Navigator for the Blind

An Interactive Qualifying Project
submitted to the Faculty of
WORCESTER POLYTECHNIC INSTITUTE
in partial fulfilment of the requirements for the
degree of Bachelor of Science

by
Maranda Allen
Samantha Boyea
Arden Carling
Connor Miholovich

Date:
May 12th, 2021

Report Submitted to:

Kira Kovnat
Eurasian Peoples' Assembly

Mikhail Koroteev
Financial University under the Government of the Russian Federation

Ivan Mardilovich and Karen Oates
Worcester Polytechnic Institute

ABSTRACT

This report discusses a proposed solution to indoor navigation for blind and visually impaired people in Russia. Utilization of built-in smartphone cameras and sensors as well as machine learning led to the investigation of a solution in the form of an iOS application. Implementation of a smartphone application as a navigation technology has been researched, decided on and worked on as a collaborative effort with a team from the Financial University under the Government of the Russian Federation, the Eurasian Peoples' Assembly, the Center for Sociocultural Rehabilitation of Diana Gurtskaya, and the Udmurt Regional Children's Public Organization.

ACKNOWLEDGEMENTS

Our team and WPI would like to thank the following people and organizations for their support throughout the course of this project:

Students from the Financial University under the Government of the Russian Federation

Professor Mikhail Koroteev

Professor Anton Losev

Eurasian Peoples' Assembly

Kira Kovnat

Julia Petrov

Center for Sociocultural Rehabilitation of Diana Gurtskaya

Katya Zaitseva

Udmurt Regional Children's Public Organization

Olga Chirkova

Professor Karen Oates

Professor Ivan Mardilovich

Professor Svetlana Nikitina

Professor John "JZ" Zeugner

AUTHORSHIP

This report was a collaborative effort by all team members. All members provided equal contribution to the research and writing.

TABLE OF CONTENTS

TABLE OF FIGURES

TABLE OF TABLES

EXECUTIVE SUMMARY

The goal of this project is to propose a solution for indoor navigation for blind and visually impaired people in Russia. Currently, in Russia, there is a lack of assistance available for blind and visually impaired people to navigate indoor spaces independently. Both orientation and object avoidance are challenging for these individuals in any environment without the help of a guide. However, challenges unique to unfamiliar indoor spaces include both static and dynamic obstacles such as chairs, tables, stairs, and other people. The lack of assistance for blind navigation in Russia demonstrates the need for a technical solution with the ability to provide both orientation and obstacle avoidance information to the user.

For our research we have determined that there are four major elements to account for in the development of such a tool. The elements that have been identified are the following: seeing from the eyes of blind people, social considerations, teaching blind individuals to navigate, and existing technologies. Each of these elements indicate an essential research topic in understanding the challenges faced by blind individuals in indoor environments. A review of literature has allowed for the formulation of survey questions that target these essential elements, and also informed our prototyping process.

Often technology is used to help individuals to understand new things. Review of literature reveals that many current technical solutions are hardware dependent fixed designs. Such designs are inflexible and hard to set up. This is because indoor spaces need to be outfitted with sensors or cameras beforehand. It was identified that the major challenges associated with the development of a navigation system are accessibility, maintainability, and cost. With this in mind it was decided that the design process should be divided into two separate sections: sensing technology and vehicles to make this technology mobile. Some of these vehicles include companion robots, cellular alert systems, and wearable or virtual reality technology.

Research into the utilization of built-in Smartphone cameras and machine learning led to a solution to be presented in its final form as an iOS application. This iOS application uses training data based on particular buildings and objects to warn blind people about hazards in their proximity while providing directions to a destination. The application converts the hazards and directions into synthesized speech, which can be played back to the user through the integrated speaker or a pair of headphones. These methods of navigation and feedback were selected based on background research, feedback from survey participants, as well as the technical expertise of the team from the Moscow Financial University.

The methodology explores solutions to determining location, orientation within a space, determining how to track objects that are spatially changing, and determining how to notify users of obstacles and orientation. Developing orientation and location solutions was done by developing a 3D model of buildings which implements a live model of the building for the user to navigate through as the user moves through the building. Tracking spatially changing objects used OpenCV image detection software to take live video from cameras and identify objects in real time. Notifying the users of obstacles used record sounds which feature clear commands such as "Walk Straight Ahead" and analog tones such as a beeping noise.

Implementation of a smartphone application as a navigation technology has been researched, decided on, and worked on as a collaborative effort with a team from the Financial University under the Government of the Russian Federation, the Eurasian Peoples’ Assembly, the Center for Sociocultural Rehabilitation of Diana Gurtskaya, and the Udmurt Regional Children’s Public Organization. The team progressed toward a viable solution for safe technology-assistive blind navigation and the collaboration between teams led to the development of a functional prototype. The assessment of prototype functionality was focused on how users' interface with the program and the ease of use of the application. This assessment also included a determination of how well the application interfaces between software and hardware. In other words, how accurate the application is at navigating are orienting users.

It was concluded that the prototype produced as a deliverable for this project was able to successfully and accurately navigate and orient a user in an unfamiliar indoor space. This paper concludes with seven recommendations that will help guide the continuation of this project. These recommendations encompass the continued software, hardware, and interface development of an application-based navigator for the blind. Suggestions were derived from the research, observations, and results that were compiled upon the conclusion of this project.

CHAPTER 1: INTRODUCTION

If you had to navigate from one floor of a building to another without your eyesight, what sort of tools or assistance would you want to help you move about? While blind people may be perceived as having low spatial awareness, lowered spatial awareness is only true for frontal spatial awareness. With the loss of eyesight, other senses such as hearing, and touch become heightened to compensate for the loss of sight (Harvey, L.). A key technique blind or visually impaired persons use for navigation is clues from the spaces around them, such as determining the size of a room from how sounds resonate through the space (Gori, M.).

The following paper describes research that could be applied to helping blind or visually impaired persons navigate indoors through the use of a smartphone application. The application is being developed for Russian students and may eventually be scaled up after the concept is proven to work efficiently. Currently in Russia, there is a lack of aid for blind and visually impaired persons to navigate through public spaces and indoors (Giudice, N. A.). Without some form of aid, blind or visually impaired persons are subject to difficulties with orientation and object avoidance. While non-moving objects such as walls are less difficult to navigate around, stairs, doors and moving objects, such as people or chairs, prove to be difficult to accurately detect and avoid. Inadequate object avoidance skills produce danger to blind or visually impaired persons as they are subject to falling and collisions. The lack of navigation assistance in Russia indicates a need for the development of a technology that can be easily accessible, mobile, and provide users with orientation and object avoidance tools.

For blind persons, existing navigation techniques are primarily seeing aides (guide dogs or person), canes, robots, handheld cameras, and smartphone applications. Many of these technologies are expensive, non-mobile, heavy, and not readily accessible (Killic, D. K). The goal of the project is to make the navigation technique as accessible as possible, so consolidating the navigation methods to a smartphone will allow for maximum accessibility to all users. Our sponsors indicated that the project should only use sensors and features available through a smartphone without the use of additional technologies for optimization of accessibility and ease of use. As a result of our sponsor's instructions, our group is limited to the use of a smartphone application as opposed to other existing methods. Our proposed methodology will be different from current smartphone navigation applications because it will not require the use of GPS or data and information from a server. The application we will be developing will be exclusively using data from the phones sensors which will improve the speed, reliability, and overall performance of the application. (Schinazi, V. R.)

The choice of using sensors built into smartphones has three primary purposes. Primarily, using smartphone included sensors allows for high mobility through reducing the need of carrying additional technologies which are traditionally large and heavy. Secondly, allowing the application to be accessible solely through smartphones makes the implementation into society practical through a readily available technology. Lastly, containing the sensor feedback within the smartphone allows the application to receive real time accurate data from the user that does not have to be transported between services such as from a satellite to a smartphone. The primary technology being will be motion analysis software paired with a machine learning algorithm to provide object avoidance output to the user. Motion analysis software is ideal for object avoidance as it works well with short ranges and is most accurate when used in dynamic environments and when receiving images from the device's camera. User orientation will be accomplished through accessing smartphones accelerometers and gyroscopes to provide the user with their current heading and position relative to themselves. (Schinazi, V. R)

Following the introductory chapter will be background information on the problem at hand, a literature review of similar studies, and a methodology for the proposed solution to the problem. The background and literature review will highlight the problem and justify the proposed solution that will be discussed within the methodology section. The current state of the project is the Russian team has begun testing the technology, primarily, how the phones cameras can be interfaced through an application installed on the phone. The Russian team has developed its initial code for testing interfaces and the American team from WPI has begun to develop code for interfacing orientation sensors (gyroscopes and accelerometers) from the smartphone. No testing has been done with subjects or environments, all work has been completed through software simulation and hardware testing. Further information regarding testing and the current state of the project is discussed in the methodology section of the report.

In order to complete the development of an audio navigator for the blind the American team will be working in conjunction with a team of students from the Financial University in Moscow. The two teams will work in collaboration towards the development of a working navigator prototype and in the assessment of this prototype. The Russian team will be primarily responsible for the IOS software and application development, and the American team will be primarily responsible for the sensor and mathematical modeling development. In collaboration between teams, the final product will be a beta development of an application that can be used to navigate through buildings and find rooms without collision with obstacles. This project is made possible by the Eurasian Peoples' Assembly, the Center for Sociocultural Rehabilitation of Diana Gurtskaya, and the Udmurt Regional Children's Public Organization. These Russian based organizations are working towards improved accommodations for the disabled in public facilities throughout Russia. The Diana Gurtskaya Center was founded by famous blind Russian-Georgian singer Diana Gurtskaya who has a passion for improving universal accessibility for the blind. These generous sponsors will help provide us with a testing environment as well as aid in the future implementation of technology based blind navigation in Russia (Eurasia Peoples' Assembly).

This project has the potential to improve the lives of Russian blind individuals. In the next chapter we will discuss the negative physical and social implications of blindness in both the United States and in Russia. Navigating for blind or visually impaired persons can have negative impacts on Social, motor, language and cognitive skills (Martinez, C.). A technical solution for blind navigation in unfamiliar indoor spaces has the potential to improve the mobility of blind or visually impaired persons while ensuring safety and comfort while performing independent movement (Martinez, C.). By providing blind or visually impaired persons with improved independent movements we are able to directly improve cognitive and motor skills through the navigation application. Through the ability to navigate and explore new environments safely, blind or visually impaired persons are able to improve social and language skills on their own accord.

CHAPTER 2: BACKGROUND

The goal of the project is to develop a system to help visually impaired persons in Russia, ideally of all ages, navigate indoor settings safely. Currently Russia has rehabilitation and education for blind people to help them navigate society, but the process heavily relies on the person to rely on their senses such as touch from a seeing cane. Hence, blind persons are subject to a steep learning curve in relying on their senses to navigate without collisions, often inducing social timidness. Navigation technologies for the blind tend to be expensive, privately owned, hardware dependent, and heavy making these solutions unideal for traveling usage or public usage. Due to the lack of public aid, there is a need for a publicly accessible, and cost-effective solution for helping blind or visually impaired persons navigate in Russia.

The team's research and review of literature on navigation aids for blind or visually impaired persons indicates there are four critical elements to be grasped to begin to solve the problem of blind persons navigating safely through indoor spaces. These elements are seeing from the eyes of blind people, social considerations, teaching blind children to navigate, and existing technologies. Each of the elements will be areas that, through our teams' research and review, highlight many of the problems the project will be looking to solve.

