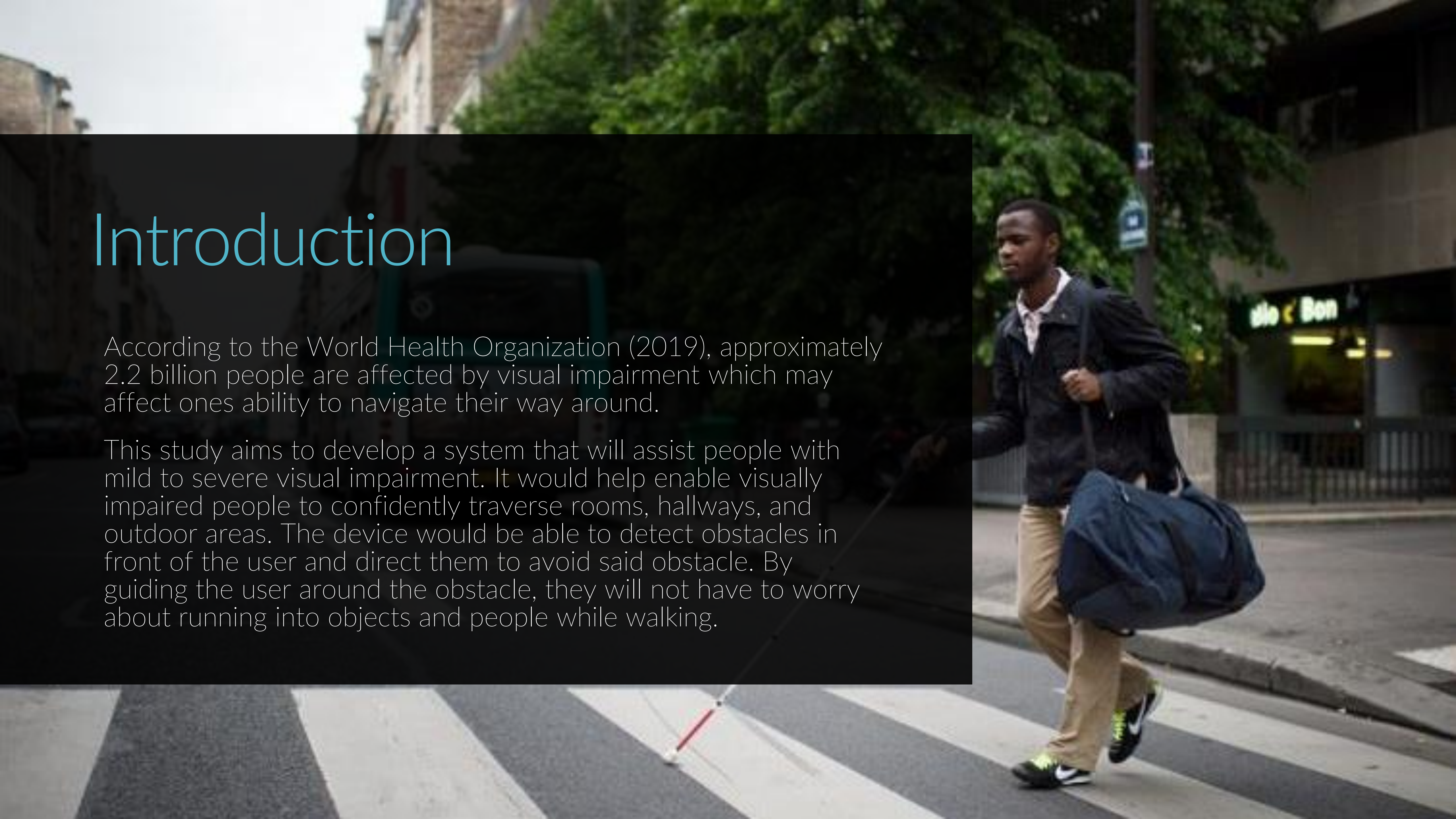

PATHFINDER
Device for Obstacle Detection and Avoidance to Help The Mobility of Visually Impaired People



