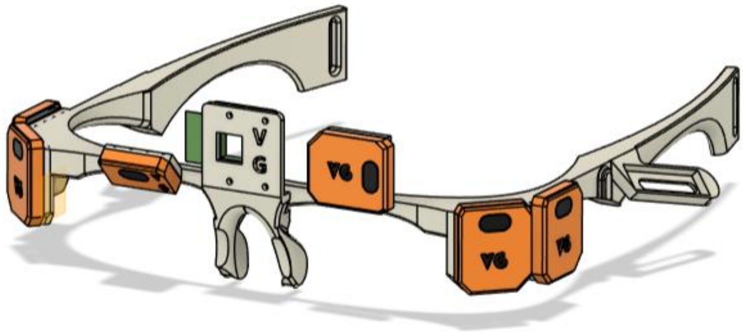
University of Hertfordshire UH

# PROJECT REPORT

## BEng




**Name:** Shenin Neleesha Nelundeniya
**Supervisor:** Ms. Sanuri Himalka

# VisionGuard

**Wearable Assistive Device for the Visually Impaired**

**01/03/2025**

www.herts.ac.uk

BACHELOR OF ENGINEERING DEGREE WITH HONOURS IN ELECTRICAL AND ELECTRONICS ENGINEERING

BEng Individual Project Report

Department of Engineering
School of Physics, Engineering and Computer Science
University of Hertfordshire

**VisionGuard: Wearable Assistive Device for the Visually Impaired**

Report by
SHENIN NELEESHA NELUNDENIYA

Supervisor
MS. SANURI HIMALKA

Date
01/03/2025

# DECLARATION STATEMENT

I certify that the work submitted is my own and that any material derived or quoted from the published or unpublished work of other persons has been duly acknowledged (ref. UPR AS/C/6.1, Appendix I, Section 2 – Section on cheating and plagiarism).

I confirm that any study conducted as part of my project that involved human participants has been approved in accordance with UH Ethical regulations and the protocol number(s) is/are:

LIST ANY ETHICAL APPROVAL REFERENCE NUMBERS HERE OR TYPE “N/A” IF NOT APPLICABLE

Student Full Name: SHENIN NELEESHA NELUNDENIYA

Student Registration Number:22044565

Signed: …………………………………………………

01/03/2025

Date: ……………………………………………………

# ABSTRACT

VisionGuard is a wearable assistive device designed for the visually impaired and blind individuals who face day-to-day challenges interacting with objects and navigating through an indoor environment. This project addresses the limitations of other available equipment such as white canes and guide dogs by incorporating object detection, path assistance and real-time feedback mechanism. The system was designed as a lightweight, head mounted system enabling hand-free navigation using time-of-flight sensors, haptic feedback and computer vision. The system compromises to main modules, path finding and object detection. The path finding module utilises TOF sensors and PWM vibrating motors mounted on to a 3D printed frame to detect nearby objects to user. The TOF sensors measure the distances in real time and pass data to the microcontroller which processes the measurements and activates the vibration motors through a PWM servo driver. The vibration feedback intensities will vary based on object proximity, proving tactile cues which will allow the user to avoid obstacles without the need of auditory distractions making it suitable for a noisy or crowded environment. Object detection utilises a microcontroller and an AI accelerator with a camera module to capture visual input and perform object and hand detection using a custom-trained model. The system identifies indoor objects based on the distance between the object and the user's hand, providing feedback through audio cues via speakers. The system was tested and implemented in controlled indoor environment using a blindfold subject to simulate visual impairment to obtain reliable results.

This report consists of seven main chapters with detailed explanations about the project. From the first chapter of the report, the user will get an idea on why this project will benefit the visually impaired and blind individuals worldwide. Chapter two will provide reviews of the projects with similar objectives as to VsionGuard. The methodology of this system will be discussed under Chapter Three followed by the tests and results in Chapter Four. Chapters Five, Six and Seven will give an detailed explanation on future development, project management and conclusion of this project. Overall, this report covers all the activities which took place throughout from the very start up til the end, providing a general idea to the party of interest.

# ACKNOWLEDGEMENTS

I would like to express my gratitude towards Nebula Institute of Technology and University of Hertfordshire for their assistance and for providing the opportunity to undertake the final year project.

First and foremost, I would like to thank my supervisor Ms, Sanuri Himalka for her continued support, encouragement, and guidance throughout this project. With her continuous feedback and timely advice this project was made a successful.

A special thanks to Mr. Darshana Bandara, our former distinguished program leader and UH coordinator for having supported us and brightening our path with understandings through this entire course of study.

I am also grateful to my parents for their continued support, knowledge, guidance and sacrifices they've made for me. Their beliefs in me and constant encouragement was the foundation of my journey, without any of their sacrifices and motivations none of this would have happened.

Lastly, I appreciate of all other lecturers, staff members, and all those who contributed their valuable time to facilitate and provide cumulative knowledge which was a source of illumination for a successful project.

# TABLE OF CONTENTS

# LIST OF FIGURES

# List of Tables

# GLOSSARY

TOF: Time Of Flight

YOLO: You Only Live Once

AR: Augmented reality

VR: Virtual reality

BCIs: Brain-computer interfaces

ETAs: Electronic Travel Aids

OCR: Optical Character Recognition

TTS: text-to-speech

# 1. Introduction

## *1.1 Project Background*

In the recent years, globally, over 2.2 million people are visually impaired, and at least 1 billion have visually impairment which can be prevented or treated with good support and technology. The visually impaired face significant challenges especially with mobility and object interaction in there day to day environments. The visually impaired usually deal with their surroundings with the help of equipment such as white canes and guide dogs. However, as technology improved with time, wearable devices with the combination of ultrasonic sensors, artificial intelligence and feedback mechanism have been designed to assist the visually impaired individuals. (WHO, 2019) (Khorrami-Nejad et al., 2016)

Although there's a wide range of tools which allow the visually impaired to navigate with their surroundings, these tools consist of some limitations. For instance, white canes, it can only detect objects below the person's waist within a short range of about 1.5 meters. On the other hand, guide dogs, while effective they must be trained extensively making them limited in availability. Due to these cons, it results in the visually impaired individuals facing challenges with mobility and orientation, affecting their ability to travel or do task independently. (Lin et al., 2012)

The proposed solution to the above problem is to design a wearable device which can guide the visually impaired in a way the individual can navigate with their surroundings and do tasks independently. This is where VisionGuard comes in handy. VisionGuard was designed to enhance the mobility and independence of visually impaired individuals by addressing key challenges in navigating with their surrounding and identifying objects near them.

VisionGuard consists of two main sections:

1. **Path Assistance Module:**
   This module utilizes time-of-flight sensors and vibrating motors to provide real-time haptic feedback to guide users in navigating through and indoor environment. The system acts as the user's senses, allowing them to perceive direction and distance through vibrations, by reducing their reliance on white canes or guide dogs.
2. **Object Detection Module**:
   In this section, a camera and an AI accelerator, helps the user identify objects closest to their hand. Haptic signals and auditory cues via speakers are used to deliver feedback to the user when an object closest to either one of the user's hands is identified.

With the combination of the two modules, VisionGuard will aid and object recognition for visually impaired to navigate and do their task independently which gives the user more privacy. VisionGuard is a promising development in assistive technology because of its creative approach, and it has the potential to be a game-changing tool for helping people with visual impairments.

## 1.2 Project Aims and Objectives

### 1.2.1 Aims

The aim of this project is to build a wearable assistive device name VisionGuard for the visually impaired, particularly the blind, to navigate through an indoor environment without having to rely on white canes or having anyone to assist them. The device makes use of object detection allowing the users to interact with objects which are commonly used in their day-today lives, for instance, cusp, chairs, doors, etc.

The final product will be assessed in a controlled indoor setting, where a test subject will wear the device to move independently and successfully recognise and interact with objects using the system's assistive features.

### 1.2.2 Objectives

The required objectives to achieve the above aims are as follows:

1. Make use of time-of-flight sensors and vibrating motors to enable real-time path assistance for the visually impaired individuals, which will help them in navigating through an indoor environment without of the need of white canes or anyone's support.
2. An AI accelerator along with a camera is used to detect objects and determine their proximity to the user's hands in an intuitive way for users to recognise objects without physical contact.
3. Design a system to provide the user with directional feedback through vibrations with varying intensities based on the distance between the user and the objects, acting as the user's senses for a better understanding.
4. Produce a mechanism which can recognise the user's hands and provide information about the object closest to the user's hand.
5. Include a voice-based feedback mechanism to inform the users of the location of the objects closest to their hand while improving special awareness and safety.

## 1.3 Methodology Overview

VisionGuard uses time-of-flight sensors for spatial awareness and a camera module connected to a Raspberry Pi equipped with a Hailo 8 AI accelerator for soft real-time object detection and hand tracking. Sensor data will be process by the system to provide vibratory feedback for navigation, while the objects will be detected closest to the user's hand using the camera and AI accelerator. A detailed explanation of the system's architecture is given in the Methodology chapter (Chapter 3), and a visual representation of the process flow is provide in Figure 3.2.

## 1.4 Project Feasibility and Marketability

The initial estimated costs of the project are as detailed in Table 1.1, with focus on the core components, the time-of-flight sensors, camera module, vibration motors and other necessary hardware components. However, the system had to incorporate a Raspberry pi with Hailo 8 AI accelerator to enhance object detection accuracy, processing speed and increased system responsiveness which provides certain advantages in terms of the performance. Although, this benefits the project, there was a change in the estimated budget.

Despite the changes, the feasibility of the projects remains strong due to the availability of the necessary components, and with improvements in embedded AI processing which makes it a feasible object detection in a wearable device and an efficient device for assistance which justifies the adjusted budget. Hence, strong and a rising need for assistive devices for the visually impaired and blind individuals, which appears to be a strong potential market for VisionGuard.

The marketability of this products is substantial and continuously growing for the visually impaired individuals. Hence, due to its focus on practical, real-time assistance for day-today navigation and interaction it’s a highly marketable product.

**Table 1-1: Estimated Project Budget**

| | Component | Specification | Units | Status | Price (LKR) |
|---|---|---|---|---|---|
| 1. | AI Module | Seeed Studio Grove Vision AI Module V2 | 1 | Allocated to the UH budget | 5500/- |
| 2. | Microcontroller | Seeed Studio XIAO ESP32-C3 | 1 | Allocated to the UH budget | 2200/- |
| 3. | Time-of-Flight Sensors | TOF200C Lidar Sensor | 5 | Allocated to the UH budget | 3800/- |
| 4. | Camera Module | Raspberry Pi Camera V2 | 1 | Allocated to the UH budget | 6500/- |
| 5. | Vibrating Motors | 3.7V Mini Vibration Motors | 5 | Allocated to the UH budget | 1000/- |
| 6. | Power Bank Module | Generic 7.4 Power Bank Module | 1 | Allocated to the UH budget | 550/- |
| 7. | Battery | 1000 mAH LiPo 701855 | 2 | Allocated to the UH budget | 3000/- |
| 8. | Audio Board | LM386 | 1 | Allocated to the UH budget | 350/- |
| 9. | Speakers | Notebook Speakers 8 Ohms | 2 | Allocated to the UH budget | 1000/- |
| 10. | 3D Printing Materials | 3D Printed Frame and Bands | - | Allocated to the UH budget | - |

| | | | | | |
|---|---|---|---|---|---|
| 11. | Miscellaneous Components | - | - | Owning in posession | - |
| Total | | | | | 24,000/- |

# 2. Literature Review

## *2.1 Introduction*

A detailed review based on the technologies and research related to the assistive devices for the visually impaired will be presented in this chapter. Existing solutions which provide obstacle detection, object recognition, navigation assistance, and real-time feedback will be examined in this chapter. This section of the report will help position VisionGuard with the current landscape, highlighting how it differentiates from other existing technologies. (WHO, 2019)

## *2.2 History and Evolution*

For a long period of time, visually impaired individuals have depended on different tools and strategies to navigate with their surroundings.