2.1 Seeing from the Eyes of Blind People

Without the sense of sight exploring surroundings may be startling and frightening to blind children due to the unpredictability of the environments around them. Blind children are left to use touch, smell, taste and hearing to understand the world around them and the space they are in. Learning basic body awareness and movements are much more difficult for a visually impaired child than for a child with vision. "In an unfamiliar environment, a blind person faces a number of difficulties such as accessing spatial information from a distance, obtaining directional cues to distant locations, keeping track of one's orientation and location and obtaining positive identification once a location is reached, whereas sighted persons use visual landmarks and signs in order to orient themselves" (Kan Kilic, 2016). An obstacle blind children face is learning spatial concepts; specifically, the understanding that objects exist even if not heard or felt. Developing searching skills (locating items) and independent movements (such as walking without collisions) are the most critical skills for a visually impaired person to have to understand the world around them. To gain an understanding of searching skills, one exercise is to set a plate and blunt cutlery on a table, and then, while blindfolded, try to accurately locate the items. In an attempt to find the objects, chances are you will have to run your hand across the table in patterns until you locate the items solely using your sense of touch to locate the items (Gori et al., 2017).

To understand how blind or visually impaired people perceive space, an important distinction must be made; the difference between people born blind and those who lose sight. People who lose their eyesight over life still "see" or have memory of what an object looks like from before losing their vision. While people who are born blind memorize non-visual aspects of space such as texture or shape and memorize important features and how they combine to make a whole shape/object. A study done "highlights the holistic and multidimensional perception of space by asking blind people to mark those places that they find particularly important in complex urban environments by verbally describing the features in the environment they attend to." (Kan Kilic, 2016). While blind people may not be able to visually create an image or construct an assembly of an object, it is important to understand that they are still able to perceive objects in multidimensions. Understanding objects in multidimensions for blind or visually impaired persons focuses on a shapes size relative to other objects through touch, rather than perceiving size relative to other objects through visual deviations. When teaching and understanding spatial awareness to blind or visually impaired persons, it is important to use multidimensionally rich environments to stimulate the understanding of space and depth perception.

2.2 Social Considerations

A prominent challenge that blind and visually impaired people face is the ability to navigate around a space that they are unfamiliar with. Going to a new place challenges their ease of mobility as indoor spaces contain many obstacles. Visual impairments can also present a challenge to one's social life (*Challenges Blind People Face When Living Life*, 2019). When planning to go to a new location, the accessibility of the location and the ability of the blind and visually impaired to navigate within it can have a large impact on their experience. Additionally, assumptions made by people with normal vision about a blind or visually impaired person's ability to be independent and responsible for themselves in a new place can be hurtful and make a social environment uncomfortable (Beteinaki et al., n.d.). In a study done in 2019 about how stereotypes of those with blind and low vision affect their social participation, it was found that "among individuals with 'normal' vision, even those who have positive intentions, their responses can have a negative impact on the social participation of older adults with vision loss" (Fraser et al., 2019). Other results from the study indicate that negative responses, ignorance, stigmas and a sense of being devalued by those with normal vision make social experiences a challenge to those with visual impairments (Fraser et al., 2019).

Eye health statistics gained from a survey done by American Academy of Ophthalmology on eye related injuries in 2015 (*Eye Health Statistics - American Academy of Ophthalmology*, n.d.), in conjunction with United States census data from 2015 (US Census Bureau, 2015), 2.81% of adults 40 years and older had a visual impairment. In the census study, visual impairment refers to anyone with low vision and legal blindness. Low vision, which is defined as the best corrected visual acuity that is worse than 20/40, made up 1.94% of Americans 40 years and older. Legal blindness, which is defined as the best corrected visual acuity worse than 20/200, was present in 0.87% (*Eye Health Statistics - American Academy of Ophthalmology*, n.d.). In the United States, there are a variety of resources for blind and visually impaired children and their families. As of 2017, there are more than 50 schools and agencies in the country that cater to the educational needs of blind and visually impaired individuals (Harris, 2017). Additionally, there are agencies such as the American Foundation for the Blind that provide families with access to information and answers to common questions. They provide information about the rights and laws surrounding visual impairments, what to do after receiving such a diagnosis, and ways for the families of blind and visually impaired people to connect. In the United States, being blind or having a visual impairment is a disability that has many supporting resources.

In the history of blind people in Russia, one of the most prominent mentions is the town of Rusinovo, as seen in Figure one, in the Kaluga region. In the Soviet Era, those who registered as being blind or visually impaired would be contacted by social services and moved to Rusinovo.

Figure 1: Town of Rusinovo

Here there were houses built to specifically accommodate the blind and visually impaired, as well as a factory in which they could work, run by the Russian Association of the Blind. In the late 80s and 90s, blind and visually impaired individuals could earn up to twice as much money as a typical engineer as their pay was based on the quantity of work they did. However, when the partner factory in Moscow shut down, the factory and the town of Rusinovo hit hard times (*Documentary: Russia's Village Of The Blind*, 2017). Information on this town was primarily found from a documentary by RadioFreeEurope RadioLiberty, though more was searched for, there was a lacking number of written resources. More generally, the perception of those with disabilities in Russia, especially children, is one of weakness and helplessness (Nosenko-Stein, 2017). Some children with less severe disabilities, such as low vision but not blindness, escape opprobrium. Those with more severe disabilities are often sheltered from even the simplest of house-tasks, experiencing an extreme amount of care that inhibits their ability to gain much independence (Nosenko-Stein, 2017).

A study published in July of 2020 in Scientific Reports discussed the prevalence of different levels of vision impairment in Russia, ranging from mild, to moderate-to-severe, to blind. The study found that out of the 5,899 participants, 3.1% reported mild vision impairment, 3.1% reported moderate-to-severe vision impairment and 0.9% reported blindness (Bikbov et al., 2020). Another report from 2014 discussing evacuation plans for blind and visually impaired individuals in the case of a fire stated that the All-Russian Association of the Blind claimed there were 218 thousand visually impaired people in Russia (Samoshin & Istratov, 2014). Out of five schools in Russia that cater to blind and visually impaired individuals, four of them are located in Moscow and one is in St. Petersburg (*ICEVI-Europe - National Pages - Russia*, n.d.). The low number of schools leaves those farther from these cities with less access to resources tailored to visual impairments and blindness. One journalist noted her perspective of the response to disability in Russia to "seem downright unfriendly" with an emphasis on the "total lack of accessibility for the disabled" (Leyshon, 2012). Leyshon (2012) notes the lack of Braille signs or elevators to help blind and visually impaired individuals navigate in and around buildings. In Russia, support educationally and navigationally are still areas in need of growth and improvement.

2.3 Navigation Skills for the Visually Impaired

For visually impaired people, developing a sense of independence and the ability to safely explore the world around them without direct supervision involves developing spatial awareness through searching skills and independent movements. Sighted guides (such as a dog or parent) and cane skills are common ways to allow visually impaired people of all ages to explore new locations. The cane requires that the person is fairly confident in their searching skills and independent movements but allows for less dependence on additional aid the sighted guides provide. Ways to teach skills primarily involve verbally describing objects and their physical construction, then having the visually impaired individual to explore the object and similar objects and comparing the two. (Martinez, 1998).

The type of senses that help blind or visually impaired persons understand orientation and space is tactile feedback, which is information that can be directly associated with physical objects or properties (for example, metal vibrating when struck) (Aggius-Vella et al., 2019). It is important to highlight and focus on describing objects with tactile feedback and complex physical properties because it allows for blind or visually impaired persons to sense intricate characteristics of an object that allow them to distinguish more than its visual properties. A good example of an object with complex physical properties is a seashell due to its ridges with both symmetric and asymmetric properties. Exploring objects allows visually impaired people, especially those who have had lessened sight since birth, to make connections and develop methods to understand/explore new concepts. However, using the method of comparisons only works for static physical environments, not dynamic, or public environments where people and objects constantly change physical positions (Aggius-Vella et al., 2019). Blind and visually impaired persons need to rely on other senses such as sound to understand how the environment around them is changing without colliding with objects.

Currently, no reliable methods work for understanding changing environments, hence the only way to teach navigating through people is by experience. However, using urban environments to teach independent movements and wayfinding strategies is ideal due to the rich sensory environment. Though it is not a perfect solution to object avoidance, it is a great environment to develop necessary navigational skills (Kan Kilic, 2016). When verbally explaining such an environment, the blind or visually impaired person will often make connections between their sensory perceptions and the environment. For example: if they are in a hallway or corridor, they may observe hearing footsteps clearly, while in a courtyard they may observe wind blowing due to the open space (Kan Kilic, 2016).

Although these teaching tools and resources may be prominent in America, in Russia “approximately two percent of ordinary Russian schools are prepared to educate disabled students" (Leyshon, 2012). While the Russian Ministry of Education allocates some of its funds to make schools more accessible, most visually impaired students must be taught critical navigation, orientation, and spatial skills from their parents. In Russia, braille markers are uncommon in public and few spaces offer handicap accessibility like elevators, making navigation difficult for blind or visually impaired persons. Furthermore, while navigation technologies exist for visually blind or visually impaired persons, they tend to be bulky and expensive, so there is a need to find an easily mobile and cost-effective solution to navigating environments (Leyshon, 2012).

2.4 Existing Navigation Systems

There has been a large amount of research conducted into technical solutions that address the issue of blind independence. As our ability to process and quantify the world around us continually improves, so does our ability to develop intuitive and accessible technical solutions. Explored solutions include wearable sensing hardware, fixed cameras, virtual reality mapping, and even Bluetooth proximity sensing. However, all these solutions fall short in two important categories: cost and accessibility. Hardware dependent fixed designs are the most explored solution for indoor navigation. However, these designs are often expensive, and require a great deal of maintenance to keep running smoothly. Indoor navigation devices indicate a need for a less expensive, mobile solution that would better address the needs of both the blind and the institution of which they belong. All technical solutions involve both hardware and software considerations. Some previously researched software platforms include Android, IOS, Linux, Windows, and hobbyist platforms such as Raspberry Pi and Arduino. However, Android and IOS operating systems appear to be the most easily accessible and reliable among the previous software list both in the United States and in Russia.

The most implemented solution for indoor navigation involves scattering low power Bluetooth beacons, as seen in Figure 2, through a building and association each beacon with a position in that building’s map. A user’s Android or iOS device can estimate its position by comparing the signal strengths of nearby beacons. The beacon system has been used for helping blind people navigate by the likes of Wayfindr and BlindMaps already to great success.

Figure 2: Bluetooth Beacon

("beacons" by jnxyz, licensed under CC BY 2.0)

Smartphone based indoor navigation can also function without Bluetooth beacons if there are many WiFi access points throughout the space (the service Canoe is a good example of WiFi-only navigation). In a similar way to the beacon-based navigation system, the phone estimates its location based on the signal strengths of nearby WiFi access points. Typically, because WiFi access points, such as the one in Figure 3, are placed far apart (with only very large rooms having more than one), the user's device needs to use its compass and accelerometer to narrow down possible locations of the user. A WiFi based system would have varying effectiveness based on building layout and the accuracy of the sensors onboard the Android or iOS device. (Dong et al., 2017).

Figure 3: WiFi Router

("Wifi router" by nicolasnova, licensed under CC BY-NC 2.0)

In the previous two navigation methods, the user's device communicates with base stations (Bluetooth, WiFi, or otherwise) to estimate its position. In virtual reality and augmented reality applications, a technology called "inside out" tracking has been gaining a foothold. These systems, such as the one in Figure 4, work by using an array of cameras to track location relative to surroundings as well as orientation in those surroundings. Camera vision systems are typically on a worn device like the oculus quest below but can also function on simpler devices like cell phones. These systems are expensive and complex to implement but are capable of delivering centimeter accuracy with imperceptible latency. These systems would not be cost effective for helping blind people navigating everyday scenarios but could be implemented in specific situations where speed and accuracy are paramount (Silva et al., 2015).

Figure 4: Example of Computer Vision Tracking Hardware

("Oculus Quest" by Stratageme.com, licensed under CC BY NC-SA 2.0)

Unlike the previously discussed solutions which depend on stationary beacons or WiFi access points, mobile robot technology, such as the one in Figure 5, provides a more maneuverable and secure option for navigation assistance. Currently, blind children in Russia are dependent on so-called “seeing adults”. Autonomous or remote-controlled robot systems seek to perform the same job as a human assistant would. Robots for autonomous applications are outfitted with facial recognition software for human identification, proximity sensing for obstacle avoidance, and speakers to provide audio alerts. Despite the technological advantages, robotic navigation systems are one of the most expensive and high maintenance of existing solutions.

