LIVERPOOL JOHN MOORES UNIVERSITY

Department of **Electronics & Electrical Engineering**

# Smart Cane for Visually Impaired

## Dissertation

## Ali Gado

**Supervisor: Qian Zhang**

**2nd Marker: Dingli Yu**

**Acknowledgments:**

I would like to express my sincere gratitude to my supervisor, **Dr. Qian Zhang**, for his invaluable guidance, support, and encouragement throughout the duration of this project. His insightful feedback and technical expertise were instrumental in shaping the direction and success of this work.

My heartfelt thanks also go to our dedicated laboratory technicians, **Stephen Murphy** and **Asmaa Al Tameemi**, whose continuous assistance and professionalism in sourcing components and providing hands-on lab support greatly contributed to the smooth execution of the project.

A special thank you is extended to the mechanical engineering technicians, particularly **Thomas Byrne**, for his support and precision in 3D printing the custom enclosure design. His attention to detail and technical skill ensured the physical aspects of the project were completed to a high standard.

I am truly grateful to all the academic and technical staff who offered guidance and support during this journey. Their contributions, whether direct or indirect, have been deeply appreciated.

**Abstract:**

Navigating safely and independently is a significant challenge for visually impaired individuals, often requiring external assistance or reliance on traditional white canes. This project presents the development of a smart cane designed to enhance mobility and obstacle detection using ultrasonic sensing and haptic feedback. The cane integrates three ultrasonic

sensors, enabling a 270-degree field of detection to identify obstacles in the user's path. When an obstacle is detected, the system provides directional haptic feedback through vibrations on the left, right, or front handle sections. If the user continues moving toward a detected obstacle, an auditory warning (buzzer) is activated to further alert them to potential danger. Additional features include an LED indicator for low-light conditions and a rechargeable battery to ensure prolonged usability. The system is designed for real-time responsiveness, compact integration, and user-friendly operation. By improving environmental awareness and reducing collision risks, this smart cane aims to enhance autonomy and safety for visually impaired users.

**Appendix**

**Acknowledgements**

**Abstract**

**Chapter 1: Introduction**



**Figures in Chapter 1:**



**Chapter 2: Literature Review**



**Figures in Chapter 2:**

**Chapter 3: Methodology**



**Figures in Chapter 3:**

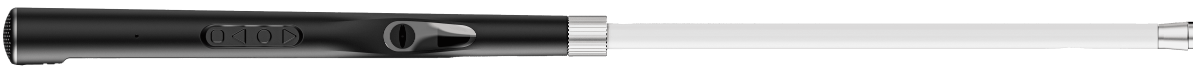
# 1 Introduction

## 1.1 Background

Navigating the environment as a visually impaired person presents considerable problems, necessitating aids that improve mobility and safety. Historically, white canes have been the most popular mobility aid for the visually impaired due to its low cost, ease of use, and dependability. However, their lack of modern technical features restricts their capacity to deliver real-time input on dynamic impediments and environmental threats. To solve these constraints, smart canes have emerged as a viable option.

Existing smart canes, such as WeWALK and UltraCane (Fig. 1), employ ultrasonic sensors and feedback systems to help users recognize impediments (WeWalk, 2024; Ultracane, 2023). While these solutions show the usefulness of assistive technology, they have significant limitations, such as expensive prices, restricted obstacle detection ranges, and the lack of multi-modal feedback systems. Furthermore, wearable assistive technologies like as OrCam MyEye provide camera-based navigation and object identification (Katie, 2021; Pauls & Pauls, 2018), but their high cost and complexity dissuade users who prefer a simpler, cane-based method.

Fig 1:



To bridge the gap between affordability and technical innovation, this project introduces the Smart Assistive Navigation Cane, a low-cost, user-friendly solution that combines ultrasonic sensors, vibration motors, and an aural feedback system. By emphasizing simplicity and usefulness, the smart cane strives to improve the freedom and safety of visually impaired people.

## 1.2 Aims

The major aim of this project is to create and develop a Smart Assistive Navigation Cane that enhances the mobility and navigation experience for visually impaired people. The cane is equipped with ultrasonic sensors to identify impediments, vibration motors to offer haptic feedback, and an audio alarm system. By emphasizing accessibility, affordability, and efficiency, this initiative aims to deliver an innovative yet practical solution for improving consumers' quality of life.