During the earliest stages, individuals with low vision used simple magnifying devices such as polished stones and lenses to improve the visual clearness, and simple tools like canes were used to assist with mobility for those who had difficulty walking. The first biggest development in assistive technology was the Braille which was created by Louise Braille in 1824. This creation allowed the blind individuals to read and write independently, profoundly changing education and communication for the visually impaired. (Pandit, 2024)

With time, as technology developed, new options emerged. In the 13th century spectacles with lenses were created and the magnifying glass was invented in the 16th century. (Pandit, 2024)

Hence, in the 20th century, there was a rise in low vision technology innovation from scientific discoveries and a greater understanding of the requirements of the visually impaired individuals. Among the significant turning points are:

- In the 1920s, telescopic lenses were developed for distance viewing.
- In the 1930s closed-circuit televisions (CCTVs) were introduced to zoom in printed materials.
- In the 1940s, Perkins Brailler typewriter was invented which changed the way the blind can read.
- In the 1950s, electronic reading ais were developed providing a path for future digital available solutions.
  (Pandit, 2024)
- In the 1970s, ultrasonic sensors were utilised ultrasonic waves to detect objects in automotive parking systems and provide an audible warning signal to the drivers. This was later modified for the use of object detection and distance measurement used in robotics and automation. (IJRCIT, 2023)
- In the 1980s, ultrasonic sensors were initially used in a device which was developed for the visually impaired to detect objects in the individual’s surroundings, providing audio feedback to

the user. This device was not practical for everyday use due to its limited size and complexity. (IJRCIT, 2023)

Looking ahead, a new era of low vision technology is accompanied by the digital age. Where smartphones and tablets consisting of various features like voice controls, text-to-speech, and screen magnification, offer a surplus of tools for daily living and communication. Augmented reality (AR) and Virtual reality (VR) have an enormous potential with navigation, education and rehabilitation for the visually impaired. Hence, rising technologies like Brain-computer interfaces (BCIs) enables restoring sight by bypassing damaged parts of the eye or provides a direct communication between the brain and external devices. With AI and embedded vision significantly improving the capabilities of wearable aids, there are devices available now image processing in real time, object recognition, and object classification. (Pandit, 2024)

## *2.3 Existing Assistive Technologies*

In the recent years, enhanced ETAs (Electronic Travel Aids) have given opportunities to help visually impaired users with navigation. These systems detect environmental obstacles and convert them to an auditory or tactile form to update conventional gear by leveraging electronic sensors. (Al Maraashli et al., 2025)

ETAs are normally designed using ultrasonic sensor due them being low costs, simple and capabilities in finding objects through contemplated sound wave measurements. However, these systems can be impacted, resulting in low accuracy and reliability in complex environments. Moreover, ultrasonic sensors feedback may have delays and may not have sufficient directional data which can avert the person’s potential to reply faster to potential boundaries. (Al Maraashli et al., 2025)

Camera based systems include use of computer vision or LiDAR which provides improved spatial resolution and can understand precise object types. Whilst current ETAs can improve in object detection and orientation, there’s still the desire in obtaining high accuracy, real-time feedback, low power consumption, and comfortably wearable shape factor. (Al Maraashli et al., 2025)

## 2.4 Current Research and Devices

### 2.4.1 Smart Vision Glass: Elevate Your Sight

Smart Vision is an AI-powered wearable device developed by SHG Technologies, a Bangalore-based tech startup, in collaboration with Vision Aid. It consists of an AI algorithm which enables the visually impaired to read printed/handwritten text, object and facial recognition, and navigation assistance. The system utilises a camera with the AI algorithm to provide the visually impaired with real-time audio feedback. (Vision Aid, 2024)

Smart Vision Glass only depends on audio prompts, whereas Vision Guard utilises ultrasonic sensors to detect obstacles depending on the distance to provide feedback in the form of vibrations for path assistance. This is specifically efficient in a noisy environment. Although Smart Vision Glass tries to focus on many challenges, VisionGuard focuses deeply on safe indoor movement, providing haptic feedback and hand-guided object detection of the visually impaired which is often the most essential environment. (Vision Aid, 2024)

Furthermore, VisionGuard focuses mainly safer indoor navigation and object detection with hand interaction making it useful in home or workplace environment where the need for safe mobility is highly needed. When we compart Smart Vision Glass to VisionGuard it has a risk in diluting performance across features due its attempts to serve many use cases. This makes VisionGuard a more refined and context-aware solution than the Smart Vision Glass. (Vision Aid, 2024)

### 2.4.2 DLWV2: a Deep Learning-based Wearable Vision-System

Shih et al. (2018) designed a Deep Learning-Based Wearable Vision-System with Vibrotactile-feedback (DLWV2). DLW2 uses a camera and HTC Vive Tracker to perform perception and guidance. Hence, vibrating motors have been used for guidance. The system produces vibrotactile cues by translating object detection data. Both the DLWV2 and VisionGuard share similar tactile feedback but ther differ in their approach to user interaction. (Shih et al., 2018)

DLWV2 utilises external sensors and depends on motion tracking offering spatial precision but compromises portability and wearability in uncontrolled settings. However, VisionGuard recognises the left/right hand of the visually impaired to provide audio feedback of the closest object to hand, using YOLOv11 and onboard AI processing. Although DLWV2 provides real-time object guidance using vibrations, it is limited for everyday use due to its reliance in bulky trackers. (Shih et al., 2018)

### 2.4.3 A Wearable Device for Indoor Imminent Danger

Li et al. (2020) proposed a system with a depth camera and performs object detection along with object perception which finally provides acoustic feedback in an indoor environment. This project performs object detections, and a similar technology is used as VisionGuard by the means of vibrating motors, but VisionGuard uses time-of-flight sensors to obtain accurate depth estimation and an AI module for object detection. (Li et al., 2020)

This system heavily relies on a depth camera which can be sensitive to lighting conditions. In contrast to this VisionGuard uses time-of-flight sensors for real-time depth estimation and vibrating motors to provide haptic feedback. Li et al.'s system relies mostly on sound-based alerts while VisionGuard improves usability in noisy indoor environments by delivering tactile signals instead of audio. (Li et al., 2020)

Overall, VisionGuard is better suited for wearability, power efficiency, and hands on object detection while Li et al.'s system does not personalise interaction beyond basic object detection.

### 2.4.4 RGB-D Wearable Vision System

The System proposed by Moghimi et al. (2014) focuses on object classification and depth perception using wearable vision. Objects are visually recognised and their distance from the user are estimated with the help of RGB imaging combined with depth sensing. The system was designed for independent activity recognition, to get an understanding of the user's actions and environmental layout. However, this system is impractical for real-world, independent daily use because it heavily relies on an external computing power. (Moghimi et al., 2014)

This system is constrained with heavy reliance on external computational resources making it less practical for real-world and mobile use. Moghimi's system lacks user interactivity while VisionGuard provides more clear, accessible, and empowering solution in environments where traditional aids are limited for the visually impaired and blind individuals. (Moghimi et al., 2014)

### 2.4.5 Multi-Functional Glasses

Bastole et al. (2023) developed a multifunctional wearable device to assist the blind and visually impaired individuals with navigation and environmental recognition. The glasses combine several components which include a camera, microphone, speaker and an online server. (Bastola et al., 2023)

The function of this devices is to send real-time visual data to a cloud server to process the data using the AI models and returns the required information back to the user. The system performs object detection, object recognition and basic path guidance. (Bastola et al., 2023)

In contrast to it, VisionGuard is an offline, real-time, and priorities user needs with focus to an indoor environment for safer navigation and intuitive object detection. Although the multifunctional glasses provide multiple use cases via online processing, VisionGuard simplifies and optimises the user experiences for important and daily indoor task without any internet access. (Bastola et al., 2023)

### 2.4.6 An AI-Based Visual Aid with Integrated Reading Assistant for the Completely Blind

The AI-based visual aid was designed by Khan el al. to help the completely blind individuals printed content and improve mobility. OCR (Optical Character Recognition) with TTS (text-to-speech) are combined in this system to help the users' read books, signs, or documents by listening to a spoke for the text. This system consists of a camera, Raspberry Pi, Tesseract OCR, eSpeak TTS engine, and a speaker. (Khan et al., 2020)

However, VisionGuard offers soft real-time object detection, tactile feedback, and better path guidance compared to the AI-based visual aid since its use is limited in real-world navigation and interaction scenarios. This clearly shows that VisionGuard is more robust, scalable, and practical for everyday use of the visually impaired and blind. (Khan et al., 2020)

## *2.5 Sensor Technologies used in Assistive Devices*

### 2.5.1 Ultrasonic Sensors

Ultrasonic sensors produce high frequency soundwaves and evaluate the echo which is received back by the sensor. The time interval between the sending the signal and receiving the echo is calculated to determine the distance to an object. It is like an infrared which reflect on the surface in any shape. Hence, when we compare the ultrasonic sensor with other sensors it provides more accurate results. For this project the TOF200C VL53L0X time-of-flight sensors are used because they have a greater range, higher accuracy, fasters readings and the ability to operate in various conditions. (Aye et al., 2016)

### 2.5.2 Depth Cameras

Assistive devices make use of depth cameras to obtain crucial spatial information and navigation assistance. This makes it beneficial for the visually impaired individuals facing mobility challenges. Depth cameras allow devise to detect objects, obstacles, and distances by capturing a 3D representation of the surrounding. This project utilises the Raspberry Pi V2 camera instead of a depth camera due to its cost and ease of use. (Kusuma et al., 2023)

### 2.5.3 Time-of-Flight Sensors

These sensors detect the time it takes for a signal to reflect from an object to measure the distance. TOF sensors are utilised in assistive devices to give real-time feedback on proximity and direction of objects, increasing safety and navigation as mentioned before the TOF200C VL53L0X is utilised in VisionGuard to assist a visually impaired user in navigating with their environment. (Al Maraashli et al., 2025)

## 2.6 AI and Object Detection in Wearable Assistive Devices

AI and Object Detection both work together to help visually impaired individuals navigate through their surroundings more safely and independently. Object detection makes used of sensors to determine the presences and distance of objects followed by a camera which captures images with the help of AI models like YOLO (You Only Look Once) to train to identify and locate objects in images. This enables complex object detection by identifying different objects including people. (Swami et al., 2025)

On the other hand, AI analyses collected data by sensors to provide profound information to the user. It can be used to provide source of action by guiding the user to avoid obstacles or to find specific objects while providing an output as feedback which can be understood by the user (i.e. vibrating patterns, audio cues and so on). (Swami et al., 2025)

# 3. Methodology

## 3.1 Introduction

This chapter briefs through the methodology followed through put the development of VisionGuard, a wearable assistive designed for the visually impaired or blind individuals. The devices were designed targeting to main objectives: indoor navigation and object detection with hand-based interaction. Methodology focuses on two key modules which utilises various tools and platforms to achieve the desired results.

## 3.2 System Architecture

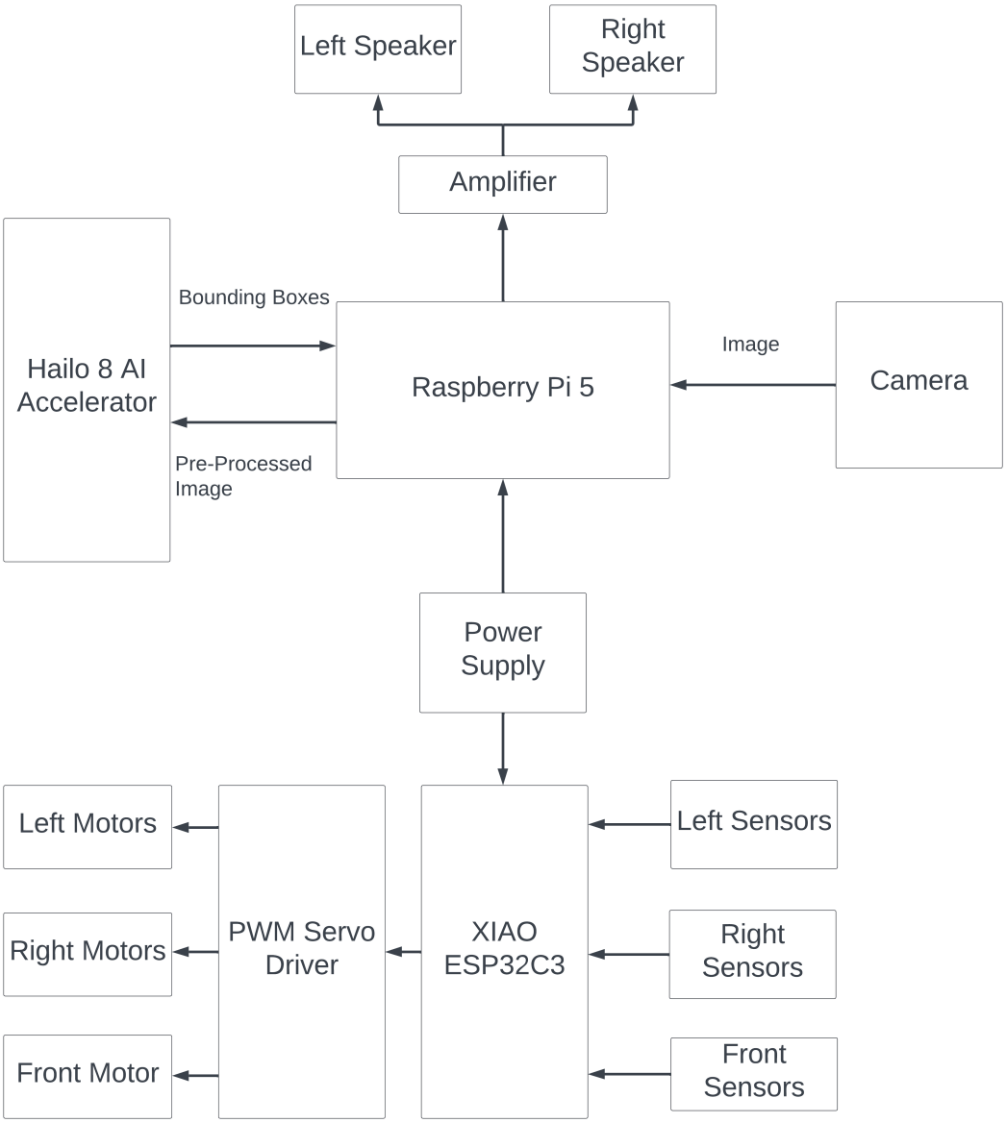



**Figure 3-1: System Architecture**

## 3.3 Design Approach

VisionGuard was developed to assist the visually impaired and blind by helping them to navigate and do tasks by interacting with objects independently in an indoor setting.

VisionGuard utilizes a modular, sensor-fusion, and user-centred design approach to develop a wearable assistive device for the visually impaired and blind individuals. The system enhances navigation and object identification by providing a soft real-time tactile and audio feedback.

The main modules of the system are Path Assistance and Object Detection modules. Where the path assistance module will be responsible for obstacle avoidance and navigation guidance, along with it the object detection module will be responsible for identifying and providing spatial information about the objects closest to the user's hand. The result will be an independent development, testing, and maintenance, and will provide a future expansion of features.