Figure 5: Delivery Robot Traveling on a Sidewalk

CHAPTER 3: METHODS

The goal of the project is to develop a system to help visually impaired persons of all ages and levels of impairment, navigate indoor environments safely. The navigation system includes moving through stairs, hallways, doorways and all other static and dynamic objects within the space. The final iteration of the project was designed to be universally accessible, primarily through an iOS-based smartphone application, due to advanced audio commands and controls on these devices. For the final version of the project, there are three main objectives the team addressed. The first objective was how to provide orientation for the application's users. The second objective was to determine how to train the program to avoid spatially changing objects. Finally, the third objective was to determine the kind of low latency output that will provide users with directions in real time.

The final deliverable was an iOS application that aids visually impaired people in orienting themselves within indoor locations and navigating around obstacles. In order to develop the most functional deliverable possible, the Worcester Polytechnic Institute and Financial University student teams decided to divide up the tasks. The navigational method was split into two separate parts to then be integrated into a final product at the end of the project. The Worcester Polytechnic Institute team was responsible for accessing the sensors, creating image detection software capable of navigating users around obstacles, and surveying potential users of the application (blind or visually impaired persons). The Financial University team was responsible for designing a 3D model of test buildings using a neural network and developing an iOS application. Both teams tested their respective parts for proof of concept then combined each feature together for a working model of a navigation system capable of navigating users through set building locations.

3.1 Providing Orientation to Application Users

From a technological standpoint, the most challenging problem to overcome is making the navigation system aware of the user's orientation within a building or room without the use of a GPS. The project's first objective of determining orientation is multifaceted, with many potential solutions to explore. The primary technological solution that was evaluated was the usage of an iPhone's built-in camera and sensors so that it can be interfaced with a machine learning system. High mobility and resolution are essential camera features for accomplishing position correction. Requiring high resolution from video makes the Apple iPhone camera an ideal choice for image data collection.

The smartphone application that was integrated with the sensors and provides an interface was written for iOS devices. iOS devices have fast image processing software, where Android uses image correction software, which occurs after the picture has been taken and turned into a saved file to be modified for quality. When using images to determine orientation and position relative to other objects, the image resolution is more important than image correction given the aim to provide real-time directions. iOS devices also feature more advanced built-in audio commands which can be accessed through the microphone tab on the keyboard. These commands are intended to help blind or visually impaired users when interacting with the application as they require no typing or switching applications.

The gyroscope and accelerometer sensors provided data for calculating the user's orientation within the building. For the task of calculating position, readings from the integrated phone accelerometer and gyroscope were used to understand where the person is heading with respect to the building. These position and orientation sensors use dynamic positioning in order to accurately perceive data, this will be coupled with a 3D model of the building to find orientation and the changing directions of users. The gyroscope measures angular velocity, which was used for calculating what angle the phone needs to be held at to accurately detect objects that are in front of the user. Users will be of different heights, and it is crucial for the image detection to work that any objects are mostly in the frame the camera is capturing. Geometric relations will be applied to the raw outputs of the gyroscope (angular acceleration) to determine the angle the phone is being held at, the application will then tell the user to rotate the phone up or down if objects need to be more in the frame.

3.2 Training the Program to Avoid Spatially Changing Objects

To address the second objective, a machine learning system was trained to avoid spatially changing objects which consisted of building a dataset, then training a neural network. A neural network is a machine learning tool that is a collection of functions forming a statistical model that is used to interpret a data set (see Figure 6), a process modelled after the logic of the human brain. Our project only focused on two-dimensional systems which were evaluated using raw pixel values as specific data (inputs) for the neural network. Neural network training roughly follows the following procedure: an image is broken down into individual pixels to find similar groups of curves, similar curves are grouped and assigned to layers, then layers are compiled into logical classifications. In neural network training, layers are datasets that have the similar characteristics that are going to be classified to specific items by the training process. Upon completion of the neural network training, the machine learning system only needs access to a live video feed to identify and locate objects. The machine learning system is then able to identify objects even if they are in motion.

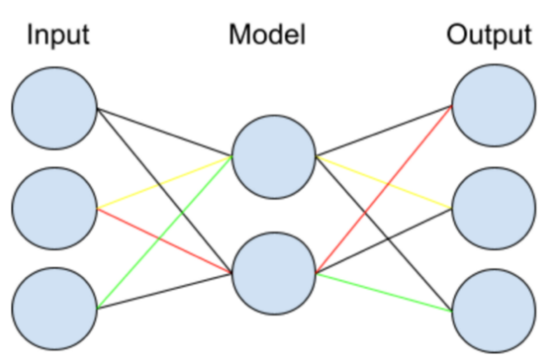

Figure 6: Simplified Diagram of how a Neural Network Functions

Each bubble represents a set of data, each line represents a function or process that is applied to the data. The Input bubbles are the images used to train the model, and the lines between the Input and Layers bubbles are the system grouping the curves of pixels. The Layers are the groups of curves that share similar features, and the lines between Layers and Output bubbles are the system compiling the Layers into classifications.

During the design and testing iterations of the project, a pre-trained neural network from an open-source project (*Computer Vision Projects*, n.d.) was used that is able to classify generic objects such as people, chairs, and tables. For this project, the neural network that was developed features specific, predetermined objects we want our users to avoid that may be commonly encountered in indoor settings. The specific objects include people, chairs, benches, stairs, doors, columns/posts, and tables. The specific list of objects (Appendix D) was determined to be those that could obstruct the user's path and cause a collision. The generic open-source list, features object classification such as windows, mirrors and paintings which did not pose a collision risk. These objects would not obstruct the user's navigation path and therefore did not need to be detected as they would have provided unnecessary feedback to the system.

The final iteration of the application used a machine learning system to accurately navigate and provide the user with object avoidance through the use of OpenCV and TensorFlow. To address the second objective of the project, OpenCV ("open source Computer Vision") was used for obstacle detection, which uses a device's camera to analyze surroundings. OpenCV works by comparing objects in an image to a list of known objects and mathematically evaluates the similarities to determine the most accurate identification of an object. OpenCV detects objects by reducing each picture into curves (as seen in Figure 7) and attempts to assign sets of curves to a group which it considers an object. TensorFlow (a machine learning framework) was used to train our software to recognize new objects and update information about existing objects. TensorFlow is a method of storage for functions that are applied to teaching machine learning (and deep learning) devices. It works by storing information in multidimensional lists of data (graphs) and comparing trends within graphs for similarities. We tested how the rate at which we captured image data affected the confidence values (seen in Figure 8) of the OpenCV platform. In the context of OpenCV, "confidence" refers to the level of similarity between the camera images and "samples" as evaluated by the computer. The higher the "confidence" value the more effective the application.

Figure 7: OpenCV Performing Background Subtracting and Showing Contour Highlights

The figure shows the OpenCV software after completing the background subtracting of a person before analyzing the contours. The image removes the walls and blank space and only highlights the person, window frame, and door.

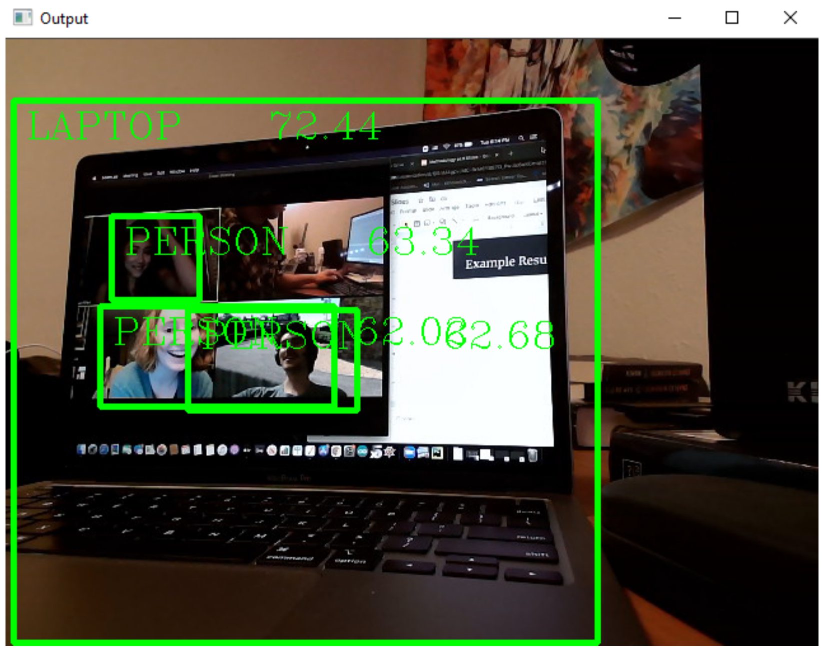

Figure 8: OpenCV Performing Object Classification and Object Highlighting with Assigned Confidence Values

The figure shows the OpenCV software classifying, and titling detected objects, listing the confidence value for each classification of an object, and using a rectangular highlight for each object. The figure shows a visual output of the object detection but does not show the coordinates of each object.

When determining the distance of an object to a user, image distortion and the area of objects in the image are critical in determining the distance. OpenCV features a command called contours which is a tool that performs shape analysis. Contours is used for determining the area of an object by taking the approximate outer curves of an object, these curves can then be bounded by length and width. Bounding an object by length and width allows the software to make an approximation of the area of an object by multiplying the length by the width (units result as pixels squared). To determine the distance of an object from a user, an object was placed at a known distance away from the camera and the software-calculated area of the object was recorded. Several areas for several objects were recorded at the same distance to effectively determine an average area for a given distance for a variety of objects. Then the area was compared to areas from previous objects to determine the change in size of an object (the distortion of the image as an object approaches or moves away from the camera). To ensure the user's safety, the area of one type of object was gathered multiple times in order to get precise results when determining the distance of an object from the user. We consider a precise result to be a distance measurement that does not deviate more than a foot from the real (or known) value.

When determining a methodology for obstacle avoidance, our team used the same logic and controls that are used for navigating robots. The basic logic was as follows: from the center of the camera if an object is to the left side the software tells the user to move to the right, and if an object is to the right side the software tells the user to move to the left. We determined that the output of the camera was 480 pixels in height and 640 pixels in width, therefore the center of the frame was at 320 pixels, so the left side of the frame was the region from 0 to 320 pixels and the right side of the frame was the region from 320 to 640 pixels. The method to achieve directional navigation was evaluated using OpenCV software. First it detected an object, then the software took the origin of the detected object and output its position as the height and width location in pixels (Height x Width). The position (in pixels) was compared to the center of the frame (320 pixels) so that the software can decide if the object lies in the left or right region of the frame to provide the user with the necessary commands to avoid the object. Then if the no object is between 200 and 400 pixels, the application tells the user to stay straight as where the user is walking has no obstructing objects. If an object takes up more than a quarter of the screen's total pixels the application tells the user to stop.

For determining the distance and navigation path for obstacle avoidance measured the assumed areas of objects in the current frame of video and compared the size of the objects to previous frames. The assumed area was an approximation of the area of the detected object. To accomplish this, OpenCV software multiplied the height by the width of the rectangle that bound the detected object. This area was calculated and output in units of pixels so that it could then be compared to the overall number of pixels being recorded in the camera's frame (480x640 pixels). If the area of a detected object was a fourth of the total area of the screen, then the user would be approximately one meter away from the camera and should turn to the left, turn to the right, or stop moving depending on other objects in the frame. The distance of one meter was determined to be a safe distance to tell the user to stop because it gives the user time to slow down before colliding with an object.

