billion people are visually impaired due to reasons that could have been prevented. Over the past decade, with no definite cure for blindness, there has been a significant increase in the number of tools available for the visually impaired (Vision Impairment and Blindness, 2020). This includes technology to help them use computers and shop in modern malls. However, one of the most significant difficulties for the visually impaired is traveling outside. Due to the unpredictable environment and the individuals impacted by a visual deficiency, there is a high risk of danger outdoors (Vision Loss - Safety Around the Home, 2014). Therefore, numerous researchers are studying methods to help improve mobility for the blind and are developing technological solutions to assist the visually impaired. Unfortunately, most companies and researchers only focus on either helping the user navigate through an application or detect objects ahead through a technologically advanced cane. With the focus split on the two potential solutions, there is a knowledge gap where almost no solution implements both (Khan, Khusro, & Ullah, 2018). Numerous technological solutions are being created to alert others of individuals with visual impairments and lower the risks in their environment, but there is still a lack of capable assistive technology for the blind to move around themselves. Thus, this project aims to design a solution that allows individuals to walk outdoors safely and independently. By finding potential flaws and successes of what is known today, a more effective solution can be made to help the visually impaired move outdoors more safely and independently. Different modules and sensors can then be used to create a working prototype that can accomplish a series of tasks that simulate real-world scenarios.

Section 2: Materials and Methodology

Section 2.1: Materials for Testing

For this project, there is a list of materials needed for testing, including specific software. The Arduino MEGA 2560 Board was the central processor that connected all the sensors and modules to the IDE (Integrated Development Environment). The primary sensors used were the ultrasonic and infrared sensors. The sensors were then mounted onto a pre-bought cane. A Bluetooth module was used to connect to the application to run. While the data was collected on a computer, the final product can be downloadable on an Android tablet with features to alert the user of where to go. A vibration module was also used as an emergency signaler in case something unexpected approached the user (See Appendix H).

For the experimental testing set up, aluminum objects of different eccentricities and blocks of aluminum for the real-life obstacle collision avoidance testing were used. The aluminum blocks used had the same height and width of 20 by 20 cm; however, their length varied based on their eccentricity. To measure the eccentricity of the blocks, dimensions and sketches were given to professionals who were able to cut them to the necessary curvature. For eccentricities of one and two, the length was 10 cm and 15 cm, respectively. Six aluminum blocks were used for the obstacle avoidance collision system, and they all had dimensions of 20 by 20 by 1 cm. Other than these materials, a protractor and measuring tape were used to set up all trials (See Appendix K).

Section 2.2: Software for Testing

Several programming platforms were used to make the final product. All the programming for the Arduino board and sensors were done on the Arduino IDE. Once data was compiled onto the Arduino MEGA board, the information was sent via Bluetooth module to the App Inventor platform. It is exactly like the MIT App Inventor Platform, except it allows the coding to be done in JAVA. The abridged version also included libraries and data previously compiled on Eclipse, which can be imported and put in the App Inventor Platform. The Eclipse platform will house most of the algorithms with the obstacle avoidance system. Finally, Gazebo was used to create real-world scenarios to test my object-avoidance algorithm (Home Automation with MIT App inventor and ESP8266, 2019).

Section 2.3: Testing Eccentricities of Object

For testing the effect that the eccentricity of an object had on the sensor's ability to detect an object, objects of varying eccentricities were used. Since these curved objects can be concave or convex in relation to the sensor, both sides of the object were tested. The objects were placed at 50 cm, 75 cm, 100 cm, 200 cm, and 300 cm away from the sensors. The sensors were individually tested on one aluminum object at a time. All the testing occurred on a flat tabletop using a protractor and measuring tape to make sure the sensor and object were the correct distance away from each other and were perpendicular to the surface. Since the straight-edge aluminum object (eccentricity of infinity) has no curve, it was only tested once with each sensor. The sensors were connected to the Arduino MEGA board, which then uploaded data into a console on the

Arduino IDE (the board was connected through a Bluetooth module). Programming was then done to calculate the accurate distance the sensors should collect and display. This was done by calculating the time it took for the light or ultrasonic wave to get to the object and back and calculating it into expected distances. The console displayed centimeter values rounded to the nearest centimeter, and these results were recorded. Ten trials were done for each distance, sensor, and eccentricity combination.

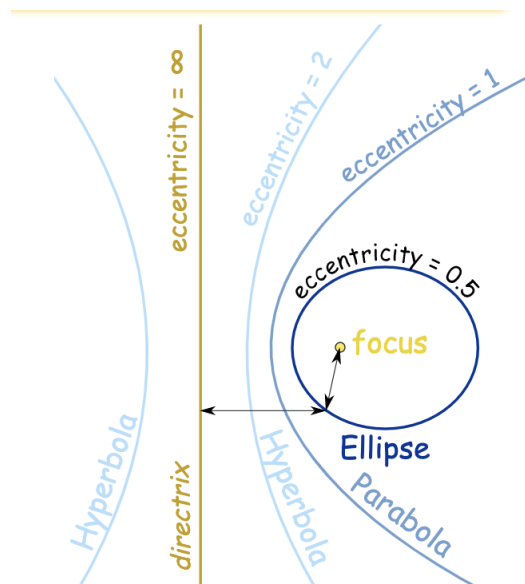


Figure 1: Eccentricities of different graphs. This figure depicts how the curvature or eccentricity of an object can be calculated to have a value from 0.5 to infinity

Taisbak, C. M. (2021). Conic Section. Retrieved March 07, 2021, from

<https://www.britannica.com/science/conic-section#ref70765>

Section 2.4: Testing Obstacle Avoidance System

A virtual environment was initially set up to demonstrate the concepts of an obstacle avoidance system (See Appendix I), which then could be applied to a real-world example. This was done in the Gazebo platform, where four, five, or six objects on 20 by 20 units were placed in a 500 by 500 units environment. Trials of 50 were done on each of the number of objects used. The test was deemed successful if no object was touched and the person got to the other side of the simulation. This simulation programming was then imported as a C++ file and added to the Arduino IDE. Now, the sensors were used with this algorithm to direct the user in real-life to avoid obstacles. There were certain values stated by the Arduino, such as "left" or "right," to direct the user. These values were then exported to the App Inventor platform to alert the user of objects nearby. The same simulation was done in real life with an environment of 5000 by 5000 cm, and masking tape was used to mark 200 by 200 cm squares. A trial was deemed successful if the person using both the cane with sensors and application could get to the other side without touching the masking tape. There was no bias in this testing as a proctor alerted the user if they touched the tape, and the participant was blindfolded. There were 50 trials done with six objects around.