Introduction

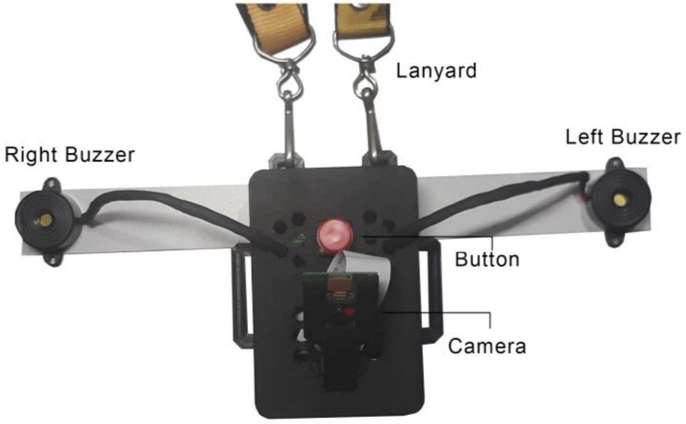
According to the World Health Organization (2019), approximately 2.2 billion people are affected by visual impairment which may affect ones ability to navigate their way around.

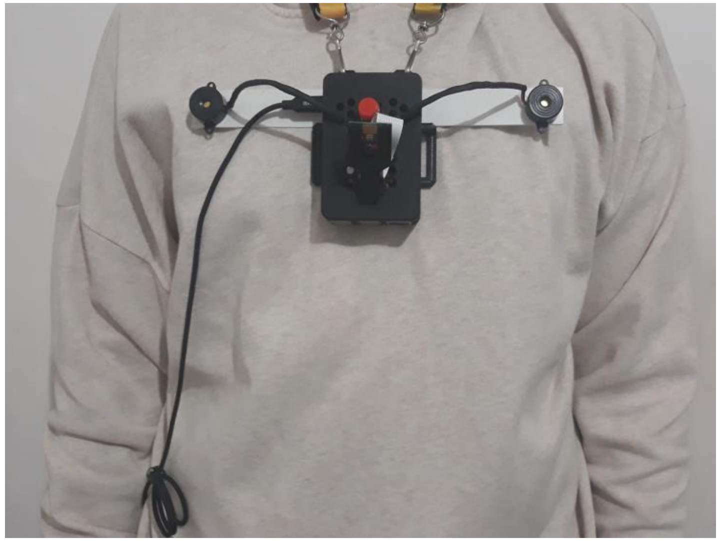
This study aims to develop a system that will assist people with mild to severe visual impairment. It would help enable visually impaired people to confidently traverse rooms, hallways, and outdoor areas. The device would be able to detect obstacles in front of the user and direct them to avoid said obstacle. By guiding the user around the obstacle, they will not have to worry about running into objects and people while walking.

Objectives

This study aims to create a portable device prototype which will help the navigation of visually impaired people

- To use digital image processing techniques for obstacle detection
- To detect obstacles within 3 meters from the user
- To notify the user of the obstacle and redirect them to avoid it





Methodology
Instrument Build Specifications

Components

- Raspberry Pi 4 (16GB storage, 4GB RAM)
- Raspian Buster OS
- 5MP Raspberry Pi camera module Rev 1.3
- 12000mAh powerbank / DC 5V 2.1A output