3.3 Determine Output Method

To address the final objective and output navigation directions for blind and visually impaired people in indoor spaces, we first needed to understand their perception of space. One way the understanding of the perception of space was gained was to have a blind or visually impaired person verbally describe how they perceive and sense a space while they are in it. For example, using survey questions from Appendix A: "Ask a blind or visually impaired person to navigate around a space and describe their perceptions and navigation choices out loud" or "Proximity: how does a blind or visually impaired person describe how close they are to something?" From understanding how a blind person develops spatial awareness, a cognitive map was built to get a better understanding of how a blind or visually impaired person explores and moves through static and dynamic environments. A cognitive map for blind or visually impaired people consists of determining where an object can be found in a physical plane with respect to other objects. An example of such a map may be understanding that a table is placed next to a wall and a couch is placed in the middle of a room instead of next to a wall. As opposed to focusing on visual clues such as lighting or physical measures of distance, blind or visually impaired persons may choose to focus on sounds echoing to understand the size and shape of the rooms.

Another necessary step was to determine how best to verbally describe a multidimensional environment to visual impaired persons. The questions from Appendix A were additionally helpful in providing a method of identifying keywords and phrases that would be beneficial to use in the navigation system when providing directions. Additionally, interviewing people with either direct or auxiliary experience with visual impairments provided further insight into effective aspects to use in the navigation solution. Such questions about what type of output would suit users best are included in Appendix B, for example: “Will audio commands (such as directions similar to google maps) be useful to you?” or “Would audio beeps help you avoid oncoming obstacles?” Many text to speech instructions or tones are built in features of smartphones that are accessible through the keyboard on the screen. The text to speech commands were not something the team needed to focus on developing given that it is a standard, built-in feature of iOS devices. Based on the results gained from this section it may be determined that having multiple options for what kind of feedback is used will be beneficial in the final application.

Audio commands are provided to the users in two forms: clear verbal directions and variable tones. The first form of commands came from survey results, responses indicated that the most helpful is clear directions such as ones you may get from asking a stranger for directions. Incorporating this in the application was done by recording audio clips that say, “Turn left”, “Turn right”, “Stop”, “Stay straight”, “Step to the left”, and “Step to the right”. These commands are simple enough to be understood at a variety of ages and clear enough so users will not face confusion that may cause misdirection. The second form of commands was designed with the intent of not startling users with commands like “Stop”. Soft beeping tones that increase with intensity as an object increases its relative area with respect to the camera's captured frame. Similar to how a car's backup system beeps to inform users of approaching objects, we want to inform our users of objects as they get closer to them, not right before they collide with them. To reduce the time, it takes to provide users with commands, stored audio recordings will play faster than a text to speech software and the audio recordings can be stored and called easier in the software.

CHAPTER 4: RESULTS

The goal of our project was to develop an iOS application that would be able to help visually impaired persons of all ages and levels of impairment navigate indoor environments safely. The navigation includes safely moving through stairs, hallways, doorways and any other static or dynamic obstacle within a space. It was determined early in the project that the navigation application would be iOS based due to the level of accessibility that is built into such devices to aid in usage by visually impaired individuals. Three main objectives were introduced and addressed in Chapter 3. The first objective was determining how to provide physical orientation to applications users. The second objective was to determine how a program needed to be trained to avoid spatially changing objects. The third objective was to determine what kind of low latency output would be used to provide users with directions in real time. The following chapter presents the progress made in the development of our project.

4.1 Providing Orientation to Application Users

The result of using onboard sensors accessed from the device is that the onboard sensors alone cannot accurately determine a user's orientation with respect to another position (such as a building). The device that was used was an iOS laptop because it was easy to test the application and make edits to the program on the laptop as needed during testing, but in the future the program will be on an iOS smartphone. The lack of onboard sensors on laptops is problematic because laptops do not feature gyroscopes or accelerometers so the testing of their accuracy will have to be done on a smartphone. Testing on a laptop was done as a proof of concept and to save time, developing the application to be tested for a phone would take many small updates to the application to iterate and improve upon the concept. As indicated by our sponsors, proof of concept for our timeline was more important than developing a working model of the application (this will be saved for future projects with funding), so the choice of testing the code separately was more time efficient.

After testing the raw output of a gyroscope as seen in Table 1, it was determined that the gyroscope and accelerometer, which outputs angular accelerations, are not alone enough to provide the user with orientation data. Our results showed that by moving a smartphone we are able to receive readings from X, Y, Z planes. The result showed that a user is turning at 1 degree per second squared in the horizontal plane (X plane), if angular velocity was required, we would simply integrate the angular acceleration with respect to the time recorded. Without some digital layout to compare the angular acceleration readings to, the raw data is only useful in determining how fast a user is turning and in what direction the user is turning. In order to provide users with orientation we will need to track the users position within a model of the building where we can then compare the users position and direction to where the user is trying to navigate to. Our orientation results show that we have the means to provide users with orientation, however, we have only been testing the orientation tools separately at this time. In a later version, angular acceleration data will be useful in keeping track of the user’s orientation when the camera is motion-blurred or obstructed.

Table 1: Raw Gyroscopic Readings from iPhone 12 Pro

Table of raw gyroscopic readings obtained by rotating an iPhone 12 Pro around all axes.

Pitch (X axis) (degrees/s^2)	Roll (Y axis) (degrees/s^2)	Yaw (Z axis) (degrees/s^2)
-21.38	9.862	97.75
-91.71	2.336	92.35
-157.0	-1.003	96.24
-16.19	-4.255	80.15
8.619	56.07	-36.03
50.55	6.964	-128.3
21.23	-62.62	-154.3
-45.48	-5.963	90.90

4.2 Machine Learning System for Avoiding Spatially Changing Objects

The result of using a pre-trained neural network with a developed machine learning system was an object detection system that was able to identify 91 different distinct objects (Appendix D). The preliminary results of the image detection software show that it was accurately able to identify separate objects within images and live video-feed, however, the software's confidence levels were relatively low (on average around 50-72 percent). The goal for the confidence levels was a range of 70-85 percent to ensure that the software was accurately able to identify separate objects. Improving the confidence level would allow the application to more accurately identify or classify objects. However, the results we achieved allow the application to distinguish objects in order to avoid them, for which classifying was not necessary. Not classifying objects will ultimately increase the speed of the software. In addition, our software was able to rapidly detect multiple objects once they move into the camera's frame and continue to track them as they move around the frame. Not classifying objects still allowed for the training and testing of the neural network so that it met the requirements for objects users need to avoid indoors. The importance of being able to track moving objects quickly allows users to safely move around and be able to rely on the application for object avoidance.

The only concern as a result of testing was that the camera solution only provides image detection for frontal obstacles. Additionally, the camera solution was fine for stationary objects, however, objects that were moving toward the user from the side or behind were undetected. However, non-frontal objects are no different than non-visually impaired persons facing in a singular direction. To detect objects coming from non-frontal angles, the user will have to turn the smartphone similar to how a person would have to turn their head and use other senses such as hearing to detect other obstacles. The image detection software is able to near instantaneously detect new objects as the camera panes, so the user will always have some sort of direction to move in.

Determining the center of the frame for navigation was a result of taking the center of the two-dimensional frame that the camera outputs. The camera outputs a video frame of 480 pixels tall by 640 pixels wide, so we are able to determine that the center of the screen is 480/2 pixels and 640/2 pixels. For our navigational method we were primarily concerned with the center width which was determined to be 320 pixels as the center allowed the software to determine if an object was to the left or right of the camera. In testing, the software was successful in reporting whether the user needed to move to the left, right or stay moving straight depending on where detected objects were located in the camera frame. If an object was detected in the right frame (between 0 and 320 pixels) the software output that the user needed to move to the left. If an object was detected in the left frame (between 320 and 640 pixels) the software output that the user needed to move to the right. If no objects were detected on the screen, then the software reported that the user could continue moving straight. One issue that was identified was that static or non-moving objects were not always detected as quickly as moving objects were. Static objects were more difficult to detect by the software because the background subtracting process done by OpenCV tracked objects with changing curves faster than curves that were not moving. To solve the static object detection problem two separate methods of OpenCV were used: one method that features object tracking to detect moving objects and one method that features object classification to detect static objects (Figure 9).

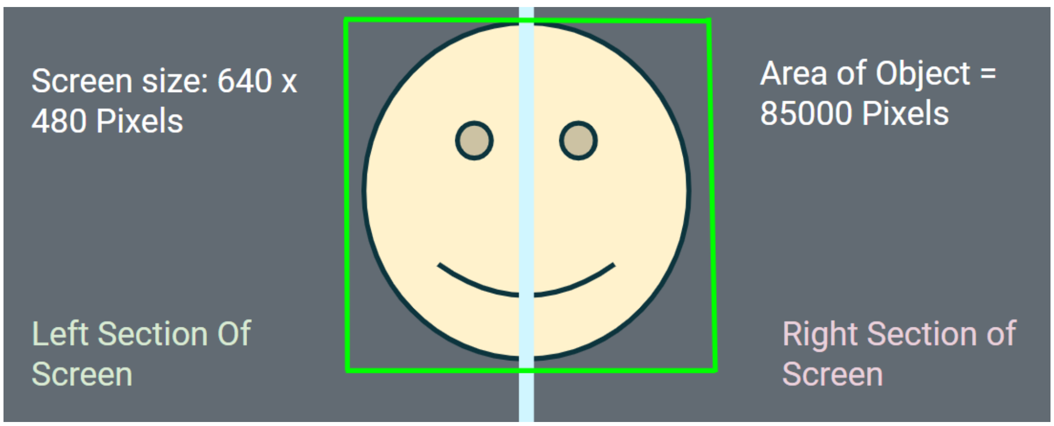

Figure 9: Diagram of Directional Decision-Making Process done by Obstacle Avoidance Program

When determining the distance of objects from the users (camera) the test distance was set to three feet, which was decided as a safe distance for a user to stop before colliding with an object. Tested objects were chairs, people, trash cans, and tables then smaller objects like a phone to determine a range of areas for objects that the user may collide with. The range of areas was determined to be 60,000 - 90,000 pixels squared for objects placed three feet away (which is approximately a quarter of the overall area of the screen). In the software, if the area is less than 60,000 pixels squared, the software gives the user navigational directions such as straight, right, and left. Then when the software receives an area greater than 60,000 pixels squared the software tells the user to stop due to an object being close to the user.

4.3 Chosen Output Method

Initial testing of the application for obstacle avoidance used a text to speech output method. Based on obstacles that were detected by the program, it would inform the user to to move to the left, to the right or stop. The lateral navigation output method was chosen as it was the simplest to accomplish and allowed the team to perform tests with the application that addressed the actual functionality of the code. Furthermore, in our testing of the commands, holding the phone or camera at a constant height and angle, the software was not able to detect some objects due to parts of them being cropped out (Figure 11 and Figure 12). In changing the angle the phone or camera was held, the software immediately would detect the object when it was entirely in the captured video frame (Figure 10). The initial implementation of a gyroscope was to determine orientation, however, an additional usage for the gyroscope will be to correct the angle the phone or camera is being held in order to have objects completely within the video frame (Figure 11 and Figure 12).

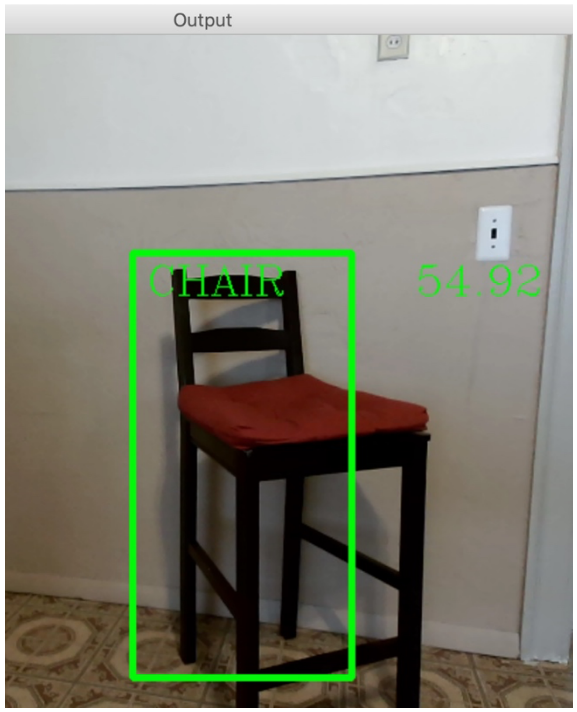

Figure 10: OpenCV Accurately Detecting and Classifying a Chair

Displays the software correctly detecting a chair. This allows for the software to properly provide the user with directions to move around the chair.