Section 2.5: Statistical Tests Used

The statistical tests used were standard deviation, mean, T-tests, and p-scores. The mean shows the accuracy of the sensors, which can be taken to make conclusions on the data and compare results. The standard deviation is used to show the precision of the data. For valid conclusions to be taken from the data, it needs to be reliable. The

standard deviation shows how reliable the data is as if there is a low deviation, the results were not happening by chance. Instead, there had to be some reasoning for those results, and the mean will help find that reasoning. T-tests were also used to identify a significant statistical change between the data sets. These tests were then converted into p-values to determine whether the curvature of an object influences the accuracy of the sensors.

Section 3: Results

Section 3.1: Results from Eccentricity Testing

Table 1: Infrared Sensor Testing on Aluminum Object of Eccentricity of One Facing Toward Sensor

Infrared Sensor Testing on Aluminum Object of Eccentricity 1 Facing Inward													
uncertainty=±0.1 cm													
Actual Distance to Sensor(cm)	T1 (cm)	T2 (cm)	T3 (cm)	T4 (cm)	T5 (cm)	T6 (cm)	T7 (cm)	T8 (cm)	T9 (cm)	T10 (cm)	Mean (cm)	Standard Deviation (cm)	Percent Difference of Mean to Actual Distance
50	51	51	50	53	52	52	52	51	51	48	51.1	1.37	-2.20%
75	76	76	76	76	78	78	74	74	74	74	75.6	1.58	-0.80%
100	103	105	104	104	104	105	105	103	103	102	103.8	1.03	-3.80%
200	203	202	201	202	202	202	202	203	205	204	202.6	1.17	-1.30%
300	305	306	306	306	306	310	305	308	309	311	307.2	2.15	-2.40%

In Table 1, all standard deviations were lower than three cm, showing that the data is highly precise even at further distances such as 300 cm. The percent differences of mean to actual distances are all less than 4% away. As the distance from the sensor to the object increases, however, there is a general increase in the mean deviation from the actual distance and a slight decrease in the precision of the data.

*Table 2: Ultrasonic Sensor Testing on Aluminum Object of Eccentricity of One Facing
Toward Sensor*

Ultrasonic Sensor Testing on Aluminum Object of Eccentricity 1 Facing Inward													
uncertainty=±0.1 cm													
Actual Distance to Sensor(cm)	T1 (cm)	T2 (cm)	T3 (cm)	T4 (cm)	T5 (cm)	T6 (cm)	T7 (cm)	T8 (cm)	T9 (cm)	T10 (cm)	Mean (cm)	Standard Deviation (cm)	Percent Difference of Mean to Actual Distance
50	51	52	53	53	53	53	53	55	52	52	52.7	1.06	-5.40%
75	77	78	79	80	80	80	85	80	73	75	78.7	3.27	-4.93%
100	101	96	97	98	96	96	96	93	94	95	96.2	2.20	3.80%
200	205	206	206	208	210	214	212	214	213	212	210	3.50	-5.00%
300	310	303	314	315	316	320	320	319	307	308	313.2	5.94	-4.40%

In Table 2, all standard deviations were lower than 6 cm, showing that the data is highly precise even at further distances such as 300 cm. The percent differences of mean to actual distances are all less than 5% away. As the distance from the sensor to the object increases, however, there is a general increase in the mean deviation from the actual distance and a slight decrease in the precision of the data.

*Table 3: Infrared Sensor Testing on Aluminum Object of Eccentricity of One Facing
Away From Sensor*

Infrared Sensor Testing on Aluminum Object of Eccentricity 1 Facing Outward													
uncertainty=±0.1 cm													
Actual Distance to Sensor(cm)	T1 (cm)	T2 (cm)	T3 (cm)	T4 (cm)	T5 (cm)	T6 (cm)	T7 (cm)	T8 (cm)	T9 (cm)	T10 (cm)	Mean (cm)	Standard Deviation (cm)	Percent Difference of Mean to Actual Distance
50	52	49	49	49	49	52	52	52	52	53	50.9	1.66	-1.80%
75	73	73	73	73	72	71	71	71	71	71	71.9	0.99	4.13%
100	101	103	104	102	102	104	103	103	102	105	102.9	1.20	-2.90%
200	205	206	206	206	206	205	204	203	203	203	204.7	1.34	-2.35%
300	304	304	306	305	306	305	306	306	308	304	305.4	1.26	-1.80%

In Table 3, all standard deviations were lower than two cm, showing that the data is highly precise even at further distances such as 300 cm. The percent differences of mean to actual distances are all less than 5% away. As the distance from the sensor to the object increases, however, there is a general increase in the mean deviation from the actual distance and a slight decrease in the precision of the data.

Table 4: Ultrasonic Sensor Testing on Aluminum Object of Eccentricity of One Facing Away From Sensor

Ultrasonic Sensor Testing on Aluminum Object of Eccentricity 1 Facing Outward														
uncertainty=±0.1 cm														
Actual Distance to Sensor(cm)	T1 (cm)	T2 (cm)	T3 (cm)	T4 (cm)	T5 (cm)	T6 (cm)	T7 (cm)	T8 (cm)	T9 (cm)	T10 (cm)	Mean (cm)	Standard Deviation (cm)	Percent Difference of Mean to Actual Distance	
50	54	54	53	54	53	52	52	52	52	53	52.9	0.88	-5.80%	
75	72	71	73	74	74	72	71	75	73	76	72.5	1.58	3.33%	
100	94	93	92	92	92	96	97	92	93	93	93.4	1.78	6.60%	
200	204	206	208	212	214	214	212	213	216	213	211.2	3.88	-5.60%	
300	311	315	317	318	318	312	314	314	317	318	315.4	2.59	-5.13%	

In Table 4, all standard deviations were lower than four cm, showing that the data is precise even at further distances such as 300 cm. The percent differences of mean to actual distances are all less than 7% away. As the distance from the sensor to the object increases, however, there is a general increase in the mean deviation from the actual distance and a slight decrease in the precision of the data. Compared to the other data, there is a significant increase in the percent mean difference in this testing batch. The standard deviation stays relatively constant, though.

Table 5: Infrared Sensor Testing on Aluminum Object of Eccentricity of Two Facing Toward Sensor

Infrared Sensor Testing on Aluminum Object of Eccentricity 2 Facing Inward														
uncertainty=±0.1 cm														
Actual Distance to Sensor(cm)	T1 (cm)	T2 (cm)	T3 (cm)	T4 (cm)	T5 (cm)	T6 (cm)	T7 (cm)	T8 (cm)	T9 (cm)	T10 (cm)	Mean (cm)	Standard Deviation (cm)	Percent Difference of Mean to Actual Distance	
50	52	51	53	53	53	50	51	51	51	50	51.5	1.18	-3.00%	
75	76	77	78	75	75	74	73	72	72	72	74.4	2.17	0.80%	
100	103	104	104	105	106	108	104	107	107	102	105	1.94	-5.00%	
200	193	192	191	190	189	193	192	190	185	184	189.9	3.14	5.05%	
300	292	289	291	291	285	286	287	288	289	285	288.3	2.54	3.90%	

In Table 5, all standard deviations were lower than four cm, showing that the data is precise even at further distances such as 300 cm. The percent differences of mean to actual distances are all less than 6% away. As the distance from the sensor to the object increases, however, there is a general increase in the mean deviation from the actual distance and a slight decrease in the precision of the data. Compared to the

other data, there is a significant increase in the percent mean difference in this testing batch. The standard deviation stays relatively constant, though.