Figure 3.2 illustrates the operations of VisionGuard. The system starts by initialising sensor values by getting sensor data from the time-of-flight sensors and a camera. The received data is then processed to calculate the distance to obstacles and identify objects and hands in the user's surrounding. Feedback is then provided to the users based on the information through vibrations of the vibrating motors and an audio output. This assists the users through their environment with information to objects closest to them.

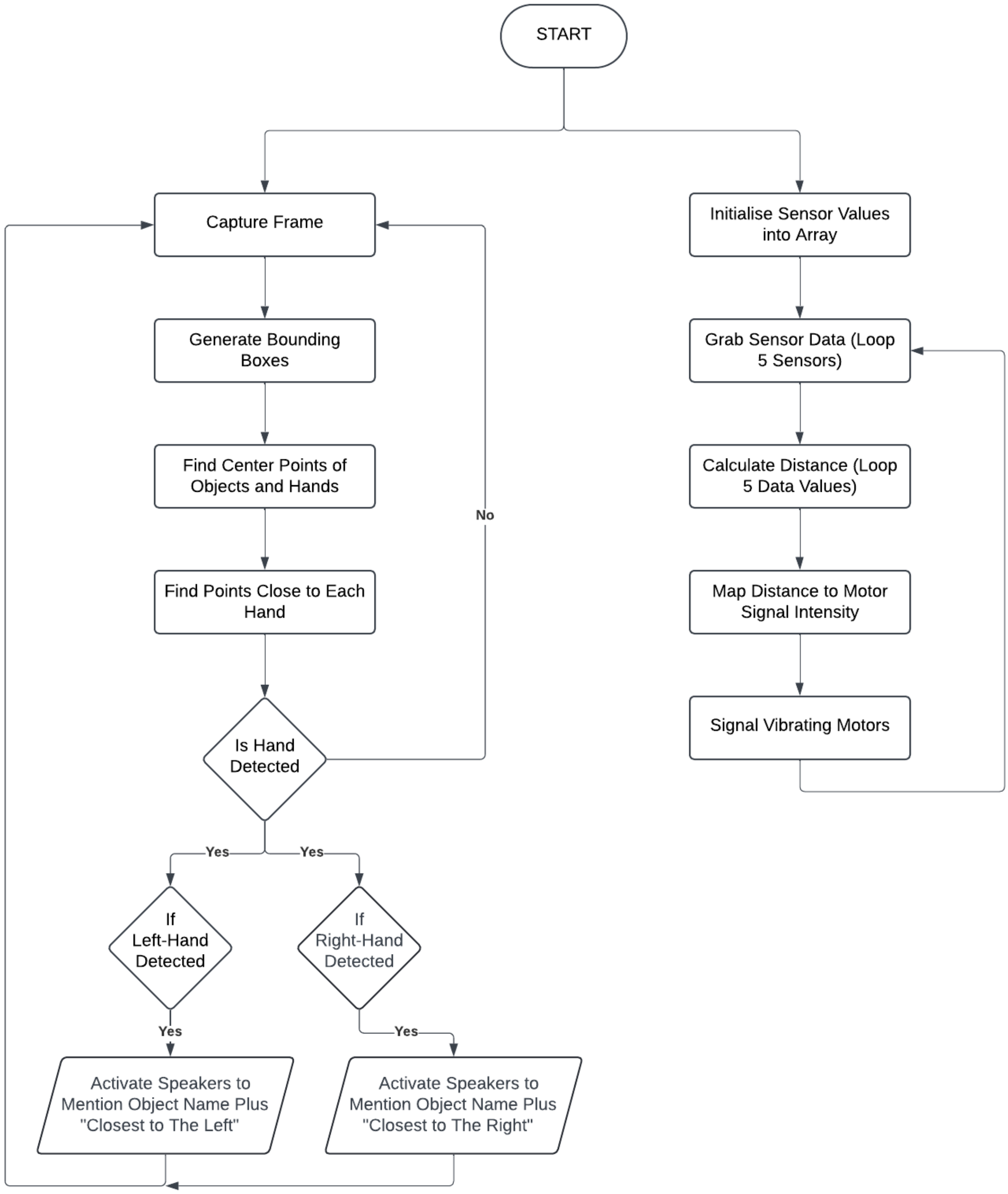



**Figure 3-2: System Flow Chart**

## 3.4 The Required Hardware Components

### 3.4.1 Seeed Studio XIAO ESP32-C3 Mircontroller

The XIAO ESP32C3 which acts as a primary microcontroller for Path Assistance module is an IoT mini development board built on the Espressif ESP32-C3 Wfi/Blutooth dua-mode chip, featuring a 32-bit RISC-V CPU for strong computing performance with its efficient architecture. It is responsible for real-time control and processing of data from the time-of-flight sensors and the handling of the vibration motors. (SeeedStudio, 2024)

The board's 11 digital I/O pins, configurable the PWM outputs are used to control the intensities of the vibrating motors. Hence the I2C helps communication with the time-of-flight sensors for getting the distance measurements, and an external antenna reliable communication with the sensors. (SeeedStudio, 2024)


**Figure 3-3: XIAO ESP32-C3 Microcontroller**

**Product Datasheet**
(XIAO ESP32-C3 Microcontroller)

**Product Justifications**

1. The Seeed Studio XIAO ESP32C3 consists of a combination of processing capabilities, efficient power consumption, and a quality interface.
2. The 32-bit RISC-V CPU, operating at 160MHz provides sufficient processing power for real-time data acquisition and control of ToF sensors and motor vibrations.
3. The 11 GPIOs with the PWM capabilities gives a precise control of vibration motor intensity.

### 3.4.2 TOF200C VL53L0X Time-of-Flight Ranging Sensor

The VL53L0X is a time-of-flight laser ranging module which provides accurate distance measurement by emitting infrared light pulses and measuring the time taken for the light to return. It can measure an absolute distance of up to 2m of various directions and different heights. (VL53L0X Time-of-Flight ranging sensor)

VisionGuard makes use of six time-of-light sensors which interact with the ESP32C3 microcontroller via the I2C interface to measure the distance between the user and the surrounding objects within a range of up to 2m. The sensors are placed as follows to provide spatial awareness:

- Two sensors on the left side of the frame and two sensors on the right side of the frame.

- One sensor facing directly forward and one sensor facing downwards from the front of the frame.

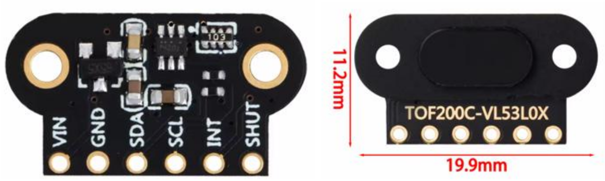



**Figure 3-4: TOF200C VL53L0X**

**Product Datasheet**

(VL53L0X Time-of-Flight ranging sensor, Datasheet)

**Product Justifications**

1. It is a fully integrated miniature module with 940nm laser VCSEL (Vertical-Cavity Surface-Emitting Laser) and a ranging sensor with advanced embedded microcontroller.
2. It is fast and provides an accurate distance ranging.
3. These sensors are Class 1 laser devices, compliant with the latest eye safety IEC 60835-1:2014 – 3rd edition, maintaining a safe operation for the user's eyes.
4. VL53L0X sensors are designed for easy integration, which is facilitated by its single reflowable component design, no additional optics, single power supply operation, I2C interface for control data, a GPIO for reset or an interrupt and a programmable I2C address.

### 3.4.3 PWM Vibration Motor Sensor Module Switch

It is a small built-in vibration motor module which can be controlled ON or OFF or the vibrations of the motor through a digital signal or PWM signal, by inputting 5V power. (Grobotronics, n.d) (Design, n.d)

VisionGuard employs five these motors to provide haptic feedback to the visually impaired or blind individuals by indicating the direction and the proximity of the obstacles. The intensity of these motors will vary with the distance between the object and the user, where stronger vibrations will be outputted indicating the closer proximity. Hence, the motors are placed as follows to provide clear directional cues to the user:

- Two motors on the left side and two motors on right side of the frame.
- One motor at the front of the frame facing forward.

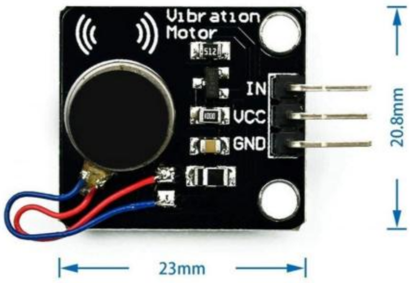



**Figure 3-5: PWM Vibration Motor**

**Product Datasheet**

(Vibration Motor, Data Sheet)

**Product Justifications**

1. This module makes used of high-quality mobile phone vibration motor, which provides a clear and tactile feedback, which is crucial for alerting the visually impaired individuals.
2. The motor's vibration can be controlled by a PWM, which means the vibration feedback can be modulated. (e.g. for different proximity levels the signal intensity can by varied)
3. This module vibrates to recall a small vibration which is suitable as a non-audible indicator allowing for discreet communication of information without any auditory clutters.

(Grobotronics, n.d)

### 3.4.4 PCA9685 12-bit 16-channel PWM Servo Driver

A PWM Servo Motor was used to efficiently manage the orientation and positioning of the TOF sensors. This was used because the ESP32C3 microcontroller had a limited number of PWM channels available, and the PCA9685 was chosen to overcome this limitation. It is an I2C controlled module with 16 independent PWM outputs, each with 12-bit resolution. This makes it ideal for fine control od signal timing applications.

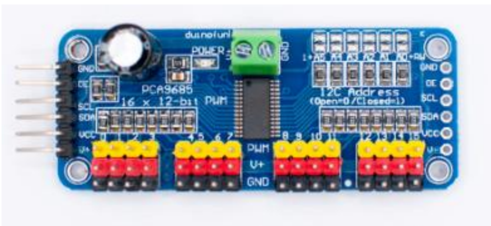



**Figure 3-6: PWM Servo Driver**

**Product Datasheet**

(Adafruit PWM Servo Motor Datasheet, 2014)

**Product Justification**

1. To directly control 5 vibration motors more output pins are required and the ESP32C3 has limited GPIO. The PCA9685 expands this capability via I2C.
2. It allows fine pulse control due to its 12-bit resolution.
3. The module is well supported in Arduino IDE and is compatible with the ESP32C3.

### 3.4.5 Raspberry Pi 5

The raspberry Pi 5 consists of a 64-bit quad-core Arm Cortex-A76 processor running at 2.4GHz.
The Raspberry Pi 5 along with the Hailo 8 AI Accelerator allows to run inference on the Hailo-8L processor, enabling soft real-time object detection with models like YOLOv11. The improved computational power of the Raspberry Pi 5, especially with the upgraded GPU increases the ability to handle high processing powers tasks. Hence, with the improved GPU, it can handle AI tasks such as object detection and image recognition. (Raspberry Pi Foundation, 2023)


**Figure 3-7: Raspberry Pi 5**

**Product Datasheet**

(Raspberry Pi5 Datasheet, 2023)

### 3.4.6 Hail- 8 AI Accelerator

The Hailo-8 is used in AI applications compatible with NGFF M.2 form fact M, B+M and A+E keys. It is integrated with the Raspberry Pi to significantly improve the speed and efficiency of the object detection process. It is specifically designed for high performance with a low power consumption. The Hailo 8 runs the trained object detection model, providing soft real-time analysis of captured camera frames. (Hailo, 2023b)


**Figure 3-8: Hailo-8 AI Accelerator**

**Product Datasheet**

(Hailo-8 AI Accelerator Datasheet, 2023b)

**Justification for The AI Component Change**

The initial plan of the project was to make use of the Seeed Studio Grove Vision AI Module V2 for AI processing. However, due to minimal performance, a decision was made to utilise the Raspberry Pi 5 in conjunction with the Hailo AI accelerator for the need of superior performance in real-time object detection. Compared to the Seeed Studio Grove Vision AI Module V2, the Hailo 8 offers higher computational throughput and dedicated AI acceleration capabilities. This increased processing power is important for analysing complex scenes and getting the accuracy and required speed to provide a timely and reliable feedback to the user in a dynamic environment.

**Validation for AI Processing Component Change**

As given in the datasheet provided in Appendix A, the technical specifications of Hailo 8 consist of a higher TOPS (Tera Operations Per Second) rating compared to Seeed Studio Grove Vision AI Module V2, allocating a large increase in processing capabilities for AI tasks. Further, the integrations of Hailo 8 with the Raspberry Pi 5 provides a more robust and flexible platform for development and deployment of complex AI models which require accurate and fast object detection. (Hailo, 2023a)

### 3.4.7 Raspberry Pi Camera V2

The Raspberry Pi Camera Module v2 is an 8-megapixel IMX219 image sensor custom deigned add-on board Raspberry Pi, featuring a fixed focus lens. The reason for choosing this module was due to its capability in delivering a crystal clear 5MP resolution image. It has Picture Resolution: 2592 x 1944, Video: Supports 1080p30, 720p60 and 640x480p90 60/90 Recording, 15-pin MIPI Camera Serial Interface - Plugs Directly into the Raspberry Pi Board. Its size is 20 x 25 x 9mm and weight is 3g. It is used to capture visual information from the user's surrounding environment. The camera module is directly interfaced with the Raspberry Pi board through a dedicated CSI (Camera Serial Interface) connector for transmission of high-resolution image data to the processing unit.
(Farnell, 2015)

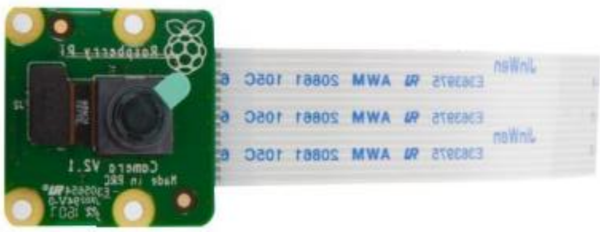


**Figure 3-9: Raspberry Pi Camera Module v2**

**Product Datasheet**

(Raspberry Pi Camera Module v2 Datasheet, 2015)

**Product Justifications**

1. It is an 8-megapixel Sony IMX219 sensor which allows capture of high-quality images allowing an accurate object detection by the AI module.
2. Its direct integration with the Raspberry Pi through the dedicated CSI interface provides an efficient and faster data transfer which is important for object detection and processing.