Figure 11: OpenCV Inaccurately Detecting No Objects in the Frame Due to Low Angle

Displays the software not being able to detect a chair due to the camera angle being too low. This indicates the need to implement and angle correction if no objects are detected on the screen.

Figure 12: OpenCV Inaccurately Detecting No Objects in the Frame Due to High Camera Angle

Displays the software not being able to detect a chair due to the camera angle being too high. This indicates the need to implement and angle correction if no objects are detected on the screen.

In testing the navigational capabilities of our application, aside from the camera angle errors, when the objects were all in frame, the application was able to accurately navigate users around objects. The physical environment also demonstrated that changing lightness between rooms is not a problem assuming there is not extreme lens glare or pitch darkness. One piece of feedback from the developers testing the initial navigational commands was the statement “Move left” and “Move right” was confusing if the user should step or turn to the left or right. In addition, testers noted that it was confusing how far or for how long they should step, and additional tone or countdown may prove helpful in determining the distance they would need to move for a particular command. With respect to the frequency of commands, testers mentioned that they only wanted commands when it was time to change direction or to stop moving. Testers mentioned that continuous cues (such as several “Stay Straight” commands) were overbearing and confused users more than helped them navigate. Testing the speech-to-text commands worked well for very specific commands with exact wording, such as: if the software was expecting the user to say, “Start the Navigation Route” and the user only said, “Start Navigation Route”, the software would not understand the request. The Python SpeechRecognition requires such specific phrases it makes it impractical for some users due to the low margin of deviation from expected results for the software. The direct result of using the SpeechRecognition was that there is a need for a more advanced speech library, particularly one that uses an AI such as a Google or iOS speech library to understand commands.

Determining the distance of objects from the user will need to be calibrated to ensure safety of the user. A focus of the testing or development of the software will be to determine a safe distance to notify the user of an object nearing them in order to give the user enough time to react and avoid the object. A problem is multiple objects in a cluster (such as a crowd of people) that could provide multiple of the same signal; the notifications to the user may become confusing if overloaded. From earlier in our results, we know that a quarter of the screen is roughly equal to 80,000 pixels, and in the physical dimension, an object with an area of 80,000 pixels appearing on camera is between 1 and 1.5 meters away from the camera (away from the user's phone in the case of the application). Some deviation of the area occurred when the object was moving laterally from the user (not towards or away) particularly arms and legs swinging were difficult for the application to register as one whole object. From these results, it was concluded that any object with an area of 80,000 pixels or greater should be considered close to the user, and the application should tell the user to stop moving.

4.3.1 Survey Results

From our surveys, which were administered by our social sponsor in Russia to two participants, and translated by a professor at WPI (Appendix C), gave our team insight on the current navigation technique and challenges of visually impaired people. Additionally, we gain some insight from these surveys on the current attitude towards such a navigation app as well as recommendations for how output might be relayed. Participates were first asked about their current navigation techniques. Some said that it was helpful for them to locate the walls in a room in order to guide them as well as to create an idea of the layout of the environment. Participants mentioned that walls and other notable parts of a room such as furniture and doorways were used to study routes so that they could more confidently move through a space. Additionally, reverberations were frequently mentioned as a way to get a sense of a room and what is around them. One survey participant noted that they wear high heels in order to produce a louder sound to reverberate. Finally, when other techniques don’t work, many participants said they would ask people with uncorrected vision around them for help.

Next, participants were asked about their current orientation and navigation challenges. From these responses our team learned two main things. First was that rooms that did not produce reverberations were difficult to navigate, such as those with a lot of carpeting. Second, noisey environments, such as those around a noisy street or construction site, produced a similar issue in which relevant reverberations cannot be heard. Finally, participants responded to the idea of our application and how the team can make it useful to them. One participant mentioned the importance of clear, concise commands. It was noted as well that these commands should not be presented too quickly or harshly to avoid startling users. There was a mixed response from the participants about whether or not they would be willing to try a new technological solution. One said they were opened to trying a new technological solution, while the other mentioned a distrust with technical solutions due to bad past experiences. The participant with distrust towards navigation applications says they found the applications gave them incorrect or inaccurate directions that caused them to fall, collide with obstacles, or go in the wrong direction. These experiences emphasize the importance of keeping navigation users safe while providing them with accurate directions.

CHAPTER 5: DISCUSSION AND RECOMMENDATIONS

From our results section, we are able to conclude that we were able to develop an effective proof of concept that can navigate users efficiently and safely. However, the user interface of our software is still required to make the project fully functional and launch a full-scale version of the application. The Discussion and Recommendations following provide details on how to better improve the Navigator for the Blind application which can hopefully be implemented in the future.

5.1 Understanding the Problem

While testing the accuracy and overall functionality of the navigation tool that was developed for object detection, it was discovered that in using OpenCV some obstacles that can not be identified or classified. Some of these obstacles include walls, doorways, and some common household appliances. Objects that are not classified pose a danger in non-controlled environments because the software may not be able to detect obstacles that the user could run into. Factors that may cause the software to not recognize an object are poor lighting, too high of a confidence threshold, fast movements of the camera, and objects that are untrained. When testing the navigation and image detection software it was noticed that some objects would be classified incorrectly, an example of this is a phone being classified as a remote. Objects that are classified incorrectly should not be considered a problem for the navigation software, because the purpose of using the image detection software is to find where obstacles are located, not what the obstacles are. To move the user around the obstacle, we only need the obstacle's location with respect to the camera and user.

We determined that iOS was the best platform for our prototype, but in the future, we will want to reach a wider audience than just apple users. Supporting a finished version of the application to Android OS in the future will be an important step if we want it to be adopted by a general audience. The Android ecosystem has many equivalent features to it's iOS counterpart, but Android hardware is much more varied. With Android devices featuring less developed audio commands, developer softwares such as Apple Developer or Android developer softwares will not be useful in supporting audio commands. Another Android problem is the lower video quality which reduces the image detection softwares ability to accurately detect objects. Android commonly features blurry videos and images with software corrections, the lower quality images cause the software to be less reliable and increase the margin for error that could impose risk to users.

At present, our obstacle avoidance software only uses simple logic to determine if an obstacle is in the path of the user. The user experience should be improved in future versions by taking advantage of more iOS capabilities. When determining the obstacle avoidance route, a goal was to keep the processing time as low as possible. Most methods for measuring distance can only be done quickly for pictures due to complex geometric relations, the video feed would require longer processing times due to the constantly changing dimensions.

Due to time and location constraints the testing of the navigation system was only conducted in a select few environments. Those environments were primarily restricted to apartment and university facilities. It is believed that in order to properly evaluate the universal applicability of this navigation tool a more comprehensive test would be required. The testing environments that have been used to evaluate the navigation tool feature ideal environments with predictable settings. Predictable meaning the environments are not changed unless done so by the testers, for example: the lighting remained constant, and persons or objects did not randomly enter the test environment. In order to declare the application safe to use for common life or released to the public, testing should be done in non-simulated environments in order to test with random obstacles.

As stated in the results section, it was determined that utilizing Text-to-Speech as a method of user notification is ineffective for our implementation. Text-to-Speech was determined ineffective for our application because Text-to-Speech does not generate audio fast enough to provide adequate notice of an obstruction. While Text-to-Speech can be easily translated between languages with software commands, the translation (if different from the original language) will take time and require trained language libraries. Our goal is to have a near instantaneous response from the time the object is detected, a benchmark for ideal response time would be a similar time the software takes to produce text responses from Text-to-Text outputs.

In reference to the previous section, relaying too much information to the user could decrease the usability of the navigation system. Many environments that the application would be implemented in will be constantly changing and produce frequent changes in commands. As a response from our surveys indicated, it is easy for blind or visually impaired persons to be overwhelmed in complex environments (such as a city), and constant commands could easily overwhelm application users. A large portion of the testing done on our application was to determine the fastest rate we could deliver an output to the user so they could safely navigate through obstacles without confusion from rapid streams of commands.

For navigation applications such as Google Maps users must click a “Start Route" button to begin getting directions from the application. To optimize the ease of use of the application for our blind or visually impaired users, using audio commands to start a navigation route or obstacle avoidance route would decrease the reliance on clicking buttons to make the application run. Where our users are blind or visually impaired, any on-screen input such as buttons or text boxes are relatively ineffective due to smartphone screens being smooth and without reference keys (such as F and J keys on a keyboard). Having the application commands or controls be set as on-screen buttons increases the chance of user error if our user is not able to understand where the buttons are on the screen.

5.2 Technical Development Considerations

Identification of obstacles that are not currently recognized by the software could be achieved by adding additional reference images to the image library. When training a neural network, for each object (formally called a class), one thousand reference images is an optimal goal, with one hundred reference images being a bare minimum to train a neural network for image detection. Accuracy of this identification could be improved by acquiring a greater amount of reference images in different environments. For example, to better detect doorways a collection of doorway images of different colors and sizes would be optimal. An alternative methodology could be to lower the confidence threshold (the value that determines if the software is able to identify a specific object due to reference images) so that the software will detect objects, but they may be inaccurately identified. As mentioned earlier, incorrectly identifying the objects is not a concern, where the goal of the project is to move users away from objects, not tell the user what the object is.

If the application is ported to Android directly, it will likely have bad performance on some devices. To mitigate this risk, the application should not be ported to Android until it is solidly functional with the full range of desired features implemented. A proposed solution to Androids less advanced audio commands could be solved by using a third-party audio recognition such as a Google Speech-to-Text API for translating the microphone inputs from the user. The lower camera quality could be solved by using an additional webcam with high resolution; however, this requires the user to spend money, lowers the user's mobility, and potentially causes hardware interfacing issues with the software.

AVdepth data could be used to more accurately determine the distance of obstacles in future versions. There are also a number of tools included in the ARkit library for analysing a user's environment. A final version of this application should be able to simplify complicated dynamic environments such as a campus courtyard or city street so that the user receives only the information they need, and no extraneous instructions. Android devices have other methods for determining depth that could be useful in the Android port of a final application. However, testing iOS AVdepth or similar Android distance measurement softwares should be done to determine the time it takes the software to function and measure the distances. If the software is tested and the results show a long response time, then an alternative solution for measuring distance should be used.

Testing the application for safety and confidence would ideally incorporate a variety of different indoor testing environments including but not limited to stores, malls, and train stations. In addition, a more comprehensive test would include the random placement of both static and dynamic obstacles within each operating environment. However, testing the application should only involve developers until the developers are certain that the application is refined to the point where test users are safe and will not be subject to any collisions. Testing of the application with members that are not on the developmental team should only be used for user interfacing and satisfaction with the application. Testing the actual effectiveness of the navigation route could put users at risk, so it is best when the developmental team is in full control over the testing environment and ensures that conditions are ideal.

An alternative method for user notification that was determined to be more effective is the use of pre-recorded audio files. These audio files contain navigation instructions that are required for successful obstacle avoidance. It was discovered that this method of providing audio feedback allowed instructions to be relayed to the user at a much faster rate, thus improving the overall functionality of the navigation system. While recording audio feedback does not allow for the text to be translated to other languages, adding additional language features would be done by recording the assorted audio feedback. Toggling languages could then be done by some sort of tab feature which controls the language of the output. Having the option for non-speech feedback (audio tones and buzzes) is also a feature that might benefit the user.