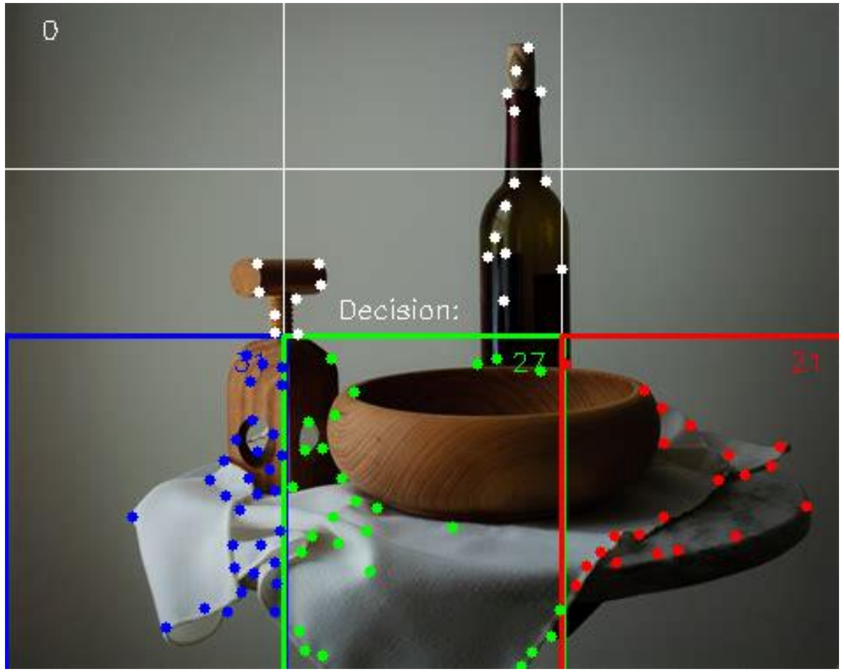
Peizoelectric Buzzers

3D Printed Housing

Methodology
Obstacle Detection

Grayscale was first be applied into the frame for better measurement of the varying pixel intensity.

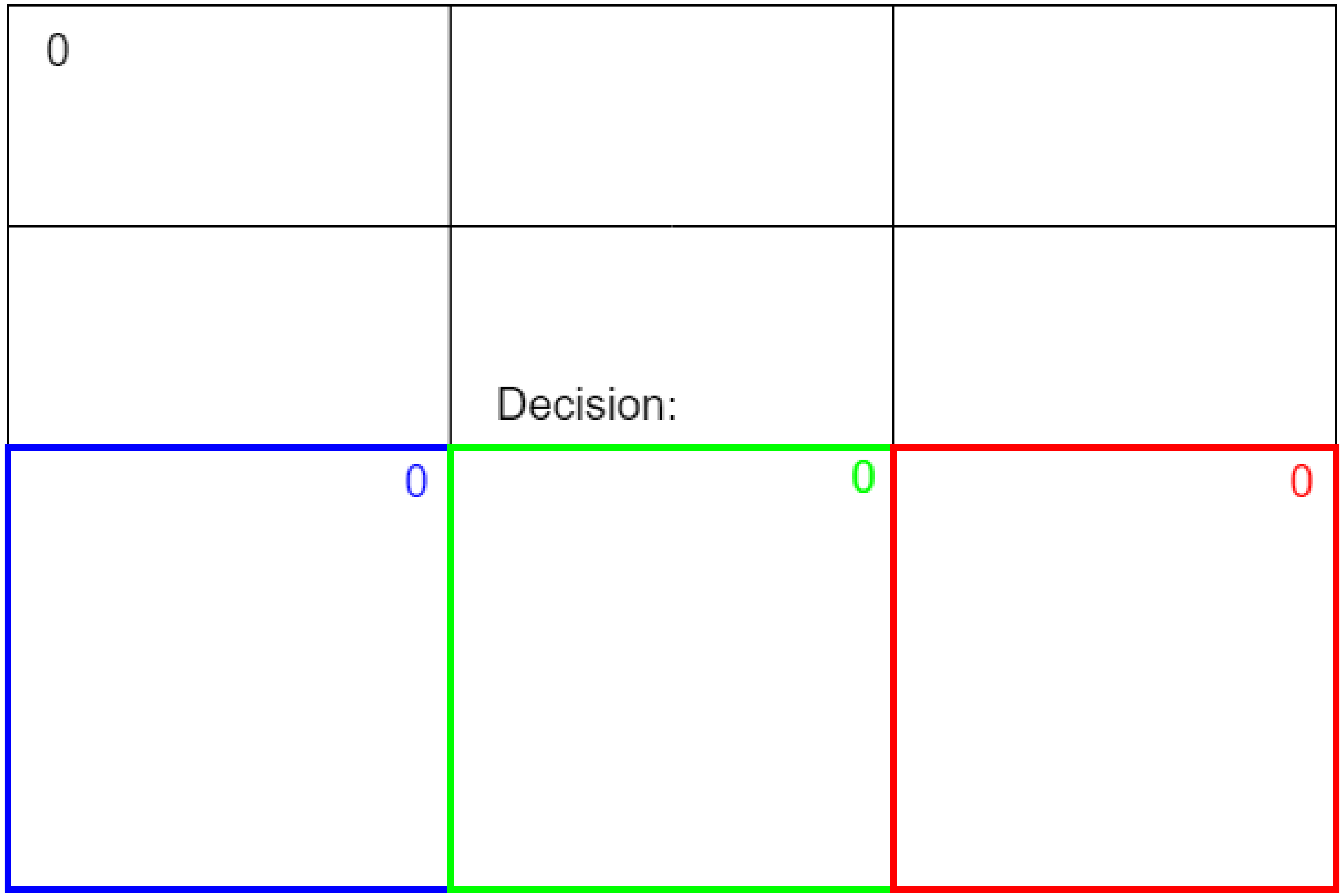
Noise reduction was accomplished by applying a Gaussian blur to smooth the frame.