## 1.3 Objectives

To accomplish the desired goal, the following objectives have been established:

1. Research and Analysis: Conduct a thorough analysis of current mobility aids for the visually impaired, such as standard white canes, smart canes, and wearable assistive gadgets, to identify limits and opportunities for improvement.
2. Component Selection and Cost Optimization: To ensure affordability while retaining performance, choose cost-effective yet efficient components such as ultrasonic sensors, vibration motors, and speakers. The objective is to keep the entire build cost around £40.
3. Hardware Development: Design and build the smart cane case from durable and lightweight materials to ensure its versatility across various terrains.
4. Sensor Integration and Functionality Testing: Install ultrasonic sensors for obstacle detection, program vibration motors to provide haptic feedback, and incorporate an auditory feedback system. Run several tests to improve sensitivity, range, and responsiveness.
5. User Testing and Validation: Wear a blindfold and test the cane in different locations to determine its usefulness in helping navigation.
6. Power Management and Battery Efficiency: Optimize the smart cane's power usage by using energy-efficient components and rechargeable battery options. Analyse charging times and total battery life to guarantee long-term usage with little maintenance.
7. Comparison with Existing Solutions: Evaluate the Smart Assistive Navigation Cane's performance against standard and smart canes, emphasizing its cost-effectiveness, usability, and efficiency.
8. Final Prototype Development and Refinement: Based on user input and test results, make any required changes to improve the cane's functionality and overall user experience.

### 1.4 Report Structure

The report is organized as follows:

Chapter 1: Introduction - Includes background information, project goals and objectives, and an explanation of the report structure.

Chapter 2: Literature Review - Examines current mobility aids, their limitations, and technical advances in smart navigation canes.

Chapter 3: Methodology - describes the smart cane's design approach, component selection, and development stages.

Chapter 4: Implementation and Testing describes hardware integration, software development, and experimental results from real-world testing.

Chapter 5: Outcomes and Discussion - Analyses findings, assesses performance, and compares outcomes to other assistive devices.

Chapter 6: Conclusion and Future Work - Summarizes important successes, highlights areas for improvement, and offers future research possibilities.

### 1.5 Summary

This project proposes a Smart Assistive Navigation Cane, which is intended to assist mobility and navigation for visually impaired users. The cane uses ultrasonic sensors, vibration motors, and a speaker system to deliver real-time feedback on nearby impediments. The project's goal is to create a cheap, user-friendly, and energy-efficient assistive gadget that promotes independence and safety. The chapters that follow will offer a complete description of the smart cane's research, design, and development process, as well as examples of its usefulness in real-world applications.

## Chapter 2: Literature review

### 2.1 Introduction

This chapter gives a thorough examination of current mobility aids tailored for people with vision impairments. It investigates classic white canes, different smart cane technologies, and wearable assistive devices, emphasizing their benefits, drawbacks, and importance to the creation of an inexpensive, efficient, and user-friendly smart cane.

## 2.2 Traditional White Canes

The white cane has long served as the traditional mobility tool for vision impaired people, providing tactile feedback about their surroundings.

Pros - Affordability and Accessibility:

- White canes are affordable and widely available, making them accessible to a wide range of users.
- Simplicity: Their basic design necessitates minimal training and maintenance.

Cons - Limited Detection Range:

- Because traditional canes detect impediments largely via physical touch, they may not give adequate warning for objects located further away or above ground level.
- Lack of Advanced Features: They do not provide real-time feedback for dynamic impediments or environmental changes.

Relevance - The simplicity and affordability of white canes highlight the need of retaining these characteristics in the development of improved mobility devices. Enhancements should strive to enhance, not make more complicated the user experience.