Studying the rate at which humans can understand and process instructions can have a positive impact on the functionality of the navigation system. In addition, an understanding of psychological appeals such as tone and speed of speech could prove beneficial when designing commands to users. It is important not to startle the user with commands, however, it is crucial that the timing of the command is accurate. If this criterion is not met, it could put the users in harm's way. Conducting research on how commands are best received by individuals will provide developers with a way to improve the user experience and satisfaction with the application and navigation system. Making the voice commands more enjoyable such as having calming voices and clear commands will make the use of the application more appealing.

Developing a feature to use audio commands would require a speech-to-text function which translates input from the microphone of the application (audio waves) to text. Speech-to-text functions would then allow the software to take input from the user's voice and perform the requested action instead of having the user type or click buttons. Audio commands could be developed in coding languages such as Python or Apple Developers software depending on what stage of testing the audio commands are being implemented in (testing audio commands for effectiveness or integrating the audio commands into the application). Adding the audio commands will become part of the user interface not the navigation software, adding the audio commands should not hinder the applications speed or performance.

5.3 Recommendations

The following represent recommendations based on results from developing the Navigator for the Blind application. These recommendations are intended to aid future groups who are looking to develop this project further.

1. Expand the number of objects that can be detected by adding additional reference images.
2. In order to make this tool more accessible to a larger group of individuals, it is recommended that the code be modified to be compatible with both Android and IOS devices.
3. Utilize more of iOS's capabilities for more effective navigation (AVdepth, ARkit, Etc.)
4. Test navigation system in different indoor environments to better understand accuracy of static and dynamic obstacle detection and avoidance.
5. Make audio instruction more efficient and more specific by combining navigation and obstacle avoidance instructions.
6. Study audio instruction psychology in order to understand how commands will be best received by users.
7. Develop a system for audio commands for the user to give the application ("Hey Navigator" type system.)

REFERENCES

2018_SHS_Table_A-6.pdf. (n.d.). Retrieved April 2, 2021, from https://ftp.cdc.gov/pub/Health_Statistics/NCHS/NHIS/SHS/2018_SHS_Table_A-6.pdf

Aggius-Vella, E., Campus, C., Kolarik, A. J., & Gori, M. (2019). The Role of Visual Experience in Auditory Space Perception around the Legs. *Scientific Reports*, *9*(1), 10992. https://doi.org/10.1038/s41598-019-47410-2

Alghamdi, S., Schyndel, R.v., Khalil, I., Accurate positioning using long range active RFID technology to assist visually impaired people, Journal of Network and Computer Applications, Volume 41, 2014, Pages 135-147, ISSN 1084-8045, https://doi.org/10.1016/j.jnca.2013.10.015.

Andreas Hub, Joachim Diepstraten, and Thomas Ertl. 2003. Design and development of an indoor navigation and object identification system for the blind. SIGACCESS Access. Comput., 77–78 (Sept. 2003 - Jan. 2004), 147–152. DOI:https://doi.org/10.1145/1029014.1028657

Bekkelien, A., Bluetooth indoor positioning.” *University of Geneva*, Mar. 2012, cui.unige.ch/~deriazm/masters/bekkelien/Bekkelien_Master_Thesis.pdf

Beteinaki, E., Bertills, K., & Hugh, E. M. (n.d.). *Social Interactions and Friendships of adolescents with vision impairments:* 55.

Bharathi, S., Ramesh, A., Vivek, S., "Effective navigation for visually impaired by wearable obstacle avoidance system." *2012 International Conference on Computing, Electronics and Electrical Technologies (ICCEET)*. IEEE, 2012

Bikbov, M. M., Kazakbaeva, G. M., Zainullin, R. M., Gilmanshin, T. R., Nuriev, I. F., Zaynetdinov, A. F., Yakupova, D. F., Uzianbaeva, Y. V., Panda-Jonas, S., Mukhamadieva, S. R., Khikmatullin, R. I., Aminev, S. K., Arslangareeva, I. I., & Jonas, J. B. (2020). Prevalence and causes of vision impairment and blindness in the Russian ural eye and medical study. *Scientific Reports*, *10*(1), 12397. https://doi.org/10.1038/s41598-020-69439-4

Blewett, Lynn A., Drew, Julia A. Rivera, King, Miriam L., & Williams, Kari C.W. (2019). *IPUMS Health Surveys: National Health Interview Survey, Version 6.4* (6.4) [Data set]. Minneapolis, MN: IPUMS. https://doi.org/10.18128/D070.V6.4

B. Mustapha, A. Zayegh and R. K. Begg, (2013) Ultrasonic and infrared sensors performance in a wireless obstacle detection system, 2013 1st International Conference on Artificial Intelligence, Modelling and Simulation, Kota Kinabalu, 2013, pp. 487-492, doi: 10.1109/AIMS.2013.89.

Bureau, U. C. (n.d.). *Age and Sex Composition in the United States: 2015*. The United States Census Bureau. Retrieved March 16, 2021, from https://www.census.gov/data/tables/2015/demo/age-and-sex/2015-age-sex-composition.html

Canu, S. (2021, January 28). Object Tracking with Opencv and Python. Pysource. https://pysource.com/2021/01/28/object-tracking-with-opencv-and-python/

Challenges blind people face when living life. (2019, April 15). https://undefined/blog/challenges-blind-people-face-when-living-life

Computer Vision Projects. (n.d.). MURTAZA'S WORKSHOP. Retrieved April 29, 2021, from https://www.murtazahassan.com/courses/opencv-projects/

Coughlan J, Manduchi R (2009) Functional assessment of a camera phone-based wayfinding system operated by blind and visually impaired users. Int J Artif Intell Tools 18: 379-397. doi:https://doi.org/10.1142/S0218213009000196

C. S. Silva and P. Wimalaratne, "State-of-art-in-indoor navigation and positioning of visually impaired and blind," 2017 Seventeenth International Conference on Advances in ICT for Emerging Regions (ICTer), Colombo, 2017, pp. 1-6, doi: 10.1109/ICTER.2017.8257826.

Cuturi, L. F, Aggius-Vella, E., Campus, C., Parmiggiani, A., Gori, M., From science to technology: Orientation and mobility in blind children and adults, Neuroscience & Biobehavioral Reviews, Volume 71, 2016, Pages 240-251, ISSN 0149 7634, https://doi.org/10.1016/j.neubiorev.2016.08.019

Dakopoulos, D., Bourbakis, N., G., (2009) "Wearable obstacle avoidance electronic travel aids for blind: a survey." *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)* 40.1: 25-35

Dharani, P., Thomas, D., & Lipson, B. (2012). RFID Navigation System for the Visually Impaired.: Worcester Polytechnic Institute.

Documentary: Russia's Village Of The Blind. (2017, April 16). https://www.rferl.org/a/russia-blind-city-rusinovo/28425942.html

Dong, K., Wu, W., Ye, H., Yang, M., Ling, Z., & Yu, W. (2017). Canoe: An Autonomous Infrastructure-Free Indoor Navigation System. *Sensors*, *17*(5), 996. https://doi.org/10.3390/s17050996

Drever, J. (1955). Early Learning and the Perception of Space. *The American Journal of Psychology, 68*(4), 605-614. doi:10.2307/1418788

Eye Health Statistics—American Academy of Ophthalmology. (n.d.). Retrieved March 16, 2021, from https://www.aao.org/newsroom/eye-health-statistics

Fadell, A. M., Pantfoerder, A., "Integrated proximity sensor and light sensor." U.S. Patent No. 7,714,265. 11 May 2010.

Fraser, S., Beeman, I., Southall, K., & Wittich, W. (2019). Stereotyping as a barrier to the social participation of older adults with low vision: A qualitative focus group study. *BMJ Open*, *9*(9), e029940. https://doi.org/10.1136/bmjopen-2019-029940

Giudice, N. A. (n.d.). Navigating without vision: Principles of blind spatial cognition. Retrieved February 03, 2021, https://umaine.edu/vemi/wp-content/uploads/sites/220/2018/02/Giudice-in-press-ch15-in-Montello-principles-of-blind-navigation.pdf

Giudice, N. A., & Legge, G. E. (2008). Blind navigation and the role of technology. In A. Helal, M. Mokhtari & B. Abdulrazak (Eds.), Engineering handbook of smart technology for aging, disability, and independence (pp. 479- 500): John Wiley & Sons.

Gori, M., Cappagli, G., Baud-Bovy, G., & Finocchietti, S. (2017). Shape Perception and Navigation in Blind Adults. *Frontiers in Psychology*, *8*. https://doi.org/10.3389/fpsyg.2017.00010

Harris, J. (2017, April 5). Schools and agencies for the blind in each state. *Council of Schools and Services for the Blind*. https://cosbvi.org/schools-for-the-blind/

Holland, S., Morse, D. & Gedenryd, H. (2002) AudioGPS: Spatial audio navigation with a minimal attention interface. *Personal Ub Comp* 6, 253–259. https://doi.org/10.1007/s007790200025

ICEVI-Europe—National pages—Russia. (n.d.). Retrieved March 10, 2021, from http://icevi-europe.org/russia.php

Ji, M., Kim, J., Jeon, J., & Cho, Y. (2015). Analysis of positioning accuracy corresponding to the number of BLE beacons in indoor positioning system. *2015 17th International Conference on Advanced Communication Technology (ICACT)*, 92–95. https://doi.org/10.1109/ICACT.2015.7224764

Kan Kilic, D. (2016, August 8). Seeing the Environment from the Eye of Blinds. *11th EAD Conference Proceedings: The Value of Design Research*. European Academy of Design Conference Proceedings 2015. https://doi.org/10.7190/ead/2015/138

Khatib, O., "Real-time obstacle avoidance for manipulators and mobile robots." *Autonomous robot vehicles*. Springer, New York, NY, 1986. 396-404

Legge GE, Beckmann PJ, Tjan BS, Havey G, Kramer K, Rolkosky D, et al. (2013) Indoor navigation by people with visual impairment using a digital sign system. PLoS ONE 8(10): e76783. https://doi.org/10.1371/journal.pone.0076783

Leyshon, R. (2012). *The invisible ones: Growing up blind in Russia—Matador Network*. https://matadornetwork.com/abroad/blind-in-russia/

Li, Y., Zhuang, Y., Zhang, P., Lan, H., Niu, X., & El-Sheimy, N. (2017). An improved inertial/wifi/magnetic fusion structure for indoor navigation. *Information Fusion*, *34*, 101–119. https://doi.org/10.1016/j.inffus.2016.06.004

Martinez, C. (1998). *Orientation and Mobility Training: The Way to Go*. https://www.tsbvi.edu/seehear/fall98/waytogo.htm

Morris, S., Perry, B., & Carcieri, S. (2009). RFID Technology to Aid in Navigation and Organization for the Blind and Partially Sighted. Copenhagen: Worcester Polytechnic Institute.

Nosenko-Stein, E. (2017). “Nos temen porque somos Otro”: Actitudes hacia las personas con discapacidad en la Rusia moderna. *Salud Colectiva*, *13*(2), 157–170. https://doi.org/10.18294/sc.2017.1140

Orientation and Mobility. (n.d.). Retrieved February 03, 2021, http://blindchildrensfund.org/for-families/orientation-and-mobility/

Plikynas, D., Žvironas, A., Budrionis, A., & Gudauskis, M. (2020). Indoor Navigation Systems for Visually Impaired Persons: Mapping the Features of Existing Technologies to User Needs. Sensors (Basel, Switzerland), 20(3), 636. https://doi.org/10.3390/s20030636

Python: Convert Speech to text and text to Speech. (2019, December 16). GeeksforGeeks. https://www.geeksforgeeks.org/python-convert-speech-to-text-and-text-to-speech/

Real, S., & Araujo, A. (2019). Navigation systems for the blind and visually impaired: Past work, challenges, and open problems. Sensors (Basel, Switzerland), 19(15), 3404. https://doi.org/10.3390/s19153404

Ruxandra Tapu, Bogdan Mocanu, Titus Zaharia,Wearable assistive devices for visually impaired: A state of the art survey, pattern recognition letters, Volume 137, 2020, Pages 37-52, ISSN 0167 8655, https://doi.org/10.1016/j.patrec.2018.10.031.