Methodology
Obstacle Detection

The Shi-Tomasi Algorithm was applied to detect the feature points of an object. The feature points are the basis for detected obstacles.

The gray and blurred picture is the basis for the Shi-Tomasi Algorithm but the result are projected unto the original image.

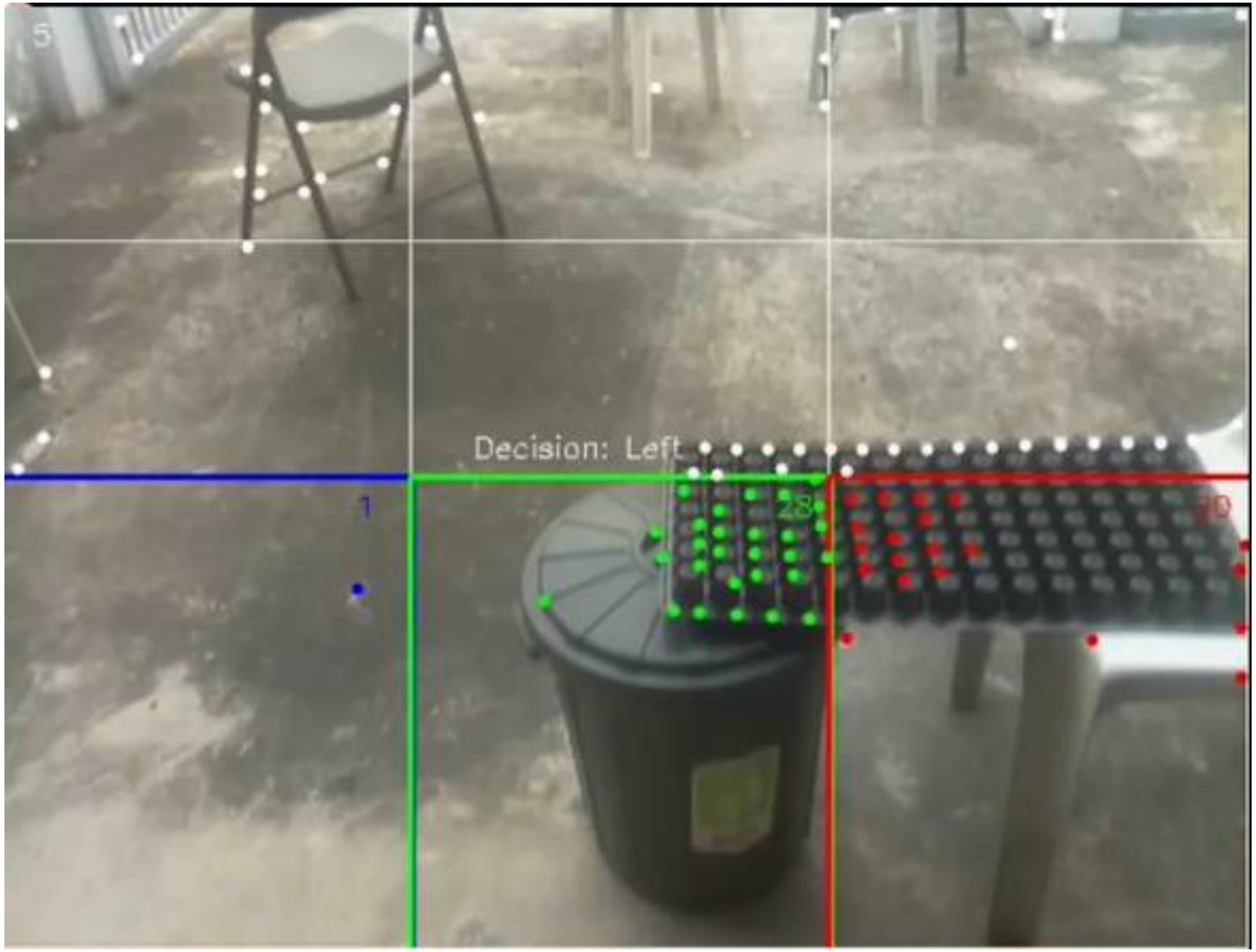


Methodology
Obstacle Detection

The top left region contained a timer which shows how much time in seconds have past.

The bottom row of the search space, divided into three parts, served as the region of interest (ROI). Each ROI were differentiated by color with the left being blue, the center being green, and the right being red.

Each ROI had a counter in their upper right corner of the number of points detected within itself. The number of feature points per region was counted. A threshold of 0.01 determined a feature point. Points Within the search space were counted as obstacles.



Methodology
Decision Making

If an obstacle appeared in any of the three areas, the program directed the user to the nearest area with the least amount of feature points via the use of two piezoelectric buzzers.

The user is directed to step to the left if the left buzzer beeps

Right if the right buzzer beeps

Forward if both the left and the right beeps.

Results and Discussion

Testing the Accuracy of the Obstacle Detection Software