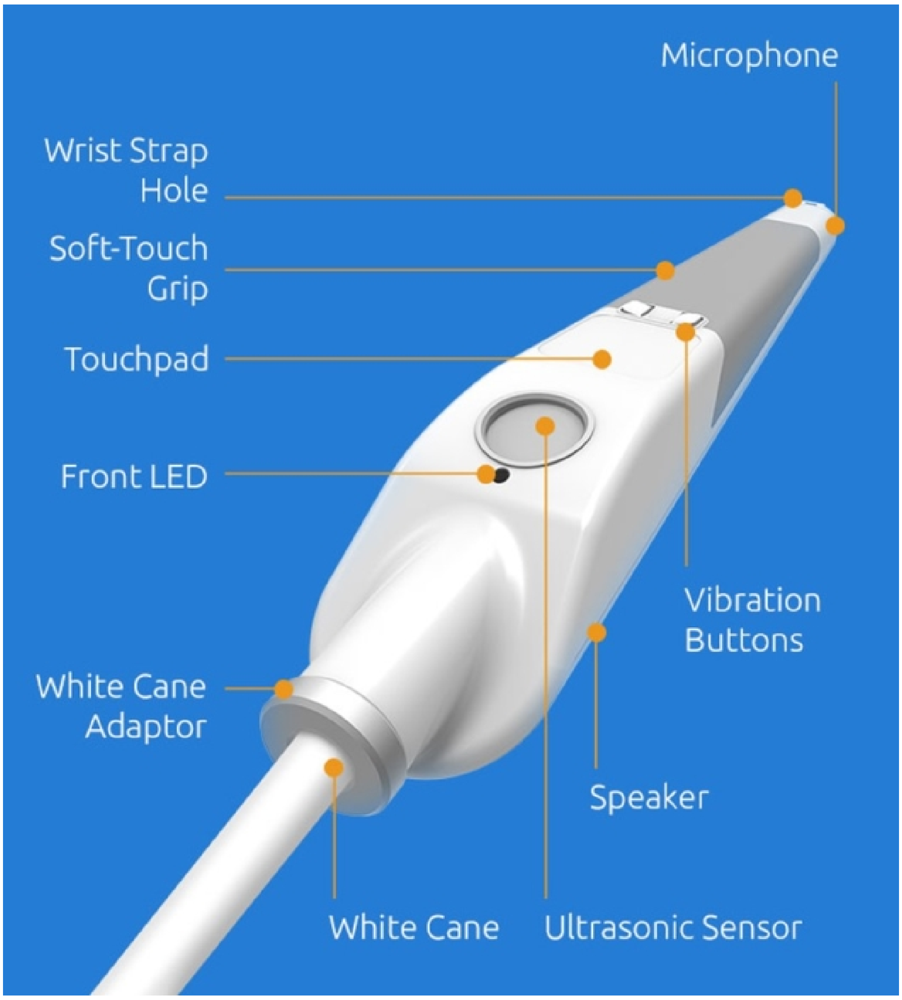
## 2.3 Existing Smart Canes

Advancements in technology have resulted in the creation of smart canes that incorporate electrical components to improve navigational aid.

Notable instances are: WeWALK Smart Cane: With ultrasonic sensors, an accelerometer, gyroscope, speakers, and a compass, WeWALK recognizes obstructions and helps you navigate. It may be added to any regular cane and provides capabilities accessible by buttons, a touchpad, or by manipulating the cane itself (Staff, 2019).

Fig 2: WeWalk features

6455ELE Smart Cane for Visually Impaired




UltraCane: Detects obstructions using ultrasonic sensors and offers haptic feedback via vibrating buttons on the handle to indicate the direction of the impediment.

Pros - Enhanced Obstacle Detection:

- These devices can identify impediments beyond the physical reach of ordinary canes, such as those at head or chest level.
- Integrated Feedback Mechanisms: They provide users with multimodal feedback, such as auditory and tactile signals, to help them understand their environment.