## 3.5 Software Tools

### 3.5.1 Arduino IDE

The Arduino IDE is a free open-source programming environment for developing and uploading codes to microcontrollers like the ESP32. Arduino IDE plays a crucial role in processing the data from the time-of-flight sensors and controlling the vibrations of the motors for the feedback system. The Arduino code written in a streamlined version of C and C++, given in Appendix B, includes the necessary libraries at the start (Adafruit_PWMServoDriver and Adafruit_VL53LoX) to have an interface with the TOF sensors and the GPIO expander.

The setAddresses() function initialises the I2C addresses of the six motors. Consequently, the getDistance() function reads the distance measurements from the sensors. The measured distances are then mapped to the vibration motor intensity using the motorMap() function on the ESP32C3.

Finally, the ActivateMotors() function controls the vibration motors based on the mapped intensity values. The setup() function configures the microcontroller and sensor pins, while the loop() function gathers sensor data, maps it to the motor intensity, and activates the vibration motors to provide real-time tactile feedback.

### 3.5.2 Roboflow

Roboflow is a software platform which can be used to train custom models, manage datasets, and deploy object detection pipelines. In this project Roboflow was used to annotate and label the objects in the images of the uploaded dataset. (Roboflow, n.d)

### 3.5.3 Ultralytics Yolov11

Utralytics YOLOv11 framework was used to train the dataset, which was manually annotated using Roboflow, and converted into YOLO format. The YOLOv11 model was set with a pre-trained weight file and using the Ultralytics training API it was fine-tuned on the custom dataset. (Ultralytics, n.d)

The reason for choosing Ultralytics over other frameworks was due to its support for the latest YOLO versions and user-friendly API.

### 3.5.4 RealVNC Viewer

The RealVNC was used when dealing with the Raspberry Pi remotely, enabling remote access and control of the Raspberry Pi. Interactions with the Raspberry Pi's GUI (graphical User Interface) over a local network was made convenient because of this software. This was useful when it came to troubleshooting and monitoring the systems.

### 3.5.5 Python IDLE

Object detection of the system was developed using Python Programming Language. Python supports AI and deep learning libraries like Utralytics YOLOv11, PyTorch, TensorFlow, and OpenCV, all which were used in the certain stages of the project.

Python Scripts were developed to load the trained YOLOv11 model to capture frames using the Raspberry V2 camera, run object detection in soft real-time and provide the user with audio feedback based on the objects detected.

## *3.6 Design Platforms*

### 3.6.1 Autodesk Fusion 360

The physical structure of VisionGuard was designed using Autodesk Fusion 360 as a primary 3D modelling platform. It is a cloud-based tool which allows precise modelling. The frame was designed in a way that it had cutouts and mounting slots for various hardware components used in the project ensuring a compact integration and ease of assembly while making it suitable for wearable use.

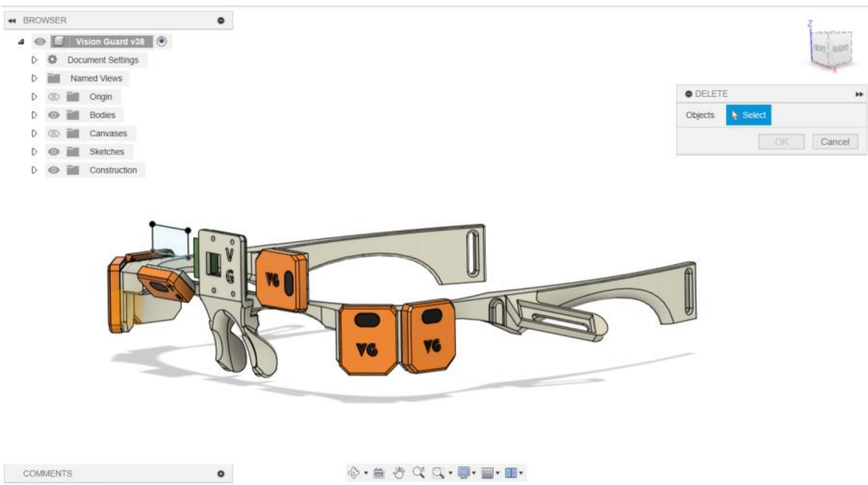



**Figure 3-10: 3D Design of the Frame**

## *3.7 Design Implementation*

### 3.7.1 Module 01: Path Assistance

**Working Principle**

The system assists navigation by detecting nearby objects to the visually impaired individual and gives the user physical feedback. This task was achieved by utilising the TOF200C VL53L0X sensors, Xiao ESP32C3 microcontroller, and vibrating motors controlled via PWM signals.

Six TOF sensors were tactically mounted onto the 3D printed frame in a way the sensors can detect the objects on either side of the user, the objects which are in front of the user and at ground level. These sensors will continuously function by measuring the distance between the user and the objects in their surrounding as they navigate through their environment. By emitting infrared signals, they can calculate the time taken for the to reflect from the surface providing accurate distance measurement in real time.

The Xiao ESP32C3 is the central processing unit of this system which gathers readings from the TOF sensors to process the data and determine whether there is an object within the predefined threshold. Based on this, the side and direction of objects to be detected are decided.

As per the sensor inputs, the vibrating motors placed on either side of the frame and at the front will be activated by the ESP32C3 based on the direction of the detected object/s. The vibration intensity of the motor will vary with distance of the object. Higher the intensity of the vibrations, closer the distance between the user and the obstacle and lower the intensity of the vibrations, further the distance between the user and the obstacle. Figures 3.11 to 3.14 demonstrates how the motors will be activated based on the sensor data.

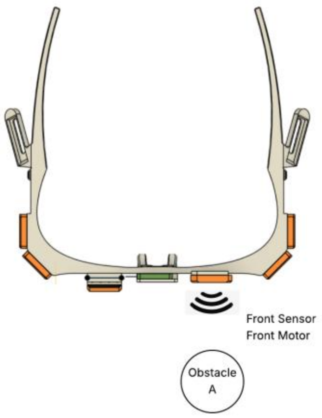



**Figure 3-11: Obstacle Infront**

Figure 3.11 illustrates how the front sensor will provide the user with feedback through front motor vibrations with varying intensities based on the distance between the user and the obstacle when an obstacle is detected in front of the user.

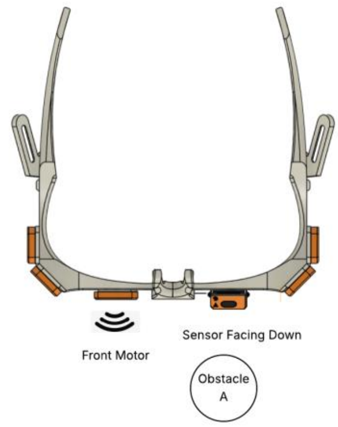



**Figure 3-12: Obstacle on the Ground**

Figure 3.12 illustrates how the sensor facing down will provide the user with feedback through all the motors vibrating with varying intensities with a vibration pattern based on the distance between the user and the obstacle when an obstacle is detected on the ground next to the user.

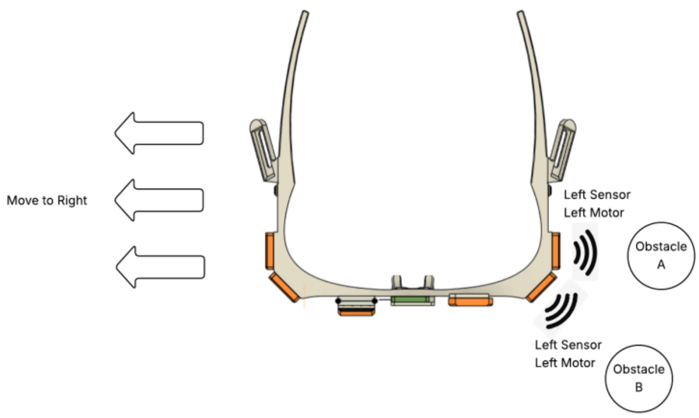



**Figure 3-13: Obstacles on the Left**

Figure 3.13 illustrates how the Left sensors will provide the user with feedback through left-side motors vibrations with varying intensities based on the distance between the user and the obstacle when an obstacle is detected on the ground next to the user.

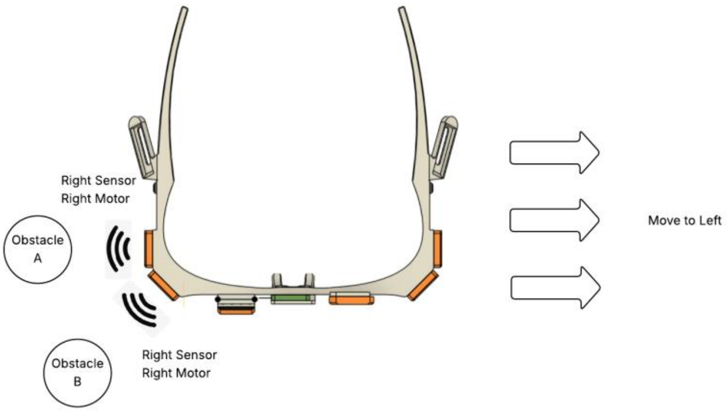



**Figure 3-14: Obstacles on the Right**

Figure 3.14 illustrates how the right sensors will provide the user with feedback through right-side motors vibrations with varying intensities based on the distance between the user and the obstacle when an obstacle is detected on the ground next to the user.

### 3.7.2 Module 02: Object Detection

**Working Principle**

Object Detection was carried out using the combination of the Raspberry Pi 5, the Hailo-8 AI accelerator, and the Raspberry Pi V2 camera module. The Raspberry Pi Camera grabs a frame, sends the frame to the Pi5. The Pi5 preprocesses this image to the needed input shape by the yolo models which is (1, 3, 640, 640). This pre-processed image is sent to the Hailo-8 AI accelerator module to perform an inference on the image for the two models, one the pretrained YOLOv11 model trained on the coco8 dataset and the custom trained model on the hand detection. The accelerator module sends the results, i.e. the bounding boxes, categories, confidences and other relevant data.

The Pi 5 then uses this data to identify the positions of each object and hand then proceeds by the aid of a virtual middle line in the frame, to allow the algorithm to know the relative position of a detected hand as either left or right. The objects centre points are identified and the one closest to the detected hand’s centre point is recognised.

An output message is generated as such, “remote next to left hand move slightly right to reach”. This text is sent to the text to the google text to speech algorithm ( gTTS ) and the message is sent to the audio amplifier and from amplifier to loudspeaker voicing out the generated output.

# 4. Results and Analysis

## *4.1 Introduction*

In this chapter the outcomes obtained for development and testing of VisionGuard can be observed. The evaluation of both path assistance and object detection have been provided. The results were gathered from controlled environment. Finally for an effective delivery in reliable feedback and the support of independent navigation, each component was assessed.

## *4.2 Path Finding*

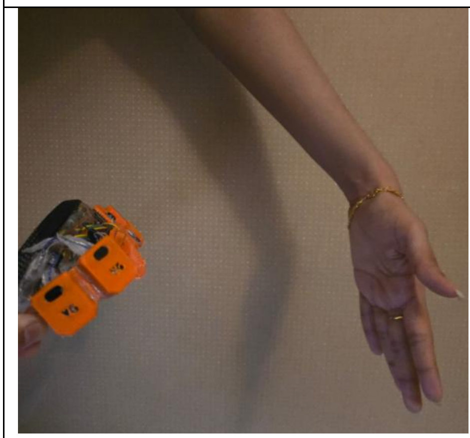
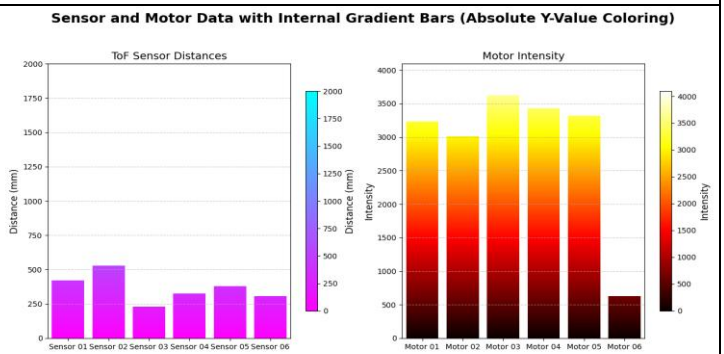
The Table 4.1 will represent the sensor data and the generated PWM output of the path finding system based on the environment in the images shown. The has been plotted using matplot lib by entering data manually into the graph.

Note: Sensor 06 corresponds to motor 06 though there is no 6th motor. This is just to represent the proportionate PWM value generated in order to provide the intensity of the warning pulse.