Testing comprised of 10 different paths with varying amounts of obstacles.

The device was run for each path and its input was recorded.

The number of obstacles found were counted by a human observer.

The number of obstacles detected by the device was also recorded.

The precision of the system pertains to the relevance of the points detected.

The recall pertains to the correctness of these points.

Results and Discussion

Testing the Accuracy of the Obstacle Detection Software

Precision = $\frac{|G \cap D|}{G}$ Recall = $\frac{|G \cap D|}{D}$

G – obstacles detected by a human observer

D – obstacles detected by the device

Table of Results

Path Number	Obstacles found by Human (G)	Obstacles found by System (D)	Precision $|G \cap D|/G$	Recall $|G \cap D|/D$
1	8	6	0.75	1
2	9	8	0.89	1
3	10	8	0.8	1
4	10	8	0.8	1
5	11	11	1	1
6	12	11	0.92	1
7	13	11	0.85	1
8	14	10	0.71	1
9	15	12	0.8	1
10	16	12	0.75	1
			0.83	1

Results and Discussion

End-user Testing

The trials are meant to compare the effectiveness of the different methods of navigation.

The first trial required the user to traverse the obstacle-ridden path with only a blindfold.

The second trial required the user to traverse the same pathway with a blindfold and a cane.

The third trial required the traversal of the same pathway using the device itself.