Table 6: Ultrasonic Sensor Testing of Aluminum Object of Eccentricity Two Facing Toward Sensor

Ultrasonic Sensor Testing on Aluminum Object of Eccentricity 2 Facing Inward													
uncertainty=±0.1 cm													
Actual Distance to Sensor(cm)	T1 (cm)	T2 (cm)	T3 (cm)	T4 (cm)	T5 (cm)	T6 (cm)	T7 (cm)	T8 (cm)	T9 (cm)	T10 (cm)	Mean (cm)	Standard Deviation (cm)	Percent Difference of Mean to Actual Distance
50	49	51	51	51	48	49	50	50	51	51	50.1	1.10	-0.20%
75	73	73	72	72	71	71	72	73	74	73	72.4	0.97	3.47%
100	102	103	103	102	103	110	111	112	115	113	107.4	5.23	-7.40%
200	194	193	192	192	195	193	192	189	189	190	191.9	2.02	4.05%
300	289	286	285	283	285	287	288	289	287	290	286.9	2.18	4.37%

In Table 6, all standard deviations were lower than 6 cm, showing that the data is precise even at further distances such as 300 cm. The percent differences of mean to actual distances are all less than 8% away. As the distance from the sensor to the object increases, however, there is a general increase in the mean deviation from the actual distance and a slight decrease in the precision of the data. Compared to the other data, there is a significant increase in the percent mean difference in this testing batch. The standard deviation stays relatively constant, though.

Table 7: Infrared Sensor Testing of Aluminum Object of Eccentricity Two Facing Away from Sensor

Infrared Sensor Testing on Aluminum Object of Eccentricity 2 Facing Outward													
uncertainty=±0.1 cm													
Actual Distance to Sensor(cm)	T1 (cm)	T2 (cm)	T3 (cm)	T4 (cm)	T5 (cm)	T6 (cm)	T7 (cm)	T8 (cm)	T9 (cm)	T10 (cm)	Mean (cm)	Standard Deviation (cm)	Percent Difference of Mean to Actual Distance
50	56	55	57	50	50	56	56	58	54	54	54.6	2.72	-9.20%
75	77	78	79	79	74	79	76	77	77	79	77.5	1.65	-3.33%
100	104	104	103	105	108	109	104	109	110	112	106.8	3.16	-6.80%
200	202	200	190	197	200	188	189	190	189	188	193.3	5.72	3.35%
300	291	294	292	292	293	290	292	286	291	290	291.1	2.18	2.97%

In Table 7, all standard deviations were lower than 6 cm, showing that the data is precise even at further distances such as 300 cm. The percent differences of mean to

actual distances are all less than 7% away. As the distance from the sensor to the object increases, however, there is a general increase in the mean deviation from the actual distance and a slight decrease in the precision of the data. Compared to the other data, there is a significant increase in the percent mean difference in this testing batch. The standard deviation stays relatively constant, though.

Table 8: Ultrasonic Sensor Testing of Aluminum Object of Eccentricity Two Facing Away from Sensor

Ultrasonic Sensor Testing on Aluminum Object of Eccentricity 2 Facing Outward														
uncertainty= ± 0.1 cm														
Actual Distance to Sensor(cm)	T1 (cm)	T2 (cm)	T3 (cm)	T4 (cm)	T5 (cm)	T6 (cm)	T7 (cm)	T8 (cm)	T9 (cm)	T10 (cm)	Mean (cm)	Standard Deviation (cm)	Percent Difference of Mean to Actual Distance	
50	55	56	54	55	55	60	48	53	52	55	54.3	3.06	-8.60%	
75	70	66	65	63	62	64	68	74	72	73	67.7	4.35	9.73%	
100	110	112	114	106	98	98	104	106	110	108	106.7	5.46	-6.70%	
200	184	183	182	185	187	189	185	190	191	180	185.6	3.60	7.20%	
300	321	323	334	322	321	321	330	328	318	317	323.5	5.44	-7.83%	

In Table 8, all standard deviations were lower than 6 cm, showing that the data is precise even at further distances such as 300 cm. The percent differences of mean to actual distances are all less than 10% away. As the distance from the sensor to the object increases, however, there is a general increase in the mean deviation from the actual distance and a slight decrease in the precision of the data. Compared to the other data, there is a significant increase in the percent mean difference in this testing batch. The standard deviation stays relatively constant, though.

Table 9: Infrared Sensor Testing of Straight Aluminum Object

Infrared Sensor Testing on Aluminum Object of Eccentricity ∞														
uncertainty= ± 0.1 cm														
Actual Distance to Sensor(cm)	T1 (cm)	T2 (cm)	T3 (cm)	T4 (cm)	T5 (cm)	T6 (cm)	T7 (cm)	T8 (cm)	T9 (cm)	T10 (cm)	Mean (cm)	Standard Deviation (cm)	Percent Difference of Mean to Actual Distance	
50	50	50	50	50	50	50	50	50	49	50	49.9	0.32	0.20%	
75	74	75	75	75	75	75	75	74	75	75	74.8	0.42	0.27%	
100	100	100	99	99	100	100	99	100	100	100	99.7	0.48	0.30%	
200	200	199	200	200	200	200	200	200	200	200	199.9	0.32	0.05%	
300	301	301	299	300	300	300	300	301	300	300	300.2	0.63	-0.07%	

In Table 9, all standard deviations were lower than one cm, showing that the data is highly precise even at further distances such as 300 cm. The percent differences of mean to actual distances are all less than 1% away. As the distance from the sensor to the object increases, however, there is a general increase in the mean deviation from the actual distance and a slight decrease in the precision of the data. Compared to the other data, the percent mean difference seems to be the lowest of all testing. This is also the highest precision of all the data, with precision values of 0.32 or 0.48 cm.