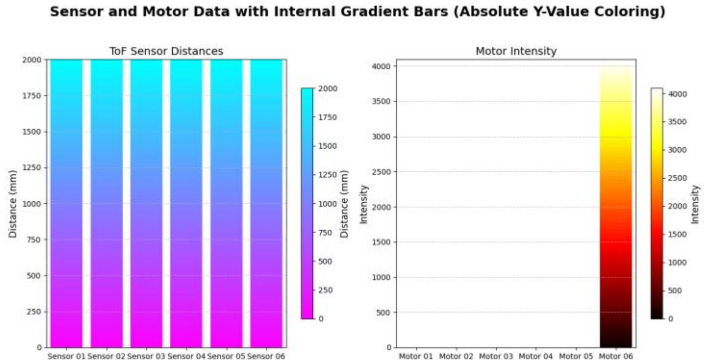
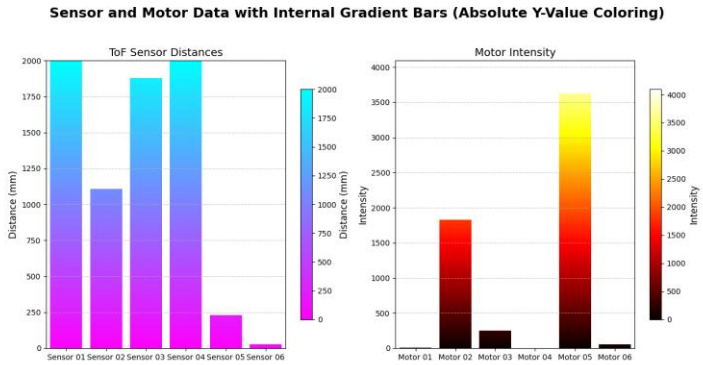
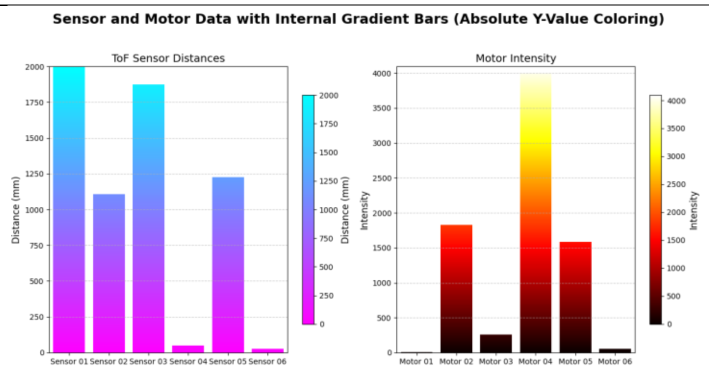
**Table 4-1: Sensor and Motor Representation**

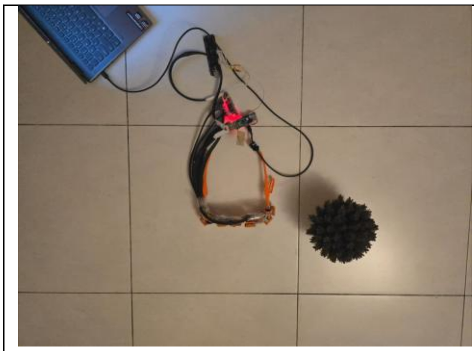
| Sensor Number | Sensor Direction | Motor Number | Motor Direction |
|---|---|---|---|
| Sensor 01 | Right side | Motor 01 | Right side |
| Sensor 02 | Right side 45 degrees | Motor 02 | Right side 45 degrees |
| Sensor 03 | Left side | Motor 03 | Left side |
| Sensor 04 | Left side 45 degrees | Motor 04 | Left side 45 degrees |
| Sensor 05 | Front | Motor 05 | Front |
| Sensor 06 | Down | Motor 06 | Down |

**Table 4-2: Results and Analysis of Path Finding**

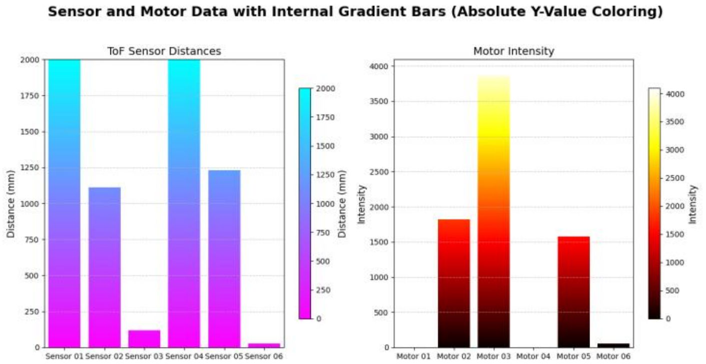
<table>
<tr><th>Environment Image</th><th>Results Obtained</th></tr>
<tr><td>
Hand placement in front of the sensor facing downwards.</td><td>

When the hand was placed in front of the sensor the above results were obtained resulting in a sudden increase in the intensities of the five motors based on the measured distance between palm and the downward sensor.</td></tr>
</table>

| | |
|---|---|
| <br>Downward sensor with no object placement in front of the sensor. | <br>The graph shows a sudden increase in distance thus resulting in a directly proportionate increase in the magnitude of the PWM signal from motor 05 |
| <br>An object placed in front of the front sensor. | <br>The graph shows when a distance of 230mm was measured from sensor five, there was a rise in the motor 05 intensity to about 3600 to warn the user of an object nearby. |
| <br>An object placed in front of the left sensor at 45 degrees. | <br>The distance measured from sensor 04 (50mm) was very low resulting in a higher motor intensity output from motor 04. |

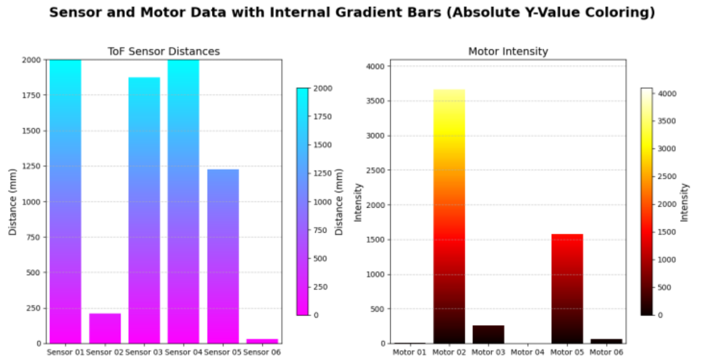


An object placed in front of the left sensor.



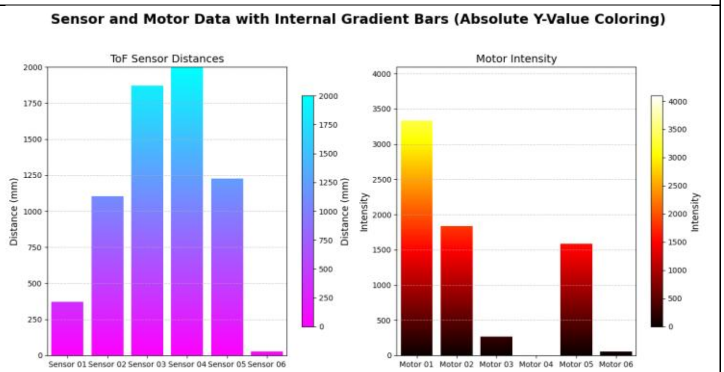

The distance measured from sensor 03 was 120mm and it resulted in a motor intensity of about 3800.

An object placed in front of the right sensor at 45 degrees.




The distance measured from sensor 02 was close to 210mm resulting in a higher motor intensity output from motor 02.

An object placed in front of the right sensor.




The distance measured from sensor 01 (37mm) was very low resulting in a higher motor intensity output from motor 01.

## 4.3 Object Detection


**Figure 4-1: Command entered to initiate model training**

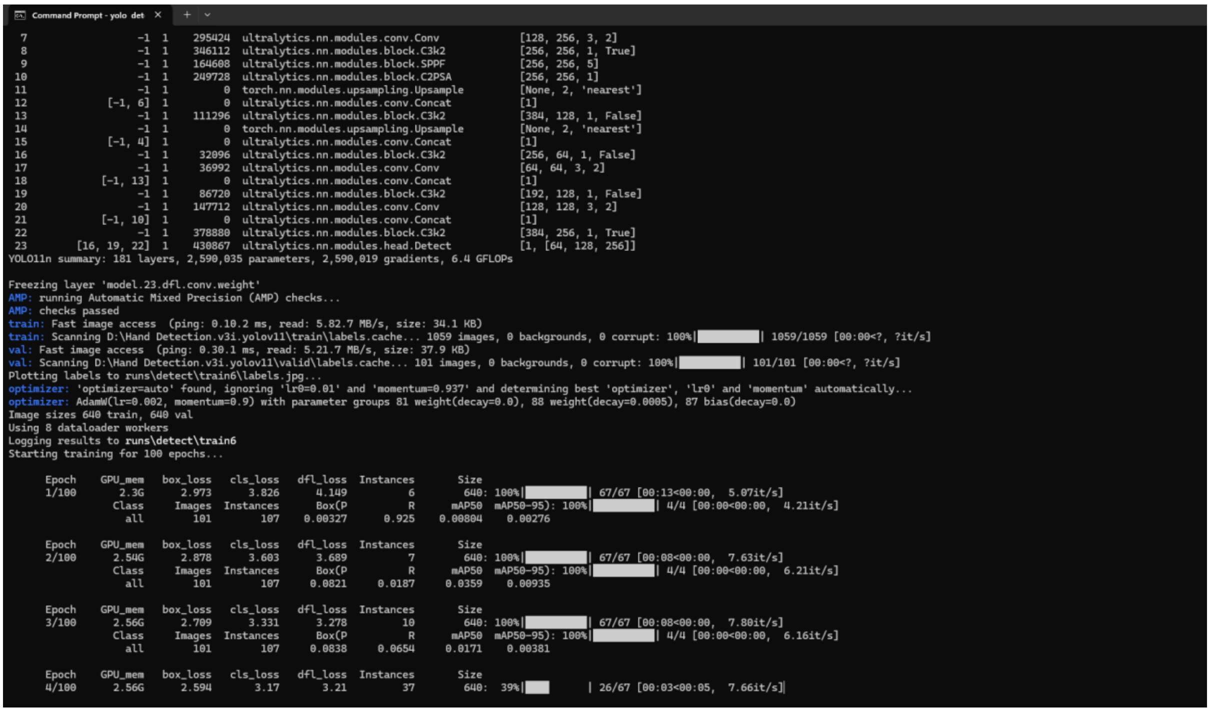



**Figure 4-2: Training in progress, continues for 100 epochs or until model reaches a constant loss and accuracy**

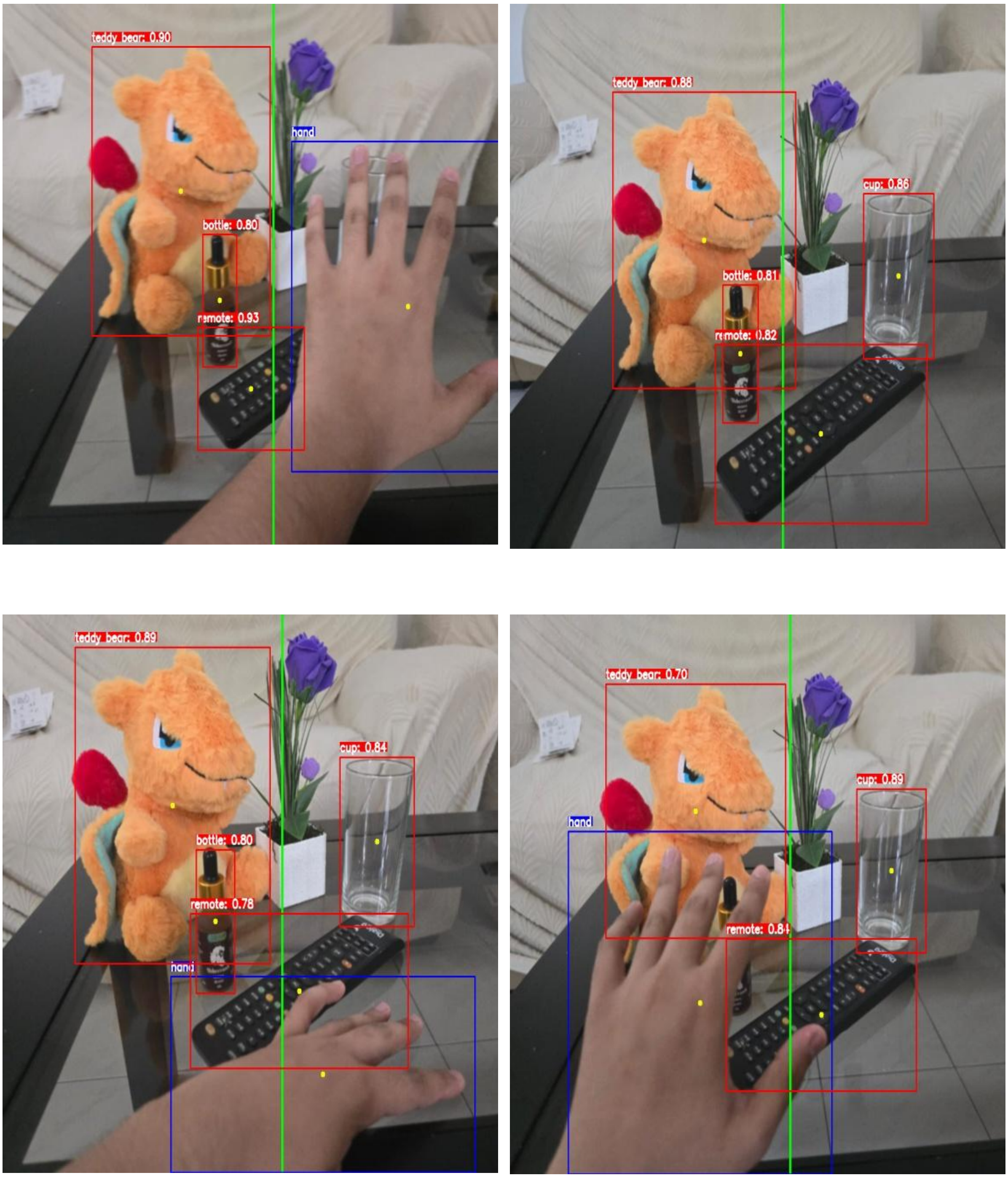



**Figure 4-3: Results and Analysis of Object detection**

The above images show inferences of images and the resulting bounding boxes along with centre points of each image coloured in yellow, and the green centre line is just to represent the division of the frame. Hence, where the hand is detected, it will classify it as either left or right respectively. Using the centre points the algorithm will find the object with the shortest distance between centre point of the hand and the centre point of the object of interest. Then as mentioned in the methodology it will generate the needed output prompt to be played by the loudspeaker.