Cons - High Cost:

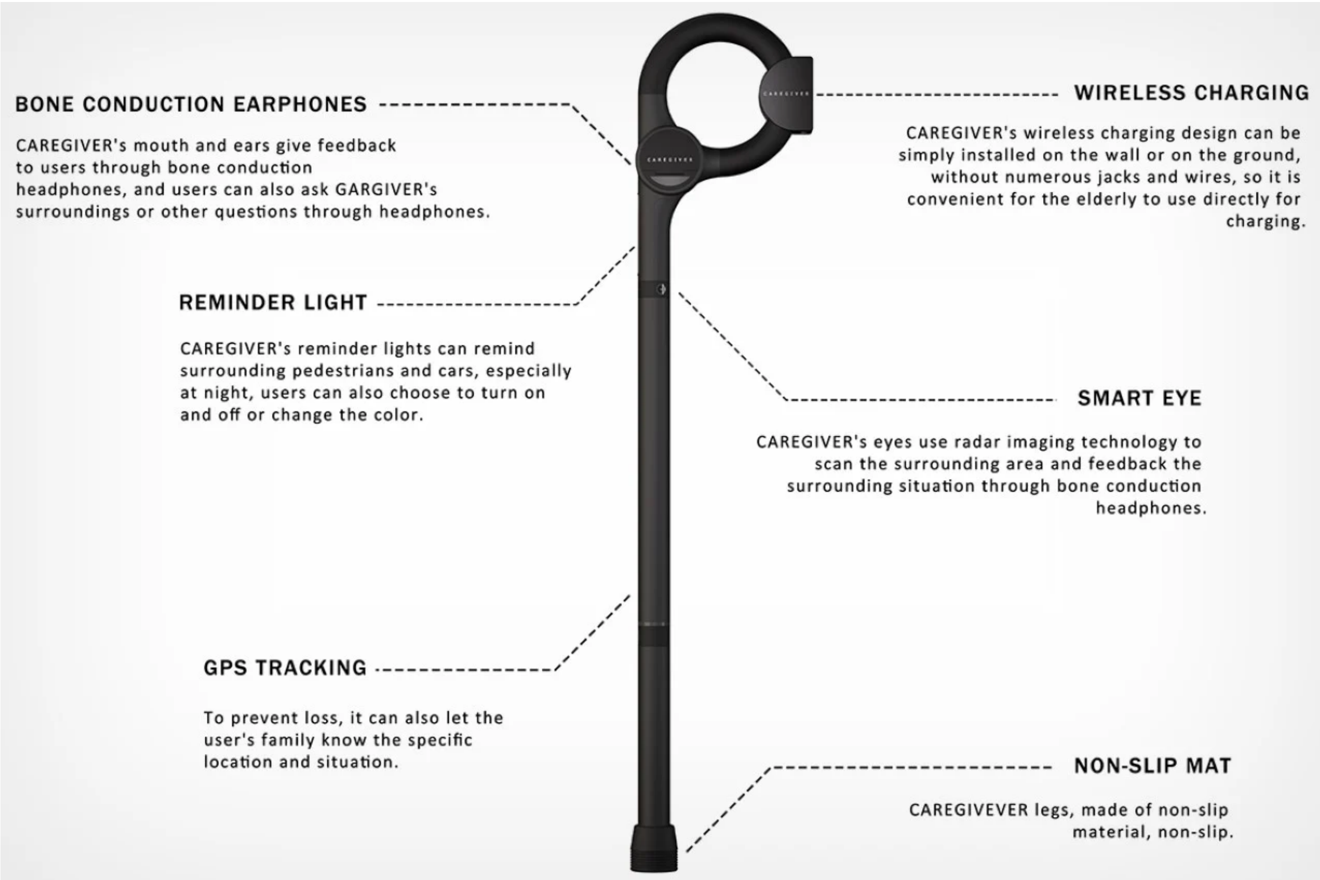
- The advanced technology included into these canes frequently results in a higher price point, which may limit accessibility for some users.
- Complexity: The presence of various functions may necessitate more training and overwhelm users who prefer simpler tools.

Ali Gado 1021438

- Limited Customization: Some smart canes lack modular capabilities that enable users to tailor feedback systems to their preferences.

Relevance: These smart canes illustrate how technology can improve movement for the visually handicapped. However, these shortcomings underscore the need for solutions that strike a compromise between enhanced capability, price, and usability.

Fig 3: Other Smart Cane features




2.4 Wearable Assistive Devices

Beyond canes, wearable solutions have been created to help visually impaired people. For example, OrCam MyEye uses an eyeglass-mounted camera to evaluate visual information and communicate it to the user verbally (Subudhi et al., 2020).