Samoshin, D., & Istratov, R. (2014). The Characteristics of Blind and Visually Impaired People Evacuation in Case of Fire. *Fire Safety Science*, *11*, 1160–1169. https://doi.org/10.3801/IAFSS.FSS.11-1160

Sanchez, J., Tadres, A., Pascual-Leone, A., & Merabet, L. (2009). Blind children navigation through gaming and associated brain plasticity. *2009 Virtual Rehabilitation International Conference*, 29–36. https://doi.org/10.1109/ICVR.2009.5174201

Satan, A. (2018). Bluetooth-based indoor navigation mobile system. *2018 19th International Carpathian Control Conference (ICCC)*, 332–337. https://doi.org/10.1109/CarpathianCC.2018.8399651

Schinazi, V. R., Thrash, T., & Chebat, D. R. (2016). Spatial navigation by congenitally blind individuals. Wiley interdisciplinary reviews. Cognitive science, 7(1), 37–58. https://doi.org/10.1002/wcs.1375

Silva, A. R., Valente, L., Clua, E., & Feijó, B. (2015). An Indoor Navigation System for Live-Action Virtual Reality Games. *2015 14th Brazilian Symposium on Computer Games and Digital Entertainment (SBGames)*, 1–10. https://doi.org/10.1109/SBGames.2015.29

Teaching blind children to navigate. (n.d.). Retrieved February 03, 2021, http://www.wayfinding.net/navigate.htm

Thinus-Blanc, C., & Gaunet, F. (1997, November 1). Representation of Space in Blind Persons: Vision as a Spatial Sense? Retrieved February 06, 2021, from https://pdfs.semanticscholar.org/0c9f/5342c44bf0fa6419d27f68c2d49a4188b5a3.pdf

VernierST/godirect-examples. (n.d.). GitHub. Retrieved May 11, 2021, from https://github.com/VernierST/godirect-examples

Vleuten, M. S., Andrew Spitz, Ruben van der. (n.d.). *BlindMaps*. BlindMaps. Retrieved April 28, 2021, from http://www.blindmaps.org

Voss, P. (2016, November 30). Auditory spatial perception without vision. Retrieved February 07, 2021, from https://www.frontiersin.org/articles/10.3389/fpsyg.2016.01960/full

Wayfindr-Open-Standard-Rec-2.0.pdf. (n.d.). Retrieved March 1, 2021, from http://www.wayfindr.net/wp-content/uploads/2018/07/Wayfindr-Open-Standard-Rec-2.0.pdf

Wilson, J., Walker, B. N., Lindsay, J., Cambias, C., Dellaert, F., (2007) "SWAN: System for Wearable Audio Navigation," *2007 11th IEEE International Symposium on Wearable Computers*, Boston, MA, pp. 91-98, doi: 10.1109/ISWC.2007.4373786

APPENDIX A: SURVEY QUESTIONS FOR USEFUL VOCABULARY

The following questions will help us understand how blind people experience the world around them. Respond to these prompts as best you can in a sentence or two. Any additional information about your experience is always appreciated.

1. What are the first things you perceive when entering a new and unfamiliar space?
2. How do you determine how close they are to an object in a familiar space?
3. How do you determine how you are positioned in a familiar environment?
4. What sense do you find most useful when exploring new environments?
5. What sense do you find most distracting when exploring new environments?
6. What clues from senses do you use to understand the environment around you (such as does the sound of your footstep allow you to determine the size of the room around you)?
7. How do dynamic elements in the environment affect your ability to successfully navigate?

APPENDIX B: SURVEY QUESTIONS FOR DETERMINING EFFECTIVE COMMUNICATION OF NAVIGATION INFORMATION

We are developing a navigation app to assist visually impaired people. The following questions will inform the development of this application. Answer the prompts as best you can.

1. What kind of output do you feel would provide the quickest and most understandable directions in a dynamic situation?
   i. Audio commands such as directions read out loud
   ii. Morse-code like vibrations
   iii. Analog beeps of varying loudness or lengths
2. Will audio directions (such as directions similar to google maps) be useful to you?
3. Would audio beeps help you avoid oncoming obstacles? What situations would an audio alert of this kind be useful to you?
4. Would cell phone vibrations help you avoid oncoming obstacles? What situations would an audio alert of this kind be useful to you?

APPENDIX C: NOTES FROM SURVEY AUDIO RESPONSES

Notes transcribed from audio survey responses, translated to English from Russian by Professor Nikitina.

- "How do you orient yourself in a space"
    - She approaches the wall and uses touch to orient
    - If there are no walls and just a post or column or more open space, she finds it difficult
        - There is less reverberation that she can use to orient without walls
        - Noisy streets make it difficult to orient

- "How do you understand where you are in space"
    - By the sound that I hear or by the sound that reverberates - sound that come from the left or right
    - If they can get close to a wall
    - Sounds from big streets can be disorienting or if there is a lot of noise covering up reverberations
        - In this case they resort to stopping people to ask for help

"How do you study a new environment"

- Studying their route - does this by touch, following a wall etc, using a cane, following furniture, windows or doors. The sounds of appliances and knowing where stairs are
- Primarily relying on sound
    - Sounds of cars and constructions sites are confusing
    - Cacophony of sounds make it difficult to pay attention to navigation
- People nearby is helpful because you can ask for help

"What kind of sensation helps you understand your position in environment"

- Wears high heels because the sound they produce helps
- Walk around room perimeter while holding onto wall to get an understanding of the location
- Empty rooms are easier because there is more sound reverberation
- Carpeting is a big hindrance because it prevents reverberations
- Ability to touch things helps
- Trees, posts and signal lights that don't emit sound make things difficult
- The sound of other people helps
- Some form of navigation but they have not experience with using maps - in this case they usually ask people
- Helps if the navigator issues very clear-cut commands and has a sound prompt

"What have they previously used for navigation"

- Cane
- Other people
- Open to trying new things

- Some are used to being without any navigation because after trying some that made mistakes, she distrusts them - feels more secure without them
- One lost her vision very gradually, so she still remembers having sight and usually tries having someone accompany her or ask a passerby for directions
- Some want to try something new, and others are distrustful of trying a new technical appliance

APPENDIX D: LIST OF ITEMS USED IN NEURAL NETWORK TRAINING

1. person
2. bicycle
3. car
4. motorcycle
5. airplane
6. bus
7. train
8. truck
9. boat
10. traffic light
11. fire hydrant
12. street sign
13. stop sign
14. parking meter
15. bench
16. bird
17. cat
18. dog
19. horse
20. sheep
21. cow
22. elephant
23. bear
24. zebra
25. giraffe
26. hat
27. backpack
28. umbrella
29. shoe
30. eye glasses
31. handbag
32. tie
33. suitcase
34. frisbee
35. skis
36. snowboard
37. sports ball
38. kite
39. baseball bat
40. baseball glove
41. skateboard
42. surfboard
43. tennis racket
44. bottle
45. plate
46. wine glass
47. cup
48. fork
49. knife
50. spoon
51. bowl
52. banana
53. apple
54. sandwich
55. orange
56. broccoli
57. carrot
58. hot dog
59. pizza
60. donut
61. cake
62. chair
63. couch
64. potted plant
65. bed
66. mirror
67. dining table
68. window
69. desk
70. toilet
71. door
72. tv
73. laptop
74. mouse
75. remote
76. keyboard
77. cell phone
78. microwave
79. oven
80. toaster
81. sink
82. refrigerator
83. blender
84. book
85. clock
86. vase
87. scissors
88. teddy bear
89. hair drier
90. toothbrush
91. hair brush

APPENDIX E: SURVEY CONSENT FORM

Informed Consent Agreement for Participation in a Research Study

Investigator:

Contact Information:

Title of Research Study: Navigation Methods for Blind or Visually Impaired Persons

Sponsor: Financial University under the Government of the Russian Federation, the Udmurtian Rehabilitation Center and the Children's "Zhuraveinik" Center.

Introduction:
You are being asked to participate in a research study. Before you agree, however, you must be fully informed about the purpose of the study, the procedures to be followed, and any benefits, risks or discomfort that you may experience as a result of your participation. This form presents information about the study so that you may make a fully informed decision regarding your participation.

Purpose of the study:
The purpose of the study is to determine how blind or visually impaired persons perceive space and the types of tools or methods used to navigate and explore environments safely.

Procedures to be followed:
The investigator will explain to the participant the purpose of the survey and cover the consent form with the participant. The participant will answer the survey. The investigator will receive the response and store it in accordance with IRB form (to WPI OneDrive). All interactions between participants and investigators will be done remotely.

Risks to study participants:
No associated risks with taking the survey. No personal information risks for the participants where the forms will be confidential.

Benefits to research participants and others:
Participants are able to explain problems with existing navigation methods and technology and aid in the development of new navigation solutions.

Alternative procedures or treatments available to potential research participants:

The alternative procedure to completing the survey is to decline to respond to the questions.

Record keeping and confidentiality:
Records of your participation in this study will be held confidential so far as permitted by law. However, the study investigators, the sponsor or it's designee and, under certain circumstances, the Worcester Polytechnic Institute Institutional Review Board (WPI IRB) will be able to inspect and have access to confidential data that identify you by name. Any publication or presentation of the data will not identify you.

Compensation or treatment in the event of injury:
Participants are not subject to any environments that would result in injury. The surveys will be administered remotely. You do not give up any of your legal rights by signing this statement.

For more information about this research or about the rights of research participants, or in case of research-related injury, contact:
<u>Investigative Team:</u>
Maranda Allen, Connor Miholovich, Samantha Boyea, Arden Carling
Tel. 508-831-5000
Email: gr-moscowd21-d3@wpi.edu

<u>IRB Manager:</u>
Ruth McKeogh
Tel. 508 831- 6699
Email: irb@wpi.edu

<u>Human Protection Administrator:</u>
Gabriel Johnson
Tel. 508-831-4989
Email: gjohnson@wpi.edu

Your participation in this research is voluntary. Your refusal to participate will not result in any penalty to you or any loss of benefits to which you may otherwise be entitled. You may decide to stop participating in the research at any time without penalty or loss of other benefits. The project investigators retain the right to cancel or postpone the experimental procedures at any time they see fit. (This section is required.)

By signing below, you acknowledge that you have been informed about and consent to be a participant in the study described above. Make sure that your questions are answered to your satisfaction before signing. You are entitled to retain a copy of this consent agreement.

____________________________ Date: ___________________
Study Participant SignatureStudy

____
Participant Name (Please print)

____________________________________ Date: ___________________
Signature of Person who explained this study

Additional clauses to add to Consent Agreements, as appropriate:
The treatment or procedures used in this research may involve risks to the subject (or to an embryo or fetus, if the subject is or may become pregnant), which are currently unknown or unforeseeable.
Additional costs to the subject that may result from participation in this research include: None expected due to remote delivery.

Significant new findings or information, developed during the course of the research, may alter the subject's willingness to participate in the study. Any such findings will be promptly communicated to all research participants.

Should a participant wish to withdraw from the study after it has begun, the following procedures should be followed: Notify the investigator immediately of willingness to withdraw. If the participant does not want to have their responses recorded, they must reach out to the investigator and inform them of how they would like their response to be withdrawn.

The consequences for early withdrawal for the subject and the research are:
All participants have the right to request their information be withdrawn or not recorded, as a result, there are no consequences for withdrawal.

Special Exceptions: Other requirements are found at 45 C.F.R. §46.116.