# 5. Conclusion

VisionGuard was designed to detect objects and provide a clear path for the user and help the user interact with daily used objects in their indoor environment. With over 2.2 billion people experiencing some form of visual impairment, the need of a wearable assistive devices is needed more than ever since traditional equipment such as white canes and guide dogs have limited capabilities. VisionGuard was designed to overcome these limitations by utilising TOF sensors, vibrations motors, microcontrollers, an AI accelerator, speakers and a camera module to provide the user with a adaptive feedback mechanism in a wearable form.

The key modules of the system were: path assistance and object detection modules. Where the path assistance module will utilise TOF sensors to measure distances of an object to provide feedback through motor vibrations. The object detection module utilised a Raspberry Pi along with an AI accelerator to identify objects in the user's environment and detect the hand closest to the object. This approach allowed the final system to provide tactile and auditory feedback to the user enhancing user awareness and enabling independent navigation with privacy.

The project was able to achieve the core modules from technical to user-oriented perspectives. The users can be reliably guided around nearby objects with the help of the sensor arrangement and mapping logic, while object detection effectively recognises everyday items and contextualises them with respect to the user's hand. For the simulation of a natural feedback, and output audio cue like "Cup to the right" will be given to the user bridging the gap between perception and response for the user.

The decision to replace Seeed Grove Vision AI module with the Raspberry Pi 5 and Hailo-8 accelerator was validated through performance testing, which resulted in better inference speeds and detection accuracy. Similarly, the hand detection part was changed from initial colour hand band method to recognising the hand with an AI-based method, increasing robustness and making the device less intrusive.

In conclusion, VisionGuard was designed to help assist the visually impaired or blind individuals to navigate through their indoor environments without any external equipment or someone having to assist them, providing them with privacy and independence. The final project was able to achieve its key goals and provide the user with a versatile and reliable product. With further developments, VisionGuard can be expanded into a commercially viable product for all visually impaired worldwide.

# 6. Further Development

The initial design of VisionGuard assists the user with navigation in an indoor setting utilising path assistance and object detection. However, this system can be further modernised both functionally and technically to meet broader user requirements and move toward commercial viability.

VisionGuard can be developed into a system which can be used both in an indoor setting and an outdoor setting which is a major area to be focused on. Through a GPS integration, routes for navigation can be detected and the system can be trained to detect traffic lights, vehicles, crosswalks, and other uban obstacles present in an outdoor environment. This enhances VisionGuard, making it a full travel companion for the visually impaired and blind individuals with the improvement of AI capabilities to detect and recognise road signs and environmental landmarks.

The user interaction with the system can be improved using advanced feedback mechanism which includes spatial 3D audio cues which creates perception of sound coming from different directions. Features like muting audio or toggling modes through simple hand gestures can be included into the smart frame to support gesture recognition allowing the user to control the features based on their liking. The performance of VisionGuard can be further improved and battery life by Quantisation and pruning, optimising the use of YOLOv11.

This product can be made further smaller and more comfortable for the user. This can be achieved by replacing the Raspberry Pi with a compact edge AI hardware. Improved power management features can be included to increase the operational features; this can be achieved by having lower -power sleep modes and wireless charging support.

In the future an mobile app can be developed for VisionGuard. This app can be developed in a way where the user can have better control over the device. The user should be able to manage modes, initiate training sessions for new objects, or review environmental logs through this app. Moreover, the system should be able to learn about the frequently used items by the user in their indoor environments to support object personalisation and provide faster object detection. Thus, for safety perspective the system could also have a built-in emergency alert feature. For instance, a fall detection with SMS notification to a caregiver can be included.

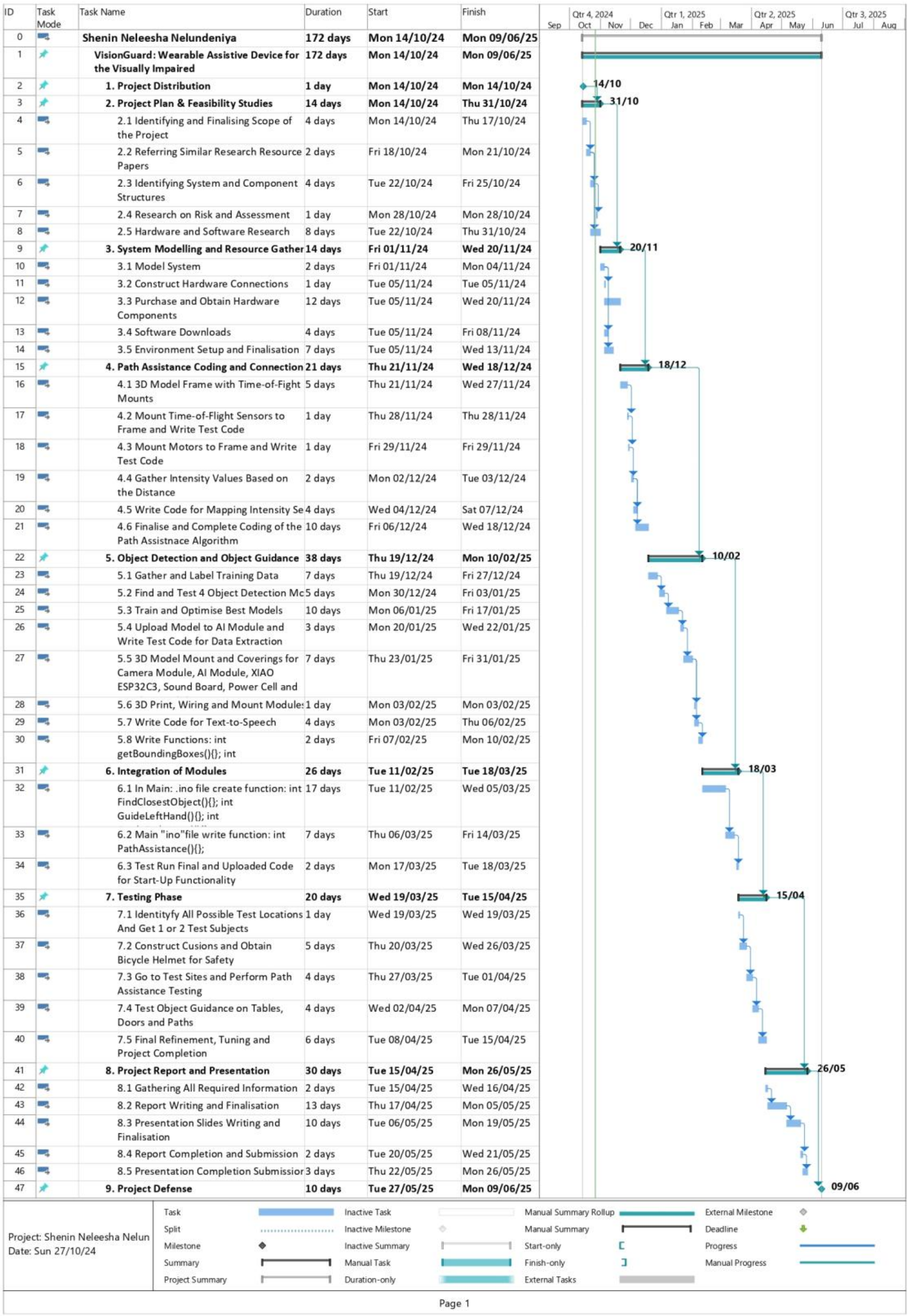
Overall, VisionGuard can be fully developed into a system able to do broader functionalities. The future system of VisionGuard can be made to read text and signs aloud, help visually impaired individuals educationally and in community-based applications. As VisionGuard continues to be developed, it has the potential to be affordable, life changing, read to market for all visually impaired and blind individuals worldwide.

# 7. Project Management Review

## *7.1 Time Evaluation*

For a successful implementation of VisionGuard an effective project management of the project was crucial. The initial Gantt chart as shown in Figure 7.1 outlines how the projects was planned to be carried out throughout the six months. Each stage had allocated buffers to hold minor setbacks, on the completion of hardware integration early, followed by model training and fine tuning.

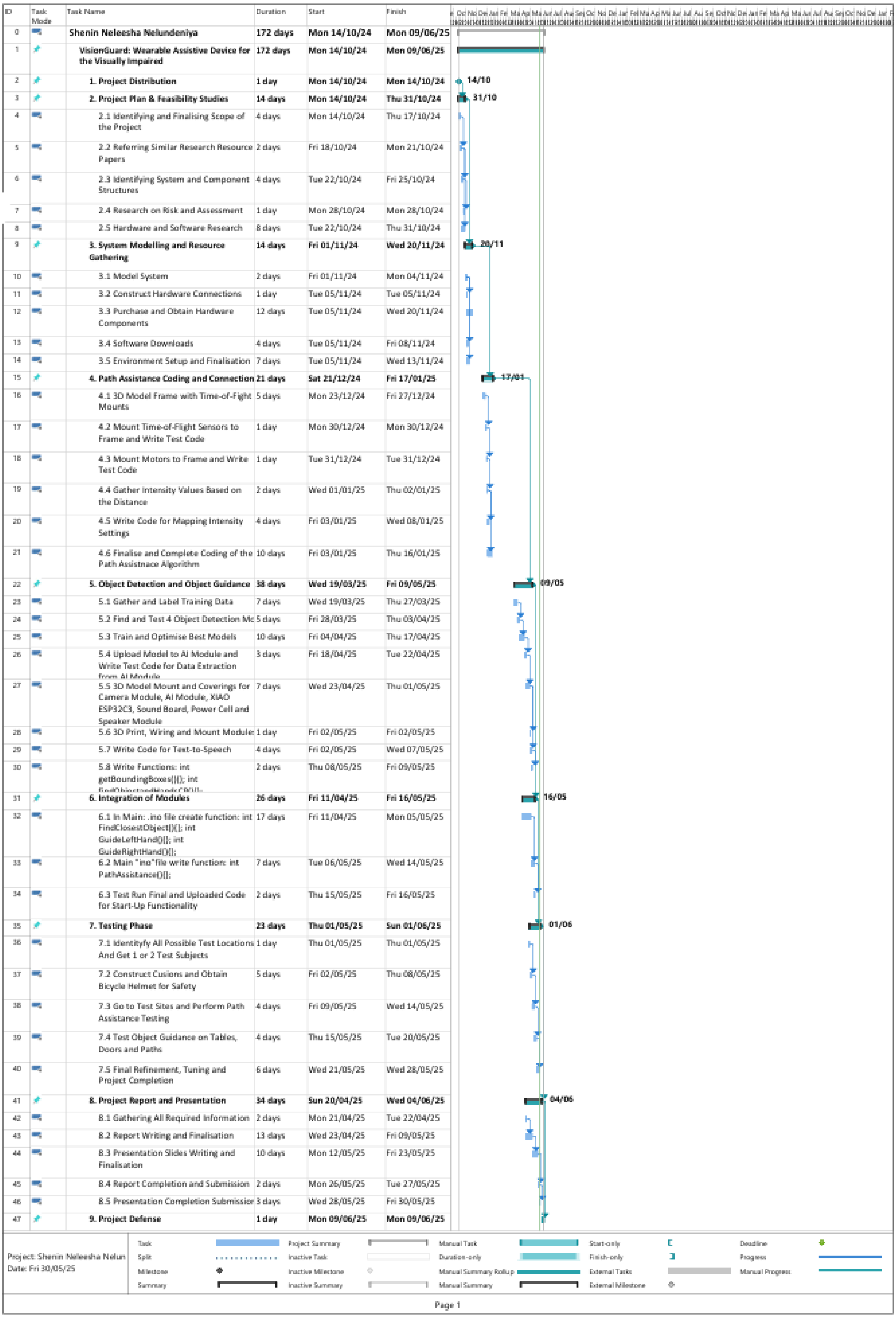
During execution the project followed the original structure, but certain changes and delays occurred. Although path finding module was completed on schedule, a change was made in components of the object detection module. This required extra time to reconfigure software for compatibility with the Raspberry Pi and Hailo-8 accelerator. Further, time was needed to train and deploy YOLOv11 instead of lightweight alternatives. Although certain changes had to delay the project, it was possible to complete every phase within the given time-period before the deadline.