Table 10: Ultrasonic Sensor Testing of a Straight-Edged Aluminum Object

Ultrasonic Sensor Testing on Aluminum Object of Eccentricity ∞													
uncertainty = ± 0.1 cm													
Actual Distance to Sensor (cm)	T1 (cm)	T2 (cm)	T3 (cm)	T4 (cm)	T5 (cm)	T6 (cm)	T7 (cm)	T8 (cm)	T9 (cm)	T10 (cm)	Mean (cm)	Standard Deviation (cm)	Percent Difference of Mean to Actual Distance
50	48	49	48	49	50	50	49	50	50	51	49.4	0.97	1.20%
75	76	75	75	75	75	75	74	73	73	73	74.4	1.07	0.80%
100	101	100	98	99	99	99	97	98	98	102	99.1	1.52	0.90%
200	201	200	203	203	197	198	203	202	200	202	200.9	2.13	-0.45%
300	303	301	300	297	304	296	298	298	299	301	299.7	2.58	0.10%

In Table 10, all standard deviations were lower than three cm, showing that the data is highly precise even at further distances such as 300 cm. The percent differences of mean to actual distances are all less than 2% away. As the distance from the sensor to the object increases, however, there is a general increase in the mean deviation from the actual distance and a slight decrease in the precision of the data. Compared to the other data, the percent mean difference seems to be the lowest of all testing. This is also the highest precision of all the data, with precision values of 0.97 or 1.07 cm.

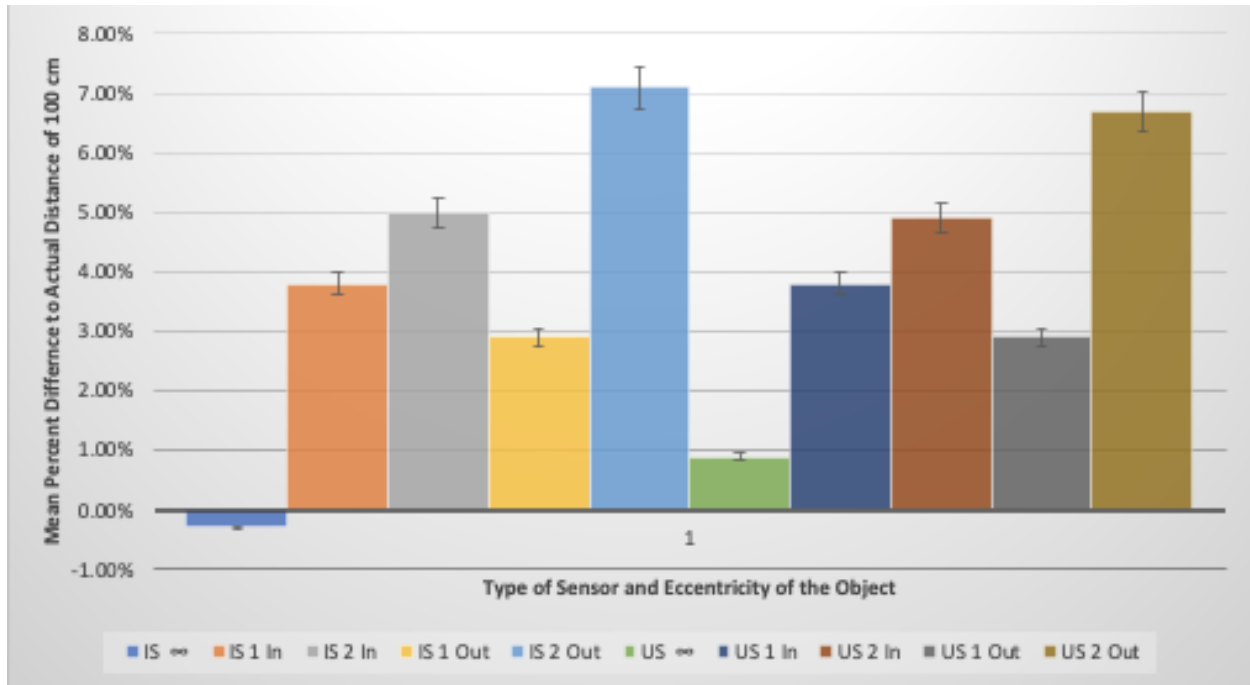


Figure 2: Percent Difference of Mean to Actual Distance of Testing at 100 cm

Here, the percent differences at different testing intervals are shown. Overall, there is an increase in percent mean difference from when the eccentricity increases subjective to both concave and convex shapes. This means that there is a decrease in accuracy as the eccentricity increases. For example, the mean percent difference from *U.S. Two Out* testing was 6.83% compared to 0.89% when the eccentricity was infinity. When the aluminum object was facing away from the sensor, there was a more considerable spike in change for percent mean differences compared to facing inward. The standard deviation overall was low no matter the eccentricity showing precise models. The standard deviations were all below 5%, even when the eccentricity was two. Relatively, the infrared sensor and ultrasonic sensors were giving similar results throughout.

A one-way T-test was done to compare the ultrasonic sensor detection of an object with an eccentricity of infinity and two at 50 cm and 300 cm. At 300 cm, there was a T-value of 11.589, and at 50 cm, there was a T-value of 2.2014. There is a significant difference between both testing values at 300 cm, however a nominal value close to 50 cm. There is also a p-value of less than 0.0001 at 300 cm, showing significance in the testing, whereas for 50 cm, there is a p-value of slightly over 0.05.

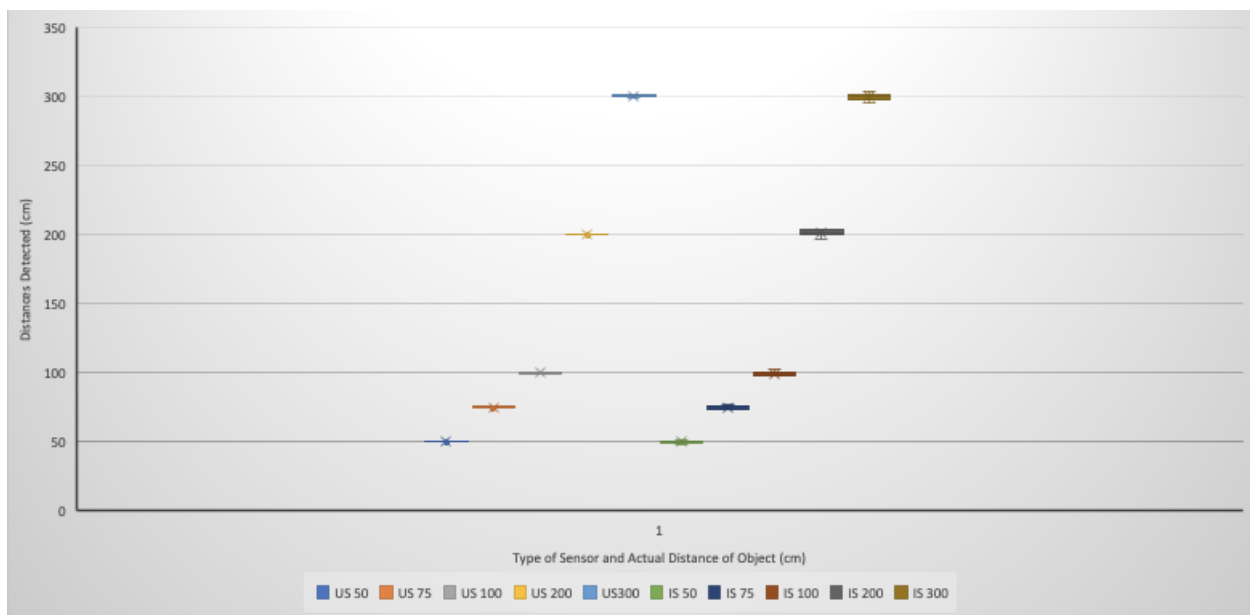


Figure 3: Testing of the Ultrasonic and Infrared Sensor on an Aluminum Object with Eccentricity of Infinity at Different Distances

The chart above shows testing on the object with an eccentricity of infinity. The data points seem to be almost exactly on their respective axes. The data appears accurate and precise as all data points are close to their axes. There is more variation in

the data for the ultrasonic sensor and greater distances, such as at 300 cm. The infrared sensor has slightly more precise data than the ultrasonic sensor.