Fig 4: MyEye product

Pros - Advanced Capabilities:

- These gadgets have complex functions like text reading and facial recognition, which provide comprehensive support beyond navigation.
- Cons: Extremely High Cost: Standing at £4.5k, the intricacy of these systems frequently results in considerable costs, making them less accessible.
  Some users may choose more uncomplicated, cane-based alternatives to wearable technology.

Relevance: While wearable gadgets demonstrate the potential of assistive technology, their limits highlight the significance of creating inexpensive, simple, and effective mobility aids such as smart canes.

Fig 5: Outdoor Obstacle detection

Fig 6: Wearable Assistive Devices Technology

| Sr. No. | Name of Paper | Year and Publication | Technology used | Results |
|---|---|---|---|---|
| 1 | Development of an Intelligent Guide-Stick for the Blind<br>Sung Jae Kang1, Young Ho, Kim1, In Hyuk Moon2 | 2001 IEEE | Servomotor<br>Microcontroller<br>ultrasonic sensors | Simulation Results |
| 2 | Bioinspired Electronic White Cane Implementation Based on a LIDAR, a Tri-Axial Accelerometer and a Tactile Belt<br>TomàsPallejà, Marcel Tresanchez, MercèTeixidó and Jordi Palacin | *Sensors* 2010 | LIDAR;<br>accelerometer;<br>tactile | Test field Results |
| 3 | The Smart Vision Local Navigation Aid for Blind and Visually Impaired Persons<br>João José, Miguel Farrajota, João M.F. Rodrigues, J.M. Hans du Buf | International Journal of Digital Content Technology and its Applications Vol.5 No.5, May 2011 | Image Processing | Simulation Results |
| 4 | Optical Device Indicating a Safe Free Path to Blind People<br>Joselin Villanueva, *Student Member, IEEE*, and René Farcy | IEEE2012 | Microcontroller<br>IR sensor<br>Ultrasonic sensor | Test field Results |
| 5 | Enhanced Independence Free Path Detector To Blind People Using GSM<br>Mr.B.Anbazhagan1, Mr.V.Nandagopal | April 2013 IJAREEIE | LED Photodiode<br>IR device | Simulation Results |
| 6 | Guiding A Safe Free Path To Visually Impaired Person<br>Ashok kumar.R1, P.Venkat Rao | September 2013 IJIEEE | ARM controller<br>2D Robot<br>Smart Phone<br>GPS | Computer vision Application & Simulation Results |
| 7 | New electronic white cane for stair case detection and recognition using ultrasonic sensor<br>Sonda Ammar Bouhamed, ImeneKhanfirKallel, DorraSellamiMasmoudi | IJACSA 2013 | Monocular camera<br>Microcontroller<br>Ultrasonic sensor<br>Bluetooth | Test field Results And Simulation |
| 8 | Wearable Obstacle Detection System for visually impaired People<br>Sylvain Cardin, Daniel Thalmann and Frederic Vexo | Springer, pp 331-349, 2010 | Microcontroller<br>Sonar sensor | Test field Results |

## 2.5 User Preferences and Needs

Visually impaired people rely extensively on sensory feedback; hence it is critical for assistive technology to provide intuitive and non-invasive advice. Studies show that tactile input, such as vibrations, is one of the most efficient ways to deliver navigation help owing to its discretion and dependability. Audio feedback is also useful, especially in dynamic contexts; however, it must be used cautiously to avoid sensory overload. Existing assisted navigation systems, such as WeWALK and UltraCane, use comparable technology but have high pricing and limited functionality. The Smart Assistive Navigation Cane, which incorporates ultrasonic sensors, vibration motors, and an aural feedback system, provides a cost-effective and user-friendly alternative.

## 2.6 Components Used in the Project

### 2.6.1 Piezo Speaker

The piezoelectric speaker, often known as a buzzer, gives audio feedback to the user. It functions by vibrating a piezoelectric element in response to an electrical pulse. This component is very effective for alerting the user about impediments using sound patterns corresponding to distance.

**Specifications:**

- Operating Voltage: 3V – 12V
- Resonant Frequency: 2 kHz – 4 kHz
- Output Sound: 85 dB – 100 dB at 10 cm

Comparison to Electromagnetic Buzzer: Piezo Speaker: Low power consumption, little size, and ideal for basic buzzer noises. Electromagnetic Buzzer: Consumes more power, produces a wider range of tones, but is larger.