Test Environment

Results and Discussion

Time and Mistakes

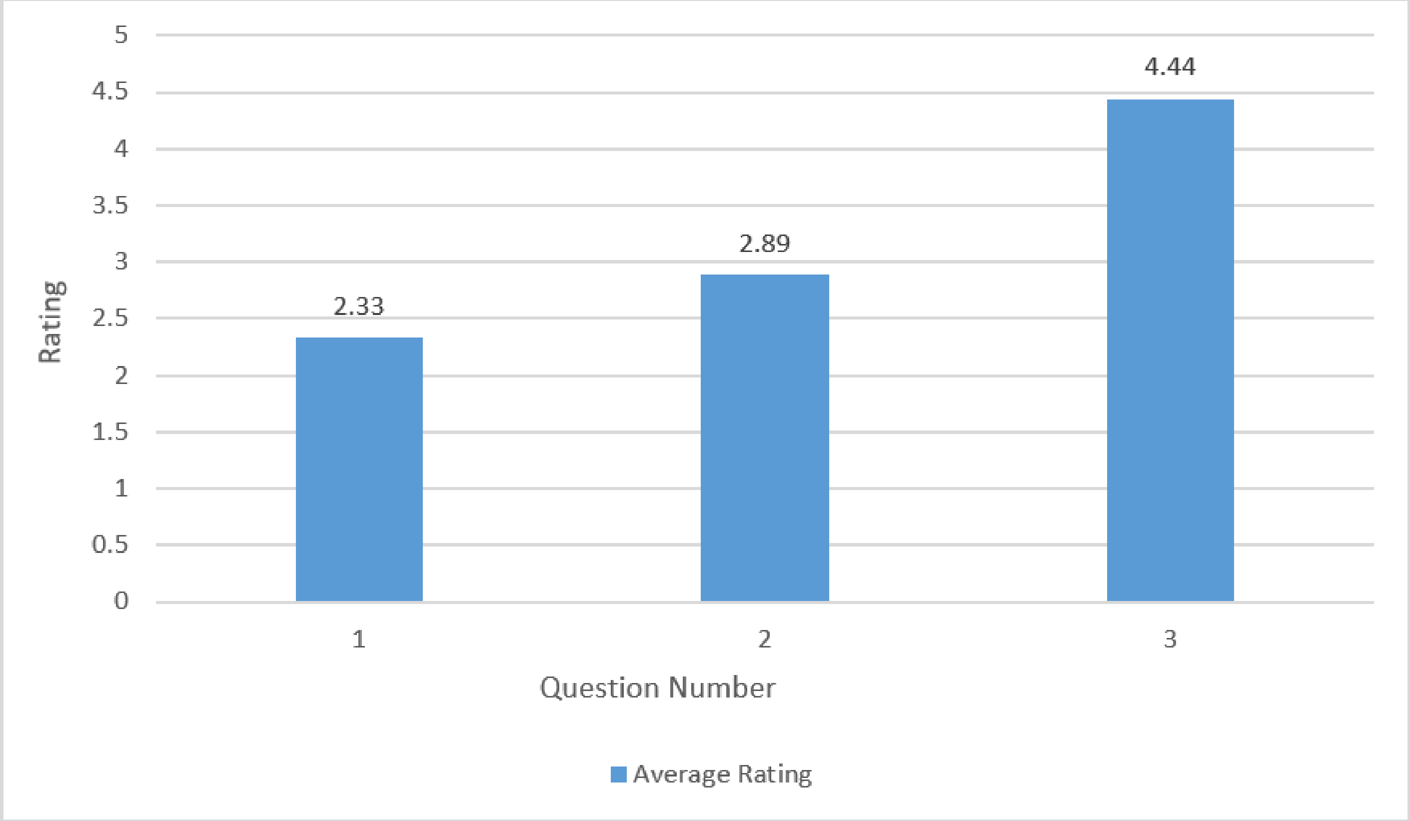
Average Time it Took to Traverse the Path

Test No.	User									Min	Max	Mean
	1	2	3	4	5	6	7	8	9			
1	20.29	20.08	16.31	17.04	31.88	36.64	43.55	22.62	38.00	16.31	43.55	27.38
2	28.24	17.21	18.07	17.02	22.90	20.91	36.64	24.22	32.62	17.02	36.64	24.2
3	36.92	16.95	19.70	18.37	29.33	41.23	40.65	23.00	39.70	16.95	41.23	29.54

Average Number of Mistakes

Test No.	User									Min	Max	Mean	Mode
	1	2	3	4	5	6	7	8	9				
1	6	3	2	2	4	3	3	2	6	2	6	3.44	2, 3
2	2	0	0	0	0	0	0	0	0	0	2	0.22	0
3	2	0	0	0	2	2	2	2	1	0	2	1.22	2

5 - Strongly agree
4 - Agree
3 - Neither Agree nor Disagree
2 - Disagree
1 - Strongly Disagree