| ID | Task Mode | Task Name | Duration | Start | Finish |
|---|---|---|---|---|---|
| 0 | | **Shenin Neleesha Nelundeniya** | **172 days** | **Mon 14/10/24** | **Mon 09/06/25** |
| 1 | | **VisionGuard: Wearable Assistive Device for the Visually Impaired** | **172 days** | **Mon 14/10/24** | **Mon 09/06/25** |
| 2 | | **1. Project Distribution** | **1 day** | **Mon 14/10/24** | **Mon 14/10/24** |
| 3 | | **2. Project Plan & Feasibility Studies** | **14 days** | **Mon 14/10/24** | **Thu 31/10/24** |
| 4 | | 2.1 Identifying and Finalising Scope of the Project | 4 days | Mon 14/10/24 | Thu 17/10/24 |
| 5 | | 2.2 Referring Similar Research Resource Papers | 2 days | Fri 18/10/24 | Mon 21/10/24 |
| 6 | | 2.3 Identifying System and Component Structures | 4 days | Tue 22/10/24 | Fri 25/10/24 |
| 7 | | 2.4 Research on Risk and Assessment | 1 day | Mon 28/10/24 | Mon 28/10/24 |
| 8 | | 2.5 Hardware and Software Research | 8 days | Tue 22/10/24 | Thu 31/10/24 |
| 9 | | **3. System Modelling and Resource Gather** | **14 days** | **Fri 01/11/24** | **Wed 20/11/24** |
| 10 | | 3.1 Model System | 2 days | Fri 01/11/24 | Mon 04/11/24 |
| 11 | | 3.2 Construct Hardware Connections | 1 day | Tue 05/11/24 | Tue 05/11/24 |
| 12 | | 3.3 Purchase and Obtain Hardware Components | 12 days | Tue 05/11/24 | Wed 20/11/24 |
| 13 | | 3.4 Software Downloads | 4 days | Tue 05/11/24 | Fri 08/11/24 |
| 14 | | 3.5 Environment Setup and Finalisation | 7 days | Tue 05/11/24 | Wed 13/11/24 |
| 15 | | **4. Path Assistance Coding and Connection** | **21 days** | **Thu 21/11/24** | **Wed 18/12/24** |
| 16 | | 4.1 3D Model Frame with Time-of-Fight Mounts | 5 days | Thu 21/11/24 | Wed 27/11/24 |
| 17 | | 4.2 Mount Time-of-Flight Sensors to Frame and Write Test Code | 1 day | Thu 28/11/24 | Thu 28/11/24 |
| 18 | | 4.3 Mount Motors to Frame and Write Test Code | 1 day | Fri 29/11/24 | Fri 29/11/24 |
| 19 | | 4.4 Gather Intensity Values Based on the Distance | 2 days | Mon 02/12/24 | Tue 03/12/24 |
| 20 | | 4.5 Write Code for Mapping Intensity Se | 4 days | Wed 04/12/24 | Sat 07/12/24 |
| 21 | | 4.6 Finalise and Complete Coding of the Path Assistnace Algorithm | 10 days | Fri 06/12/24 | Wed 18/12/24 |
| 22 | | **5. Object Detection and Object Guidance** | **38 days** | **Thu 19/12/24** | **Mon 10/02/25** |
| 23 | | 5.1 Gather and Label Training Data | 7 days | Thu 19/12/24 | Fri 27/12/24 |
| 24 | | 5.2 Find and Test 4 Object Detection Mc | 5 days | Mon 30/12/24 | Fri 03/01/25 |
| 25 | | 5.3 Train and Optimise Best Models | 10 days | Mon 06/01/25 | Fri 17/01/25 |
| 26 | | 5.4 Upload Model to AI Module and Write Test Code for Data Extraction | 3 days | Mon 20/01/25 | Wed 22/01/25 |
| 27 | | 5.5 3D Model Mount and Coverings for Camera Module, AI Module, XIAO ESP32C3, Sound Board, Power Cell and | 7 days | Thu 23/01/25 | Fri 31/01/25 |
| 28 | | 5.6 3D Print, Wiring and Mount Module: | 1 day | Mon 03/02/25 | Mon 03/02/25 |
| 29 | | 5.7 Write Code for Text-to-Speech | 4 days | Mon 03/02/25 | Thu 06/02/25 |
| 30 | | 5.8 Write Functions: int getBoundingBoxes(){}; int | 2 days | Fri 07/02/25 | Mon 10/02/25 |
| 31 | | **6. Integration of Modules** | **26 days** | **Tue 11/02/25** | **Tue 18/03/25** |
| 32 | | 6.1 In Main: .ino file create function: int FindClosestObject(){}; int GuideLeftHand(){}; int | 17 days | Tue 11/02/25 | Wed 05/03/25 |
| 33 | | 6.2 Main "ino"file write function: int PathAssistance(){}; | 7 days | Thu 06/03/25 | Fri 14/03/25 |
| 34 | | 6.3 Test Run Final and Uploaded Code for Start-Up Functionality | 2 days | Mon 17/03/25 | Tue 18/03/25 |
| 35 | | **7. Testing Phase** | **20 days** | **Wed 19/03/25** | **Tue 15/04/25** |
| 36 | | 7.1 Identityfy All Possible Test Locations And Get 1 or 2 Test Subjects | 1 day | Wed 19/03/25 | Wed 19/03/25 |
| 37 | | 7.2 Construct Cusions and Obtain Bicycle Helmet for Safety | 5 days | Thu 20/03/25 | Wed 26/03/25 |
| 38 | | 7.3 Go to Test Sites and Perform Path Assistance Testing | 4 days | Thu 27/03/25 | Tue 01/04/25 |
| 39 | | 7.4 Test Object Guidance on Tables, Doors and Paths | 4 days | Wed 02/04/25 | Mon 07/04/25 |
| 40 | | 7.5 Final Refinement, Tuning and Project Completion | 6 days | Tue 08/04/25 | Tue 15/04/25 |
| 41 | | **8. Project Report and Presentation** | **30 days** | **Tue 15/04/25** | **Mon 26/05/25** |
| 42 | | 8.1 Gathering All Required Information | 2 days | Tue 15/04/25 | Wed 16/04/25 |
| 43 | | 8.2 Report Writing and Finalisation | 13 days | Thu 17/04/25 | Mon 05/05/25 |
| 44 | | 8.3 Presentation Slides Writing and Finalisation | 10 days | Tue 06/05/25 | Mon 19/05/25 |
| 45 | | 8.4 Report Completion and Submission | 2 days | Tue 20/05/25 | Wed 21/05/25 |
| 46 | | 8.5 Presentation Completion Submissior | 3 days | Thu 22/05/25 | Mon 26/05/25 |
| 47 | | **9. Project Defense** | **10 days** | **Tue 27/05/25** | **Mon 09/06/25** |



**Figure 7-1: Initial Gantt Chart**



| ID | Task Mode | Task Name | Duration | Start | Finish |
|---|---|---|---|---|---|
| 0 | | **Shenin Neleesha Nelundeniya** | **172 days** | **Mon 14/10/24** | **Mon 09/06/25** |
| 1 | | **VisionGuard: Wearable Assistive Device for the Visually Impaired** | **172 days** | **Mon 14/10/24** | **Mon 09/06/25** |
| 2 | | **1. Project Distribution** | **1 day** | **Mon 14/10/24** | **Mon 14/10/24** |
| 3 | | **2. Project Plan & Feasibility Studies** | **14 days** | **Mon 14/10/24** | **Thu 31/10/24** |
| 4 | | 2.1 Identifying and Finalising Scope of the Project | 4 days | Mon 14/10/24 | Thu 17/10/24 |
| 5 | | 2.2 Referring Similar Research Resource Papers | 2 days | Fri 18/10/24 | Mon 21/10/24 |
| 6 | | 2.3 Identifying System and Component Structures | 4 days | Tue 22/10/24 | Fri 25/10/24 |
| 7 | | 2.4 Research on Risk and Assessment | 1 day | Mon 28/10/24 | Mon 28/10/24 |
| 8 | | 2.5 Hardware and Software Research | 8 days | Tue 22/10/24 | Thu 31/10/24 |
| 9 | | **3. System Modelling and Resource Gathering** | **14 days** | **Fri 01/11/24** | **Wed 20/11/24** |
| 10 | | 3.1 Model System | 2 days | Fri 01/11/24 | Mon 04/11/24 |
| 11 | | 3.2 Construct Hardware Connections | 1 day | Tue 05/11/24 | Tue 05/11/24 |
| 12 | | 3.3 Purchase and Obtain Hardware Components | 12 days | Tue 05/11/24 | Wed 20/11/24 |
| 13 | | 3.4 Software Downloads | 4 days | Tue 05/11/24 | Fri 08/11/24 |
| 14 | | 3.5 Environment Setup and Finalisation | 7 days | Tue 05/11/24 | Wed 13/11/24 |
| 15 | | **4. Path Assistance Coding and Connection** | **21 days** | **Sat 21/12/24** | **Fri 17/01/25** |
| 16 | | 4.1 3D Model Frame with Time-of-Fight Mounts | 5 days | Mon 23/12/24 | Fri 27/12/24 |
| 17 | | 4.2 Mount Time-of-Flight Sensors to Frame and Write Test Code | 1 day | Mon 30/12/24 | Mon 30/12/24 |
| 18 | | 4.3 Mount Motors to Frame and Write Test Code | 1 day | Tue 31/12/24 | Tue 31/12/24 |
| 19 | | 4.4 Gather Intensity Values Based on the Distance | 2 days | Wed 01/01/25 | Thu 02/01/25 |
| 20 | | 4.5 Write Code for Mapping Intensity Settings | 4 days | Fri 03/01/25 | Wed 08/01/25 |
| 21 | | 4.6 Finalise and Complete Coding of the Path Assistnace Algorithm | 10 days | Fri 03/01/25 | Thu 16/01/25 |
| 22 | | **5. Object Detection and Object Guidance** | **38 days** | **Wed 19/03/25** | **Fri 09/05/25** |
| 23 | | 5.1 Gather and Label Training Data | 7 days | Wed 19/03/25 | Thu 27/03/25 |
| 24 | | 5.2 Find and Test 4 Object Detection Mc | 5 days | Fri 28/03/25 | Thu 03/04/25 |
| 25 | | 5.3 Train and Optimise Best Models | 10 days | Fri 04/04/25 | Thu 17/04/25 |
| 26 | | 5.4 Upload Model to AI Module and Write Test Code for Data Extraction from AI Module | 3 days | Fri 18/04/25 | Tue 22/04/25 |
| 27 | | 5.5 3D Model Mount and Coverings for Camera Module, AI Module, XIAO ESP32C3, Sound Board, Power Cell and Speaker Module | 7 days | Wed 23/04/25 | Thu 01/05/25 |
| 28 | | 5.6 3D Print, Wiring and Mount Module: | 1 day | Fri 02/05/25 | Fri 02/05/25 |
| 29 | | 5.7 Write Code for Text-to-Speech | 4 days | Fri 02/05/25 | Wed 07/05/25 |
| 30 | | 5.8 Write Functions: int getBoundingBoxes[](); int | 2 days | Thu 08/05/25 | Fri 09/05/25 |
| 31 | | **6. Integration of Modules** | **26 days** | **Fri 11/04/25** | **Fri 16/05/25** |
| 32 | | 6.1 In Main: .ino file create function: int FindClosestObject[](); int GuideLeftHand()[]; int GuideRightHand()[]; | 17 days | Fri 11/04/25 | Mon 05/05/25 |
| 33 | | 6.2 Main "ino"file write function: int PathAssistance()[]; | 7 days | Tue 06/05/25 | Wed 14/05/25 |
| 34 | | 6.3 Test Run Final and Uploaded Code for Start-Up Functionality | 2 days | Thu 15/05/25 | Fri 16/05/25 |
| 35 | | **7. Testing Phase** | **23 days** | **Thu 01/05/25** | **Sun 01/06/25** |
| 36 | | 7.1 Identityfy All Possible Test Locations And Get 1 or 2 Test Subjects | 1 day | Thu 01/05/25 | Thu 01/05/25 |
| 37 | | 7.2 Construct Cusions and Obtain Bicycle Helmet for Safety | 5 days | Fri 02/05/25 | Thu 08/05/25 |
| 38 | | 7.3 Go to Test Sites and Perform Path Assistance Testing | 4 days | Fri 09/05/25 | Wed 14/05/25 |
| 39 | | 7.4 Test Object Guidance on Tables, Doors and Paths | 4 days | Thu 15/05/25 | Tue 20/05/25 |
| 40 | | 7.5 Final Refinement, Tuning and Project Completion | 6 days | Wed 21/05/25 | Wed 28/05/25 |
| 41 | | **8. Project Report and Presentation** | **34 days** | **Sun 20/04/25** | **Wed 04/06/25** |
| 42 | | 8.1 Gathering All Required Information | 2 days | Mon 21/04/25 | Tue 22/04/25 |
| 43 | | 8.2 Report Writing and Finalisation | 13 days | Wed 23/04/25 | Fri 09/05/25 |
| 44 | | 8.3 Presentation Slides Writing and Finalisation | 10 days | Mon 12/05/25 | Fri 23/05/25 |
| 45 | | 8.4 Report Completion and Submission | 2 days | Mon 26/05/25 | Tue 27/05/25 |
| 46 | | 8.5 Presentation Completion Submissior | 3 days | Wed 28/05/25 | Fri 30/05/25 |
| 47 | | **9. Project Defense** | **1 day** | **Mon 09/06/25** | **Mon 09/06/25** |



**Figure 7-2: Final Gantt Chart**

## 7.2 Commercial Evaluation

Based on the feasibility study conducted as described in Chapter 01, a BOQ (Table 1.1) listing the costs of the components intended to be used for the project.