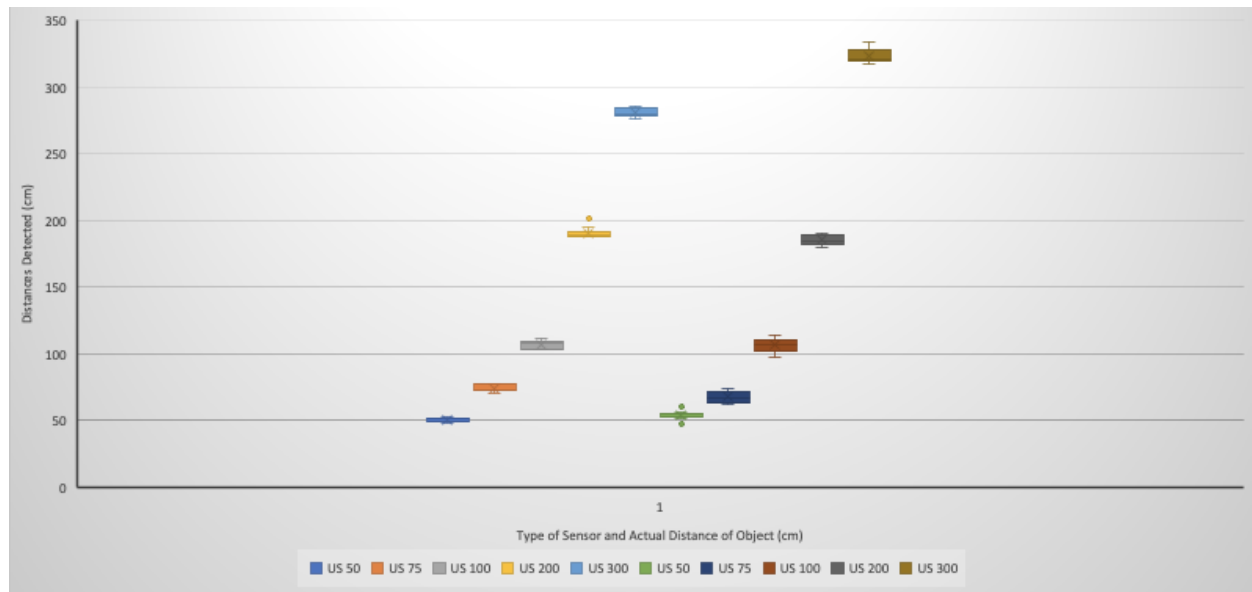


Figure 4: Testing of the Ultrasonic and Infrared Sensor on an Aluminum Object with Eccentricity of Two Convex Outward at Different Distances

The chart above portrays testing for when the eccentricity was two. The data points seem to be relatively far from their respective axes. The data appears to be not as accurate or precise as the box and whisker plots show a significant variation in the data and not that close to their axes. There is more variation in the data for the ultrasonic sensor and greater distances, such as at 300 cm.

Section 3.2: Results from A* Algorithm Testing

The overall success rate of the final version of the A* Algorithm was 98%. The tests were done through a series of 50 trials, where 49 were successful. No matter the number of obstacles present in physical testing, there was an accuracy of more than 92%. There was a 94% accuracy when using five objects and a 92% accuracy when tested with six objects. Trials of 50 were done for the physical testing as well.

Section 4: Discussion

The testing objective was to help develop a model that would allow individuals with visual impairments to walk outdoors safely and independently. Objects vary in curvatures, so it was vital to test whether the sensors detect the correct distances (from the sensor to the object). A navigation system was also built to allow users to navigate sidewalks by themselves and reach their final destinations. Overall, the results successfully created a model to help the visually impaired. For eccentricity testing, the programming model can detect the accurate distances of closer objects no matter the object's curvature. The navigation system showed more than 92% accuracy, showing a highly accurate model. As demonstrated, this model proves to be efficient and helpful to the client.

At further distances such as at 300 cm, there was a p-value of less than 0.0001, suggesting that the null hypothesis can be rejected. This rejection implies that the object's curvature did have a significant effect on the accuracy of the sensors. For closer distances like 50 cm, there was a p-value of slightly over 0.05, indicating an

insignificant change in the data, so the null hypothesis was accepted. This acceptance implies that the object's eccentricity does not affect the accuracy of the sensors close by. This optimizes the model because it shows that the sensors are still capable of accurately detecting objects at closer distances, just not at further distances. One possible way to fix the effect eccentricity has at farther distances is to employ two sensors angled toward the object (Khan et al., 2018). However, since entities will have different eccentricities, no set angle can account for all curvatures. The eccentricity did not have as much of an effect on the sensors at close distances. Thus, the path-finding algorithm should depend only on closer ranges for maximum accuracy.

For the algorithm and physical object testing, the accuracy was over 92.0% in the final stages of testing. This shows that the algorithm should simulate everyday objects on sidewalks for the most basic scenarios. The A-Star algorithm had a success rate of 98%, showing that it finds the shortest way to get from the starting node to the end node. The algorithm's success rate can be used to update the decision matrix and decide whether the new prototype is more effective than other solutions in the market. Using both the A* and path-finding algorithms, the program can fill the knowledge gap and create a cohesive solution to help the visually impaired navigate outdoors.

The model can be compared to other solutions available in the market. The BlindSquare application, for example, only uses a navigation system like Google Maps, but it does not alert the user of obstacles in front. This can cause hindrance as the route that the application suggests may not be feasible. The presented model allows the navigation system to work with obstacle detection to get around these objects.

Additionally, the path-finding algorithm is optimal for the user in real-time (Blum et al., 2020).

On the other hand, technologically advanced canes only allow users to know if they are getting closer to an object. The presented model allows the user to know their distance from the object, how to get around it, and then updates its navigation system to accurately identify the quickest path to their destination. Thus, the model is more effective than current technological solutions in the market (Rizzo, 2017).