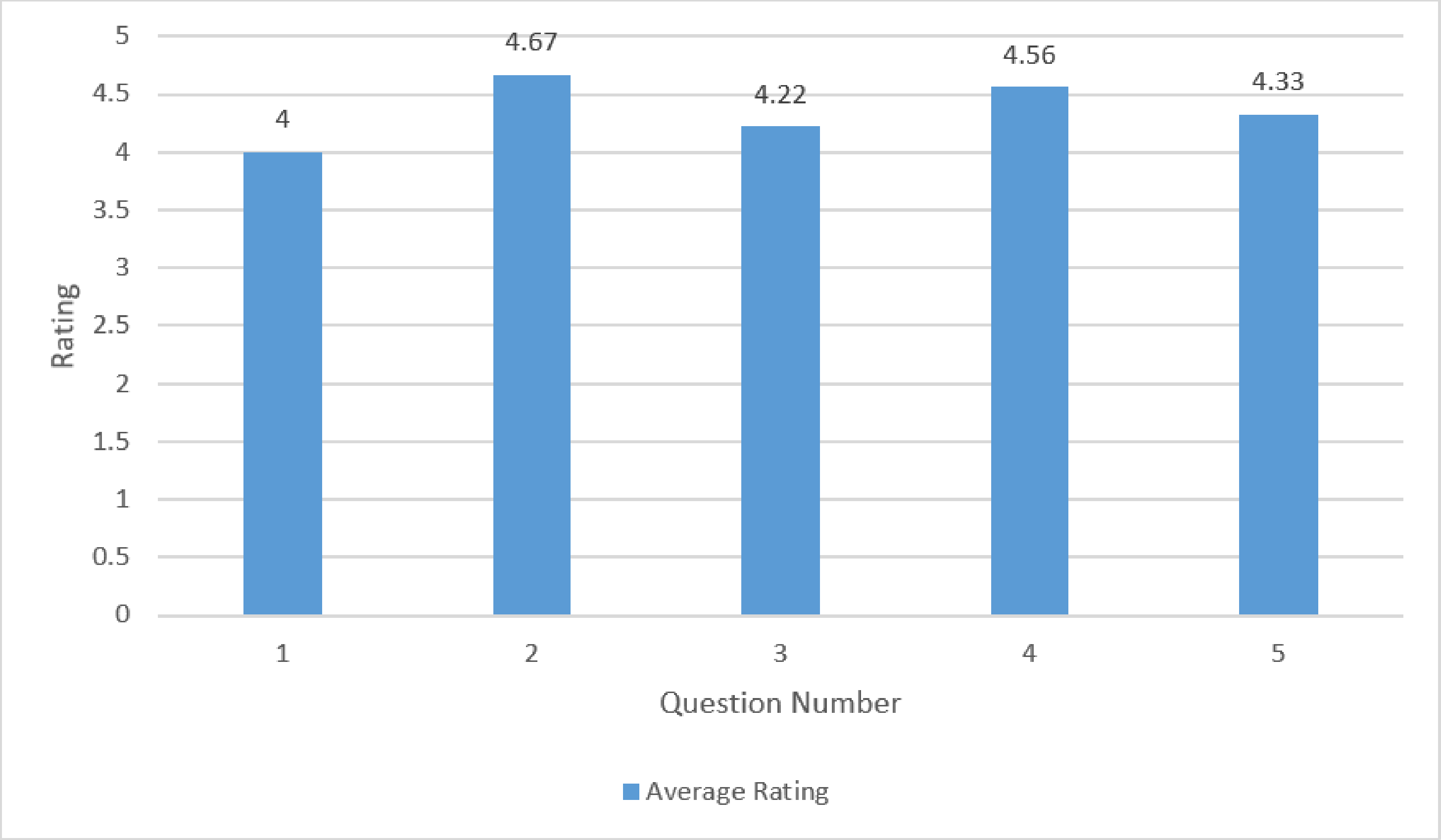


Average Ratings of Survey Results for Blindfold Only

Results and Discussion

Survey Results for Blindfold Only

5 - Strongly agree
4 - Agree
3 - Neither Agree nor Disagree
2 - Disagree
1 - Strongly Disagree