APPENDIX F: IRB EXEMPT APPROVAL LETTER

WORCESTER POLYTECHNIC INSTITUTE

100 INSTITUTE ROAD, WORCESTER MA 01609 USA

Institutional Review Board
FWA #00015024 - HHS #00007374

Notification of IRB Approval

Date:	01-Apr-2021
PI:	Mardilovich, Ivan P
Protocol Number:	IRB-21-0389
Protocol Title:	IQP Moscow D21 - Navigator for the Blind
Approved Study Personnel:	Miholovich, Connor~Allen, Maranda~Boyea, Samantha~Carling, Arden~Mardilovich, Ivan P~
Effective Date:	01-Apr-2021
Exemption Category:	2
Sponsor*:	

The WPI Institutional Review Board (IRB) has reviewed the materials submitted with regard to the above-mentioned protocol.We have determined that this research is exempt from further IRB review under 45 CFR § 46.104 (d). For a detailed description of the categories of exempt research, please refer to the IRB website.

The study is approved indefinitely unless terminated sooner (in writing) by yourself or the WPI IRB. Amendments or changes to the research that might alter this specific approval must be submitted to the WPI IRB for review and may require a full IRB application in order for the research to continue. You are also required to report any adverse events with regard to your study subjects or their data.

Changes to the research which might affect its exempt status must be submitted to the WPI IRB for review and approval before such changes are put into practice. A full IRB application may be required in order for the research to continue.

Please contact the IRB at irb@wpi.edu if you have any questions.

*if blank, the IRB has not reviewed any funding proposal for this protocol

APPENDIX G: PROJECT CODE

```
# Python program to translate
# speech to text and text to speech
import speech_recognition as sr
import pyttsx3
# Initialize the recognizer
r = sr.Recognizer()
# Function to convert text to
# speech
def SpeakText(command):
    # Initialize the engine
    engine = pyttsx3.init()
    engine.say(command)
    engine.runAndWait()
while (1):
    # Exception handling to handle
    # exceptions at the runtime
    try:
        # use the microphone as source for input.
        with sr.Microphone() as source2:
            # wait for a second to let the recognizer
            # adjust the energy threshold based on
            # the surrounding noise level
            r.adjust_for_ambient_noise(source2, duration=0.2)
            # listens for the user's input
            audio2 = r.listen(source2)
            # Using ggogle to recognize audio
            MyText = r.recognize_google(audio2)
            MyText = MyText.lower()
            print("Did you say " + MyText)
            SpeakText(MyText)
    except sr.RequestError as e:
        print("Could not request results; {0}".format(e))
    except sr.UnknownValueError:
        print("unknown error occured")
    if MyText == "Start Navigation":
        #call navigation code
    elif MyText == "Start object avoidance":
        #call directional code
    else:
        #standby
```

Above image: Developed code showing Speech-text-example with proposed commands to be implemented in our application. (“Python: Convert speech to text and text to speech,” 2019)

Navigator.py | directional.py | text_to_speech.py

```python
import cv2
import numpy as np
thres = 0.60
#img = cv2.imread("Resources/person.jpg")
cap = cv2.VideoCapture(0)
cap.set(3,640)
cap.set(4,480)
classNames = []
classFile = 'Resources/coco.names'
with open(classFile,'rt') as f:
    classNames = f.read().rstrip('\n').split('\n')
configPath = 'Resources/ssd_mobilenet_v3_large_coco_2020_01_14.pbtxt'
weightsPath = 'Resources/frozen_inference_graph.pb'
net = cv2.dnn_DetectionModel(weightsPath,configPath)
net.setInputSize(480,640)
net.setInputScale(1.0/127.5)
net.setInputMean((127.5,127.5,127.5))
net.setInputSwapRB(True)
while True:
    success,img=cap.read()
    classIds, confs, bbox = net.detect(img, confThreshold=thres)
    #print(classIds,bbox)
    gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
    # threshold
    thresh = cv2.threshold(gray, 128, 255, cv2.THRESH_BINARY)[1]
    # get contours
    result = img.copy()
    contours = cv2.findContours(thresh, cv2.RETR_EXTERNAL, cv2.CHAIN_APPROX_SIMPLE)
    contours = contours[0] if len(contours) == 2 else contours[1]
    for cntr in contours:
        x, y, w, h = cv2.boundingRect(cntr)
        cv2.rectangle(result, (x, y), (x + w, y + h), (0, 0, 255), 2)
        #print("x,y,w,h:", x, y, w, h)
    # save resulting image
    # show thresh and result
    cv2.imshow("bounding_box", result)
    if len(classIds) !=0:
        for classId, confidence,box in zip(classIds.flatten(),confs.flatten(),bbox):
            cv2.rectangle(img,box,color=(0,255,0),thickness=3)
            cv2.putText(img,classNames[classId-1].upper(),(box[0]+10,box[1]+30),cv2.FONT_HERSHEY_COMPLEX,1,(0,255,0),1)
            cv2.putText(img,str(round(confidence*100,2)),(box[0]+200,box[1]+30),cv2.FONT_HERSHEY_COMPLEX,1,(0,255,0),1)
        if box[0] < 200 or box[0] > 400:
            print('Stay Straight')
        elif box[0] >= 200 and box[0] <= 300:
            print('Move Right')
        elif box[0] >= 300 and box[0] <= 400:
            print('Move Left')
    cv2.imshow("Output", img)
    cv2.imshow('mask', thresh)
    key = cv2.waitKey(100)
    if key == 27:
        break
cap.release()
cap.destroyAllWindows()
```

Navigation method #1 which uses a mask which proved less useful for moving users, is ideal for a stationary camera. (*Computer Vision Projects., n.d.*) (Canu, 2021).

Navigator.py directional.py text_to_speech.py

```python
import cv2
import numpy as np
from tracker import *
from playsound import playsound
cap = cv2.VideoCapture(0)
tracker = EuclideanDistTracker()
#object detection
object_detector = cv2.createBackgroundSubtractorMOG2(history=100, varThreshold=30)
thres = 0.60
#img = cv2.imread("Resources/person.jpg")
#cap = cv2.VideoCapture(0)
#cap.set(3,640)
#cap.set(4,480)
classNames = []
classFile = 'Resources/coco.names'
#setting up navigational outputs for users and developers for when an object is detected
def move_left():
    if commands[0] != 'Left':
        # print('Left pixels:', left_pixels)
        commands.remove(commands[0])
        commands.append('Left')
        # if commands[0] != 'Right':
        print('Walk to your Left')
        playsound('Turn-left.wav')
def move_right():
    if commands[0] != 'Right':
        # print('Right pixels:', right_pixels)
        commands.remove(commands[0])
        commands.append('Right')
        # if commands[0] != 'Left':
        print('Walk to your Right')
        playsound('Turn-right.wav')
def stop_now():
    if commands[0] != 'Stop':
        commands.remove(commands[0])
        commands.append('Stop')
        # if commands[0] != 'Stop':
        print('Stop')
        playsound('Stop-walking.wav')
```

```python
def stay_straight():
    commands.remove(commands[0])
    commands.append('Straight')
    # if commands[0] != 'Straight':
    print('Walk Straight')
    playsound('Walk-straight-ahead.wav')
#def stop_soon()
#calling trained information
with open(classFile,'rt') as f:
    classNames = f.read().rstrip('\n').split('\n')
configPath = 'Resources/ssd_mobilenet_v3_large_coco_2020_01_14.pbtxt'
weightsPath = 'Resources/frozen_inference_graph.pb'
net = cv2.dnn_DetectionModel(weightsPath,configPath)
#setting capture and export window sizes
net.setInputSize(640,480)
net.setInputScale(1.0/127.5)
net.setInputMean((127.5,127.5,127.5))
net.setInputSwapRB(True)
#confidence and rectangle placement
while True:
    success, img = cap.read()
    classIds, confs, bbox = net.detect(img, confThreshold=thres)
    #print(classIds, bbox)
    if len(classIds) != 0:
        for classId, confidence, box in zip(classIds.flatten(), confs.flatten(), bbox):
            cv2.rectangle(img, box, color=(0, 255, 0), thickness=3)
            cv2.putText(img, classNames[classId - 1].upper(), (box[0] + 10, box[1] + 30), cv2.FONT_HERSHEY_COMPLEX, 1,
                        (0, 255, 0), 1)
            cv2.putText(img, str(round(confidence * 100, 2)), (box[0] + 200, box[1] + 30), cv2.FONT_HERSHEY_COMPLEX, 1,
                        (0, 255, 0), 1)
    ret, frame = cap.read()
    #obj detect
    #extract region of interest
    height, width, _ = frame.shape
    #print(height,width)
    roi = frame[0:480, 0:640]
    mask = object_detector.apply(roi)
    _, mask = cv2.threshold(mask,254,255,cv2.THRESH_BINARY)
    contours,_ = cv2.findContours(mask,cv2.RETR_TREE, cv2.CHAIN_APPROX_SIMPLE)
```

```python
    boxes_ids = tracker.update(detections)
    for box_id in boxes_ids:
        x, y, w, h, id = box_id
        cv2.putText(roi,str(id), (x,y-15),cv2.FONT_HERSHEY_COMPLEX,1,(255,0,0),2)
        cv2.rectangle(roi, (x, y), (x + w, y + h), (0, 255, 0), 3)
    # Crop left and right half of mask
    x, y, w, h = 0, 0, roi.shape[1] // 2, roi.shape[0]
    left = mask[y:y + h, x:x + w]
    right = mask[y:y + h, x + w:x + w + w]
    # Count pixels to determine where along the frame of reference the object falls
    left_pixels = cv2.countNonZero(left)
    right_pixels = cv2.countNonZero(right)
    commands = ['Left']
    if area_calc[len(area_calc) - 1] > 5000:  # and commands[0] != 'Stop':
        playsound('warning_beep.wav')
    elif area_calc[len(area_calc)-1] > 10000: #and commands[0] != 'Stop':
        stop_now()
    elif left_pixels > right_pixels and left_pixels > 500: #and commands[0] != 'Left':
        move_left()
    elif right_pixels > 500: #and commands[0] != 'Right':
        move_right()
    else:
        stay_straight()
    #print(commands)
    #cv2.imshow("roi",roi)
    cv2.imshow("Frame",frame)
    cv2.imshow("Mask",mask)
    cv2.imshow("Output",img)
    key = cv2.waitKey(100)
    if key == 27:
        break
cap.release()
cap.destroyAllWindows()
```

Final iteration of Navigation Code which uses audio commands (most effective) as the output to the user which is good for moving and static camera positions. (*Computer Vision Projects,* n.d.) (Canu, 2021).

```python

import os
import sys
# This allows us to import the local gdx module that is up one directory
gdx_module_path = os.path.abspath(os.path.join('.'))
# If the module is not found, uncomment and try two dots. Also, uncomment the print(sys.path)
#gdx_module_path = os.path.abspath(os.path.join('..'))
if gdx_module_path not in sys.path:
    sys.path.append(gdx_module_path)
# If there is an error trying to find the gdx module, uncomment this to see where
# the program is looking to find the gdx folder
#print(sys.path)
from gdx import gdx
gdx = gdx.gdx()
import math
gdx.open_usb()
#gdx.open_ble()
gdx.select_sensors([1,3,4])   # Hand Dynamometer sensors to use: 3 - y axis accel, 4 - z axis accel
gdx.start(period=200)
import turtle
square = turtle.Turtle()
square.shape("square")
square.shapesize(5,20)
#for i in range(0,100):
a = True
```

```python
while a:
    measurements = gdx.read()
    if measurements == None:
        break
    force = measurements[0]/5
    x_direction = measurements[1]/9.8
    y_direction = measurements[2]/9.8
    angle = math.atan2(y_direction, x_direction)  #compute the roll using two accelerometers (radians)
    print("angle radians =", angle)
    angle = (angle*180)/3.14  #convert radians to degrees
    print("angle degrees= ", angle)
    square.tiltangle(angle)
    square.shapesize(2+force,8+force)
    print("force/5 = ", force)
    if force>10:
        a = False
gdx.stop()
gdx.close()
turtle.done()
if x_direction < 25:
    print("Tilt phone up")
elif x_direction > 40:
    print("Titl phone down")
else:
    print("Phone is fine")
```

Gyroscope to angle output with proposed tilting camera function to allow our OpenCV/Navigation code to adjust to taller or shorter objects (*VernierST/Godirect-Examples*, n.d.).