There is a possibility that some errors could have occurred while testing. For example, when measuring the distance of the objects, there is an uncertainty value of one cm. The objects were placed using a straight edge to ensure the sensors were in a linear position, but there could be some human error. The humidity and temperature may also have affected the ultrasonic sensor as the sound is affected by the dampness in the air.

Overall, this solution helps bridge the knowledge gap by having a system that can help the user navigate and allow for the use of map applications such as the Google Maps API. It will help the user be safe while walking by knowing of objects up ahead and guiding them around objects in a contained environment.

References

- Bahmееi, F., Riazi, A., & Yoosfi, R. (2016). Outdoor difficulties experienced by a group of visually impaired Iranian people. *Journal of Current Ophthalmology*, 28(2), 85-90. doi:10.1016/j.joco.2016.04.002.
- Blum, J. R., Cooperstock, J. R., Panëels, S. A., & Varenne, D. (2013). The Walking Straight Mobile Application: Helping the Visually Impaired Avoid Veering. *International Conference on Auditory Display*. Retrieved October 17, 2020, from <http://srl.mcgill.ca/publications/2013-ICAD-WS.pdf>
- Chandakkar, P. S., Li, B., Paladugu, A., & Zhang, P. (2013). Supporting Navigation of Outdoor Shopping Complexes for Visually Impaired Users through Multi-Modal Data Fusion. *2013 IEEE International Conference on Multimedia and Expo (ICME)*. doi:10.1109/icme.2013.6607564.
- Gassart, R., Kim, Y., Oggier, T., & Riesch, M. (2012). *U.S. Patent No. W.O. 2012/040703 A2*. Washington, DC: U.S. Patent and Trademark Office.
- Gharpure, C., Kulyukin, V., Nicholson, J., & Pavithran, S. (2004). RFID in robot-assisted indoor navigation for the visually impaired. *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) (IEEE Cat. No.04CH37566)*, Sendai, 2004, pp. 1979-1984 vol.2, doi: 10.1109/IROS.2004.1389688.
- Home Automation with MIT App inventor and ESP8266. (2019, April 25). Retrieved March 07, 2021, from <https://iotdesignpro.com/projects/esp8266-iot-home-automation->

using-mit-app-inventor

Javed, A., & Minhas, R. A. (2018). X-EYE: A Bio-smart Secure Navigation Framework for Visually Impaired People. *International Conference on Signal Processing and Information Security (ICSPIS)*, DUBAI, United Arab Emirates, 2018, pp. 1-4, doi: 10.1109/CSPIS.2018.8642718.

Khan, I., Khusro, S., & Ullah, I. (2018). Technology-assisted white cane: Evaluation and future directions. *PeerJ*, 6. doi:10.7717/peerj.6058

Maps Javascript API Overview. (2020). Retrieved November 24, 2020, from <https://developers.google.com/maps/documentation/javascript/overview>

Patil, P., & Sonawane, A. (2017). Environment sniffing smart portable assistive device for visually impaired individuals. *2017 International Conference on Trends in Electronics and Informatics (ICEI)*, Tirunelveli, pp. 317-321, doi: 10.1109/ICOEI.2017.8300939.

Rizzo, J. R. (2017). *U.S. Patent No. US 9,646,514 B2*. Washington, DC: U.S. Patent and Trademark Office.

Satellite Navigation - GPS - How It Works. (2020). Retrieved November 02, 2020, from https://www.faa.gov/about/office_org/headquarters_offices/ato/service_units/tech_ops/navservices/gnss/gps/howitworks/

Taisbak, C. M. (2021). Conic Section. Retrieved March 07, 2021, from <https://www.britannica.com/science/conic-section#ref70765>

Vision Impairment and Blindness. (2020). Retrieved November 16, 2020,

from <https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment>

Vision Loss - Safety Around the Home. (2014, September 30). Retrieved March 07, 2021, from

<https://www.betterhealth.vic.gov.au/health/ConditionsAndTreatments/vision-loss-safety-around-the-home>

Appendices

Appendix A: Project Limitations

- Is not waterproof
- Less accurate with moving objects
- Does not detect drops in floor level (staircases or open maintenance holes)
- Does not detect objects past 300 cm away from sensors
- Does not use a GPS for navigation
- Only tested with curvatures of aluminum objects

Appendix B: Project Assumptions

- All sensors were optimally functional
- No other factor influenced the sensors' accuracies other than the distance and eccentricity of the object
- There was no bias by the person when walking in a physically simulated environment (the user was not purposefully avoiding an obstacle but rather following instructions of application)

Appendix C: Acknowledgements

Special thanks to Dr. Crowthers, Mrs. Taricco, and Mr. Loven for their guidance in their respective fields and for providing general advice to this project. Their continuous support allowed this project to become a reality.