**Table 7-1: Initial Costs and Final Costs**

| **Component** | **Estimated Component Requirement and Costs** | | | **Final Component Requirement and Costs** | | |
|---|---|---|---|---|---|---|
| | **Specification** | **Units** | **Price (LKR)** | **Specification** | **Units** | **Price (LKR)** |
| AI Module | Seeed Studio Grove Vision AI Module V2 | 1 | 5500/- | Hailo-8 AI Accelerator | 1 | - |
| Microcontroller | Seeed Studio XIAO ESP32-C3 | 1 | 2200/- | Raspberry Pi 5 | 1 | - |
| Time-of-Flight Sensors | TOF200C Lidar Sensor | 5 | 3800/- | TOF200C Lidar Sensor | 5 | 3800/- |
| Camera Module | Raspberry Pi Camera V2 | 1 | 6500/- | Raspberry Pi Camera V2 | 1 | 6500/- |
| Vibrating Motors | 3.7V Mini Vibration Motors | 5 | 1000/- | 3.7V Mini Vibration Motors | 5 | 1000/- |
| Power Bank Module | Generic 7.4 Power Bank Module | 1 | 550/- | Generic 7.4 Power Bank Module | 1 | 550/- |
| Battery | 1000 mAH LiPo 701855 | 2 | 3000/- | 1000 mAH LiPo 701855 | 2 | 3000/- |
| Audio Board | LM386 | 1 | 350/- | LM386 | 1 | 350/- |
| Speakers | Notebook Speakers 8 Ohms | 2 | 1000/- | Notebook Speakers 8 Ohms | 2 | 1000/- |
| 3D Printing Materials | 3D Printed Frame and Bands | - | - | 3D Printed Frame and Bands | - | 2500/- |
| Miscellaneous Components | - | - | - | - | - | 5000/- |
| **Total** | | | 24,000/- | | | 23,800/- |

The actual cost for VisionGuard was lower compared to the estimated cost since the new components were already available. However, there were some additional costs for the miscellaneous components and the 3D printing materials which weren't included in the initial costs. This was an offset by the costs of components that were intended to be used, although the components were purchased outside the estimated cost. This enabled the project to be completed within the allocated budget limit.

Estimated budget for the project = 25,000/-
Estimated budget for the components = 24,000/-
Actual Budget for the components = 23,800/-
Deviation of components = [(Actual budget – Estimated budget)/ Estimated budget] * 100%
Deviation components = -0.83%

## *7.3 Scope Evaluation*

The scope of VisionGuard was to address mobility and object interaction challenges faced by visually impaired individuals in an indoor environment. The main aim of this projects was to design a wearable smart frame which performed path assistance and object detection. The two objectives were implemented within the provided time frame, budget, and technical constraints while providing a foundation for future developments.

The path finding module was successfully implemented using TOF sensors and vibration motors and through blindfolded trials real-time detection of obstacles up to 1.5m were detected. The object detection module's initial plan was to use Seed Grove Vision AI module unfortunately it had to be upgraded to a Raspberry Pi 5 with the Hailo-8 accelerator for better performance, allowing on-device execution of YOLOv11 for object detection.

Overall, this project remained within the defined scope achieving the two main objectives. The navigation and object detection based on closeness to the user's hands within an indoor setting was appropriate, given the complexity of AI processing on embedded devices. However, project met its technical objectives balancing feasibility and laid groundwork for future development.

# REFERENCES

1. WHO (2019) World Report on vision, World Health Organization. Available at: https://www.who.int/publications/i/item/world-report-on-vision (Accessed: 31 May 2025).
2. Khorrami-Nejad, M. et al. (2016) The impact of visual impairment on quality of life, Medical hypothesis, discovery & innovation ophthalmology journal. Available at: https://pmc.ncbi.nlm.nih.gov/articles/PMC5347211/ (Accessed: 31 May 2025).
3. Lin, K.W. et al. (2012) ‘A wearable stereo vision system for visually impaired’, 2012 IEEE International Conference on Mechatronics and Automation, pp. 1423–1428. doi:10.1109/icma.2012.6284345.
4. Pandit, S. (2024) The evolution of low vision technology: Past, present, and future, Medium. Available at: https://medium.com/@sanikapandit1997/the-evolution-of-low-vision-technology-past-present-and-future-56c7dfc14554#:~:text=Some%20notable%20milestones%20include%3A,revolutionizing%20literacy%20for%20the%20blind. (Accessed: 31 May 2025).
5. IJRCIT (2023) IJRCIT. Available at: https://www.ijrcit.co.in/downloads/volume_8/issue_3/23.pdf (Accessed: 31 May 2025).
6. Vision Aid (2024) Smart vision glass: Elevate your sight: Vision aid, Vision. Available at: https://visionaid.org/about-smart-vision-glasses/ (Accessed: 31 May 2025).
7. Shih, M.-L. et al. (2018) ‘DLWV2: A deep learning-based wearable vision-system with vibrotactile-feedback for visually impaired people to reach objects’, 2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 1–9. doi:10.1109/iros.2018.8593711.
8. Li, Z. et al. (2020) ‘A wearable device for indoor imminent danger detection and avoidance with region-based ground segmentation’, IEEE Access, 8, pp. 184808–184821. doi:10.1109/access.2020.3028527.
9. Moghimi, M. et al. (2014) ‘Experiments on an RGB-D wearable vision system for egocentric activity recognition’, 2014 IEEE Conference on Computer Vision and Pattern Recognition Workshops, pp. 611–617. doi:10.1109/cvprw.2014.94.
10. Bastola, Ashish et al. (2023) ‘Multi-functional glasses for the blind and visually impaired: Design and development’, Proceedings of the Human Factors and Ergonomics Society Annual Meeting, 67(1), pp. 995–1001. doi:10.1177/21695067231192450.
11. Khan, M.A. et al. (2020) ‘An AI-based visual aid with integrated reading assistant for the completely blind’, IEEE Transactions on Human-Machine Systems, 50(6), pp. 507–517. doi:10.1109/thms.2020.3027534.
12. Aye, H.N., Naing, A.T. and Tun, H.M. (2016) Ijstr. Available at: https://www.ijstr.org/final-print/june2016/Assistive-Device-For-Visually-Impaired-People-By-Using-Ultrasonic-Sensors.pdf (Accessed: 31 May 2025).
13. Kusuma, H., Attamimi, M. and Sintara, J. (2023) ‘Navigation Aid Device for visually impaired using depth camera’, Advances in Science, Technology and Engineering Systems Journal, 8(3), pp. 48–53. doi:10.25046/aj080306.

14. Al Maraashli, M. et al. (2025) A novel head-mounted time-of-flight sensor array for the visually impaired to enhance real-time spatial awareness [Preprint]. doi:10.20944/preprints202504.0678.v1.
15. Swami, S. et al. (2025) ‘An affordable low-cost wearable solution for object detection in visual impairment’, EPJ Web of Conferences, 325, p. 01017. doi:10.1051/epjconf/202532501017.
16. SeeedStudio (no date) Getting started with Seeed Studio Xiao ESP32C3: Seeed Studio Wiki, Seeed Studio Wiki RSS. Available at: https://wiki.seeedstudio.com/XIAO_ESP32C3_Getting_Started/ (Accessed: 02 June 2025).
17. Grobotronics (no date) Vibration Motor Module, Grobotronics. Available at: https://grobotronics.com/arduino-vibration-motor-module.html?sl=en (Accessed: 02 June 2025).
18. Design, C. (no date) Cirkit designer tutorials, How to Use PWM Vibration Motor Sensor Module Switch: Pinouts, Specs, and Examples | Cirkit Designer. Available at: https://docs.cirkitdesigner.com/component/d47baacb-0253-4e2b-bef4-68dfed7a7bf6/pwm-vibration-motor-sensor-module-switch (Accessed: 02 June 2025).
19. Roboflow. (no date) *Deploy: Serverless Object Detection* [Online]. Available at: https://docs.roboflow.com/deploy/serverless/object-detection (Accessed: 2 June 2025).
20. Ultralytics. (no date) *Modes: Train* [Online]. Available at: https://docs.ultralytics.com/modes/train/ (Accessed: 2 June 2025).

# BIBLIOGRAPHY

1. Ada Computer Science. (no date) Pathfinding [Online]. Available at: *https://adacomputerscience.org/topics/pathfinding* (Accessed: 2 June 2025).
2. Autodesk. (no date) Fusion 360: 3D Modeling Software [Online]. Available at: *https://www.autodesk.com/solutions/fusion-360-3d-modeling* (Accessed: 2 June 2025).
3. Roboflow. (2022) Getting Started with Roboflow [Online]. Available at: *https://blog.roboflow.com/getting-started-with-roboflow/* (Accessed: 2 June 2025).
4. Viso.ai. (no date) Object Detection: The Ultimate Guide [Online]. Available at: *https://viso.ai/deep-learning/object-detection/* (Accessed: 2 June 2025).

# APPENDIX A: Datasheets

**XIAO ESP32C3 Microcontroller**

XIAO ESP32-C3 Microcontroller (no date) *Seeed studio xiao series som datasheet*. Available at: https://files.seeedstudio.com/wiki/XIAO/Seeed-Studio-XIAO-Series-SOM-Datasheet.pdf (Accessed: 02 June 2025).

**VL53L0X Time-of-Flight Ranging Sensor**

VL53L0X Time-of-Flight ranging sensor (no date) *Datasheet - VL53L0X - time-of-flight ranging sensor*. Available at: https://www.st.com/resource/en/datasheet/vl53l0x.pdf (Accessed: 02 June 2025).

**PWM Vibration Motor**

Vibration Motor Data Sheet (no date) *Vibration Motor Data Sheet*. Available at: https://education.ti.com/html/webhelp/EG_Innovator/EN/content/eg_innovsys/m_io-datasheets/io_ds_vibrationmotor.HTML (Accessed: 02 June 2025).

**Adafruit Servo Motor**

Adafruit. (2014) *PCA9685 16-channel, 12-bit PWM FM+ I2C-bus LED controller* [Online]. Available at: https://cdn-shop.adafruit.com/datasheets/PCA9685.pdf (Accessed: 2 June 2025).

**Raspberry Pi Camera Module V2**

Farnell. (2015) *Raspberry Pi Foundation Raspberry Pi 3 Model B* [Online]. Available at: https://www.farnell.com/datasheets/2056179.pdf?_ga=1.152577328.880870297.1479740269 (Accessed: 2 June 2025).

**Raspberry Pi 5**

Raspberry Pi Foundation. (2023) *Raspberry Pi 5 Product Brief* [Online]. Available at: https://datasheets.raspberrypi.com/rpi5/raspberry-pi-5-product-brief.pdf (Accessed: 2 June 2025).

**Hailo-8 AI Accelerator**

Hailo. (2023b) *Hailo-8 M.2 Key B_M Datasheet Rev1.1* [Online]. Available at: https://premio.blob.core.windows.net/premio/uploads/resource/data-sheet/Hailo-8/Hailo-8%20M.2%20Key%20B_M%20Datasheet%20Rev1.1.pdf (Accessed: 2 June 2025).

# APPENDIX B: Codes

**Path Assistance Module Code**

https://github.com/SheninNeleesha/VisionGuard/blob/main/PathFinder.ino

**Object Detection Module Code**

https://github.com/SheninNeleesha/VisionGuard/blob/main/VisionGuard.py

# APPENDIX C: 3D Model Design

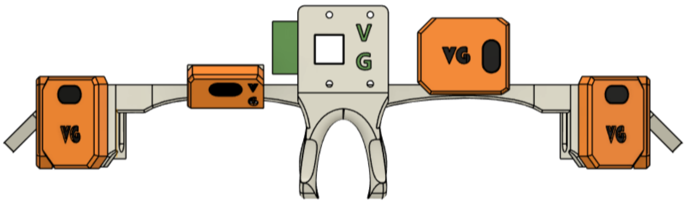


**Figure 0-1: Front View of VisionGuard Frame**

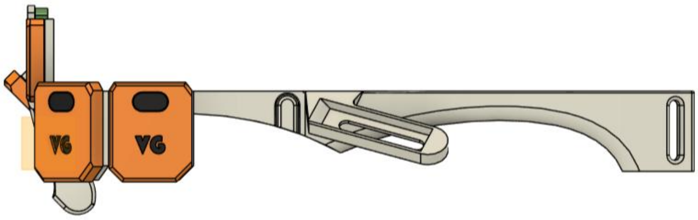


**Figure 0-2: Right Side View of Vision Guard Frame**

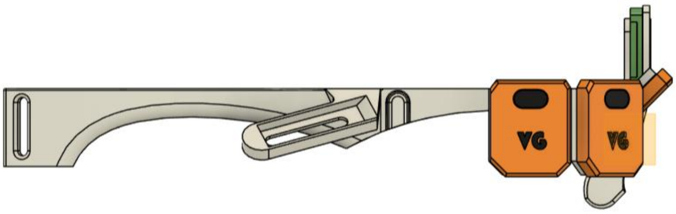


**Figure 0-3: Left Side View of VisionGuard Frame**

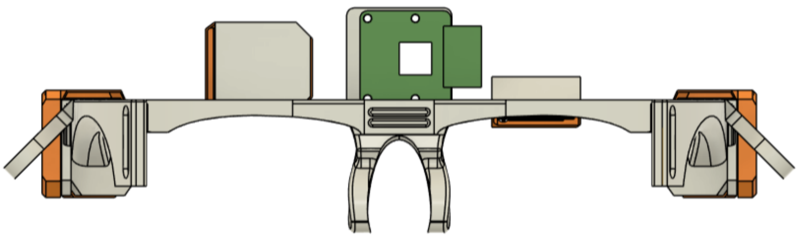


**Figure 0-4: Back View of VisionGuard frame**