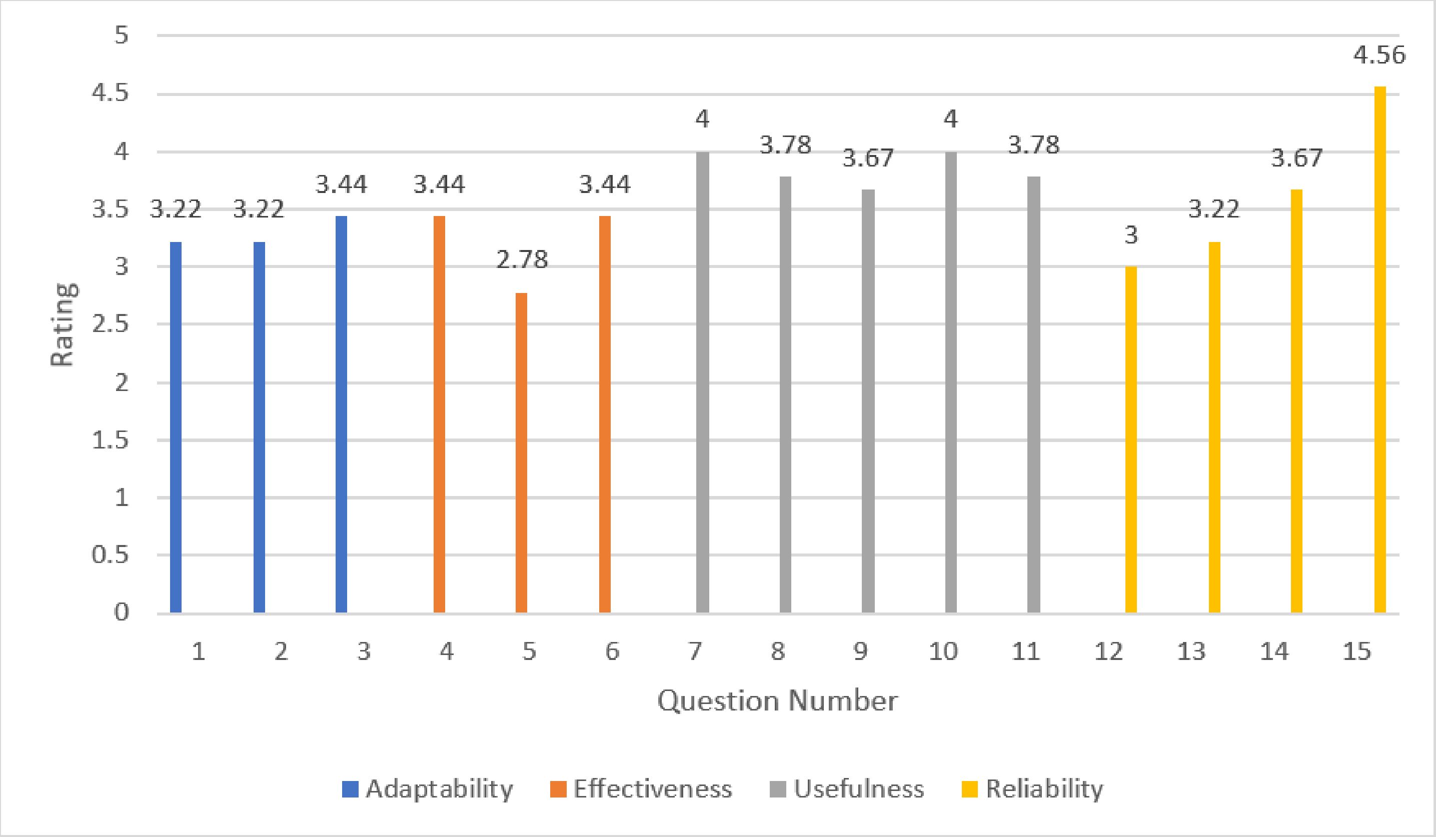


Average Ratings of Survey Results for Blindfold and Cane

Results and Discussion

Survey Results for Blindfold and Cane

5 - Strongly agree
4 - Agree
3 - Neither Agree nor Disagree
2 - Disagree
1 - Strongly Disagree



Average Ratings of Survey Results for System Use

Results and Discussion

Survey Results for System Use

Conclusion and Recommendations

4.56

Average Rating on production of a successful device.

1.22

Average Mistakes made with the device following the 0.22 average mistakes with the cane.

Hardware

Gimbal system

Buzzers

Software

Distance Covered instead of time

Wall Detection