Appendix D: Arduino Circuit Diagram

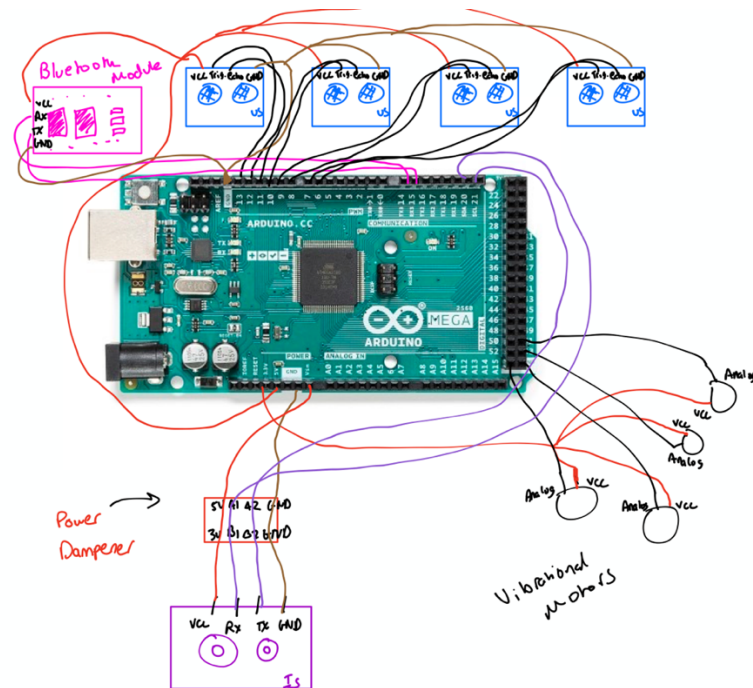


Figure 5: Wiring of the Arduino MEGA 2560

Appendix E: Code of MIT App Inventor IDE

```
package org.appinventor;

// this downloads all necessary classes for the features needed in the program
import com.google.appinventor.components.runtime.HandlesEventDispatching;
import com.google.appinventor.components.runtime.EventDispatcher;
import com.google.appinventor.components.runtime.Form;
import com.google.appinventor.components.runtime.Component;
import com.google.appinventor.components.runtime.Label;
import com.google.appinventor.components.runtime.ListPicker;
import com.google.appinventor.components.runtime.TextToSpeech;
import com.google.appinventor.components.runtime.BluetoothClient;
import com.google.appinventor.components.runtime.util.Yaillist;
class Screen1 extends Form implements HandlesEventDispatching {
    //uses list picker to pick from a variety of bluetooth modules around the user. Once they pick the
    //Arduino (through text-to-voice on their mobile device), the serial data will go to the application.
    private Label Label1;
    private ListPicker ListPicker1;
    private TextToSpeech TextToSpeech1;
    private BluetoothClient BluetoothClient1;
    private int direction;
    protected void $define() {
        //part of this just makes the application look visually appealing, however considering the visually
        //impaired are using this, it is not as important for the user as much as for the ease of the creation of
        //the app.
        this.AppName("steego");
        this.BackgroundColor(0xFFFFC800);
        this.Title("Screen1");
        Label1 = new Label(this);
        Label1.BackgroundColor(0xFF0000FF);
        Label1.FontBold(true);
        Label1.FontSize(20);
        Label1.Height(1010);
        Label1.Width(LENGTH_FILL_PARENT);
        Label1.Text("Welcome! This is the BlindPath Application!");
        Label1.TextColor(0xFFFFFFFF);
        ListPicker1 = new ListPicker(this);
        ListPicker1.BackgroundColor(0xFF0000FF);
        ListPicker1.FontBold(true);
        ListPicker1.Width(LENGTH_FILL_PARENT);
        ListPicker1.ItemBackgroundColor(0x00FFFFFF);
        ListPicker1.Text("Bluetooth");
        ListPicker1.TextColor(0x00FFFFFF);
        TextToSpeech1 = new TextToSpeech(this);
        BluetoothClient1 = new BluetoothClient(this);
        direction = 0;
        EventDispatcher.registerEventForDelegation(this, "BeforePickingEvent", "BeforePicking" );
        EventDispatcher.registerEventForDelegation(this, "AfterPickingEvent", "AfterPicking" );
        EventDispatcher.registerEventForDelegation(this, "InitializeEvent", "Initialize" );
    }
    public boolean dispatchEvent(Component component, String componentName, String eventName, Object[]
    params){
        if( component.equals(ListPicker1) && eventName.equals("BeforePicking") ){
            ListPicker1.BeforePicking();
            return true;
        }
        if( component.equals(ListPicker1) && eventName.equals("AfterPicking") ){
            ListPicker1.AfterPicking();
            return true;
        }
        if( component.equals(this) && eventName.equals("Initialize") ){
            this.Initialize();
            return true;
        }
        return false;
    }
    // this is the part that makes list of bluetooth modules
    public void ListPicker1BeforePicking(){
        ListPicker1.Elements(Yaillist.makeList(BluetoothClient1.AddressesAndNames()));
    }
    public void ListPicker1AfterPicking(){
        if(BluetoothClient1.Connect(ListPicker1.Selection())){
            ListPicker1.Elements(Yaillist.makeList(BluetoothClient1.AddressesAndNames()));
        }
    }
    // This part calls upon the serial data of the Arduino and matches with the directions given. If a
    //certain direction is present, it will use text-to-voice to alert the user of where to go.
    public void thisInitialize(){
        if(Integer.valueOf(0)==Integer.valueOf("0")){
            TextToSpeech1.Speak("Move forward");
        }
        if(Integer.valueOf(0)==Integer.valueOf("1")){
            TextToSpeech1.Speak("Take a left");
        }
        if(Integer.valueOf(0)==Integer.valueOf("2")){
            TextToSpeech1.Speak("Take a right");
        }
        if(Integer.valueOf(0)==Integer.valueOf("3")){
            TextToSpeech1.Speak("Retrace your steps");
        }
    }
}
```

Figure 6: Full code of the MIT App Inventor IDE JAVA bridged version

Appendix F: Final Cane Model



Figure 7: Front side view of the final product with all sensors, modules, and Arduino board fitted to respective angles and locations

Appendix G: MIT App Inventor Final Direction Application

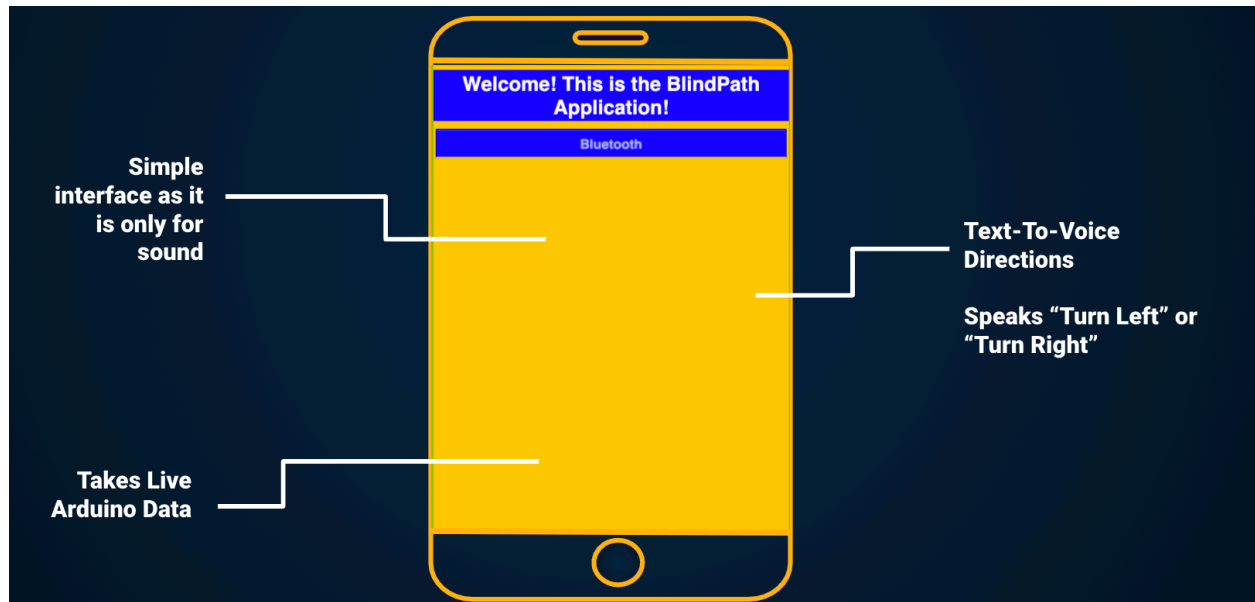


Figure 8: MIT App Inventor final application U.I. and features

Appendix H: Primary Cane Drawing

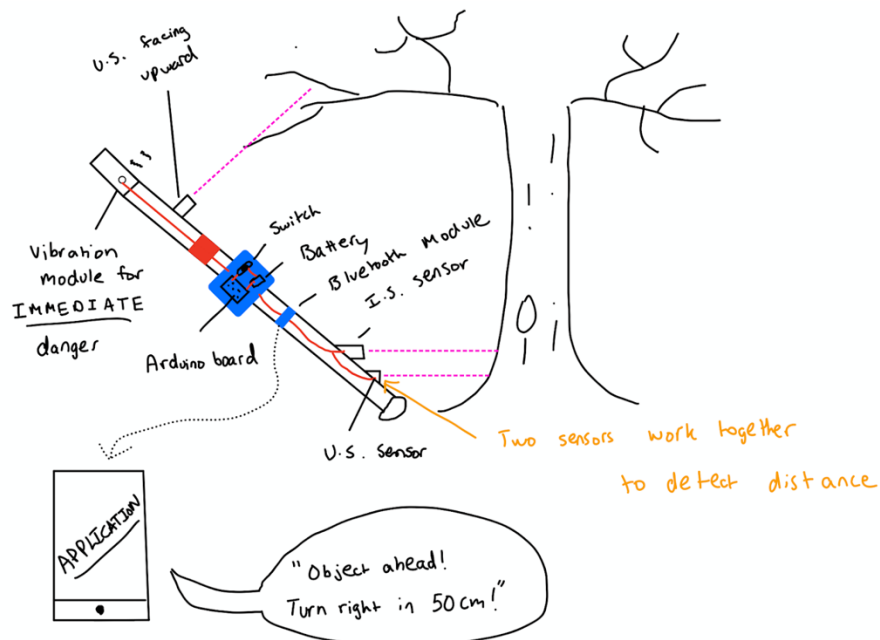


Figure 9: Cane drawing with built in features and modules

Appendix I: Experimental Testing Set-Up

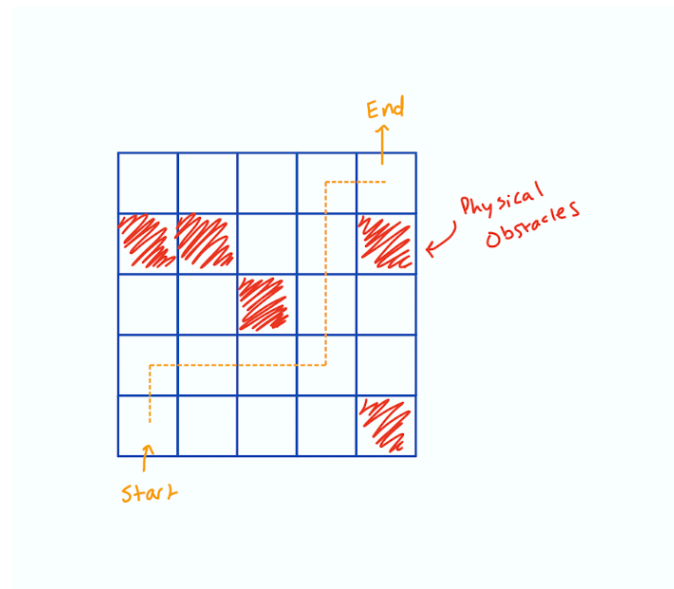


Figure 10: Experimental testing with five aluminum objects in simulated environment

Appendix J: Cane CAD Model

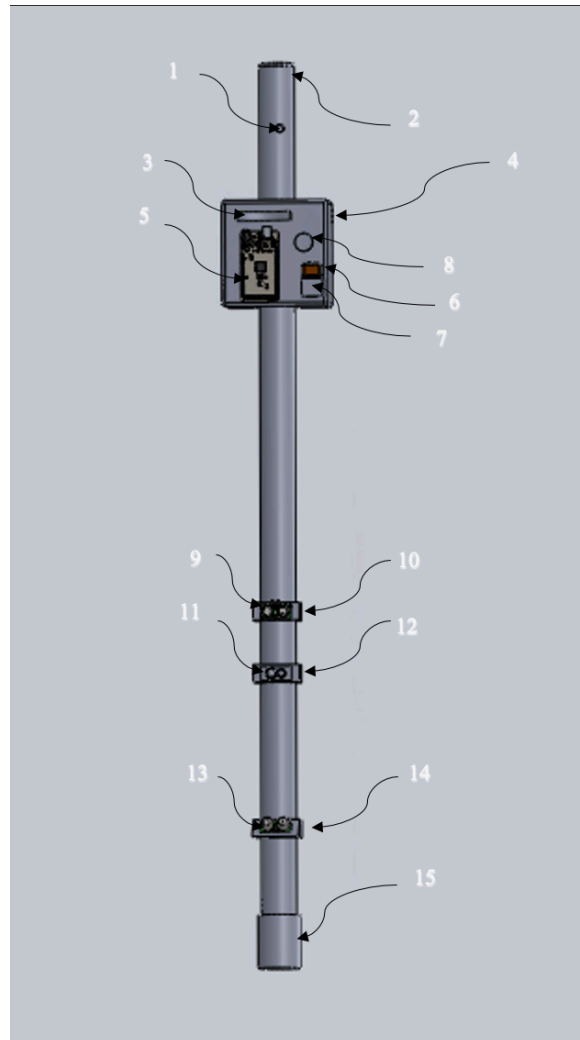


Figure 11: Front view of Computer Aided Design model of prototype

Appendix K: Eccentricity Testing

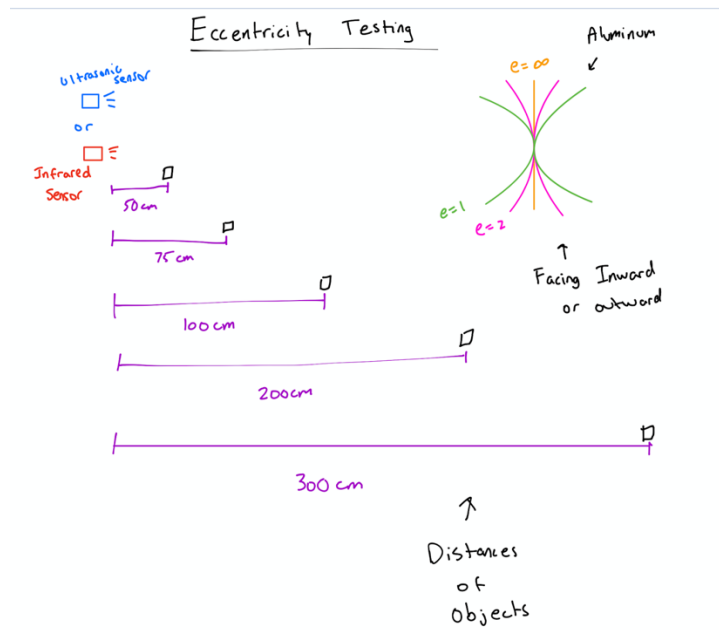


Figure 12: Testing the curvature, distance away from the sensor, and concavity of the objects with two different sensors