### 2.6.2 Ultrasonic Distance Sensor

The HC-SR04 is a popular ultrasonic sensor that detects distance by transmitting and receiving ultrasonic waves. It is the Smart Cane's principal sensor component, detecting obstructions and providing real-time feedback.

**Specifications:**

- Operating Voltage: 5V DC
- Measuring Distance: 2 cm – 400 cm
- Accuracy: ± 3 mm
- Signal Frequency: 40 kHz Beam Angle: ~15 degrees

**Functionality:** The sensor generates an ultrasonic pulse that bounces off an item before returning to the sensor. The distance is calculated based on the time it takes for the pulse to return.

**Fig 7. Comparison with IR Sensor:**

| Feature | Ultrasonic (HC-SR04) | IR Sensor |
|---|---|---|
| Accuracy | High (±3 mm) | Moderate |
| Range | 2 cm – 400 cm | 2 cm – 150 cm |
| Light Sensitivity | Not affected | Affected by ambient Light |
| Cost | Low | Moderate |

Ultrasonic sensors are favored over infrared sensors because they work consistently in a variety of illumination situations and give superior precision.

### 2.6.3 Mini Vibration Motor

The vibration motor offers haptic feedback to notify the user to potential obstructions. When the ultrasonic sensor detects an item within a certain range, the motor is activated and vibrates at varied intensities depending on the distance.

**Specifications:**

- Operating Voltage: 3V – 5V
- Rated Speed: 12,000 – 15,000 RPM
- Vibration Amplitude: ~1.5G
- Power Consumption: 0.1W – 0.3W

**Fig 8. Comparison with Eccentric Rotating Mass (ERM) Motor**

| Feature | Mini Vibration Motor | ERM Motor |
|---|---|---|
| Power Consumption | Low | Moderate |
| Response Time | Fast | Moderate |
| Cost | Low | Higher |

Due to its low power consumption and quick response time, the mini vibration motor is an extraordinary component for this application.

### 2.6.4 Arduino Uno R3

The system's microcontroller, the Arduino Uno R3, processes sensor inputs and controls outputs such as vibrations and noises.

**Specifications:**

- Microcontroller: ATmega328P
- Operating Voltage: 5V
- Digital I/O Pins: 14
- Flash Memory: 32 KB
- Clock Speed: 16 MHz

Arduino was chosen because of its simplicity of programming, broad community support, and interoperability with a variety of sensors and modules.

#### 2.6.5 9V Battery

The system is powered by a 9V battery, which provides enough voltage to run the Arduino and its attached components.

#### 2.6.6 Red LED

A red LED serves as a visible signal of system status, notifying the user when the cane is turned on or detects an obstruction.

#### 2.6.7 Sideswitches

Side switches allow the user to toggle the smart functionality on and off, enabling them to use the cane as a traditional mobility aid when needed.

#### 2.6.8 Resistors (1kΩ and 400Ω)

Resistors are used to control current flow and safeguard circuit components. The 1kΩ resistor is utilized for pull-down in switches, while the 400Ω resistor is used in LED circuits.

### 2.7 Conclusion

The literature review looked at the user demands and preferences that informed the design of the Smart Assistive Navigation Cane. It also examined each component, outlining its specs and functional purpose within the system. Comparisons with other components were presented to substantiate component selection. The use of ultrasonic sensors, vibration motors, and an aural feedback system results in a reliable and economical navigation aid for visually impaired people. The following chapter will concentrate on the mathematical model that underpins the system's operation.

## Chapter 3: Methodology

### 3.1 Overview of approach

To achieve the goal and goals of designing an assistive navigational cane for the visually impaired, a systematic procedure was used. Starting with Tinkercad simulation and idea testing, then physical prototyping with the Arduino IDE, and finally a completely functioning hardware prototype incorporated into a 3D-printed design. This project is hardware and software demanding, with sensor integration, real-time feedback via vibration and audio output, and a completely integrated system controlled by an Arduino Uno R3. The process consisted of simulation, component sourcing, prototyping, wiring, debugging, 3D design and manufacture, code development, and iteration.

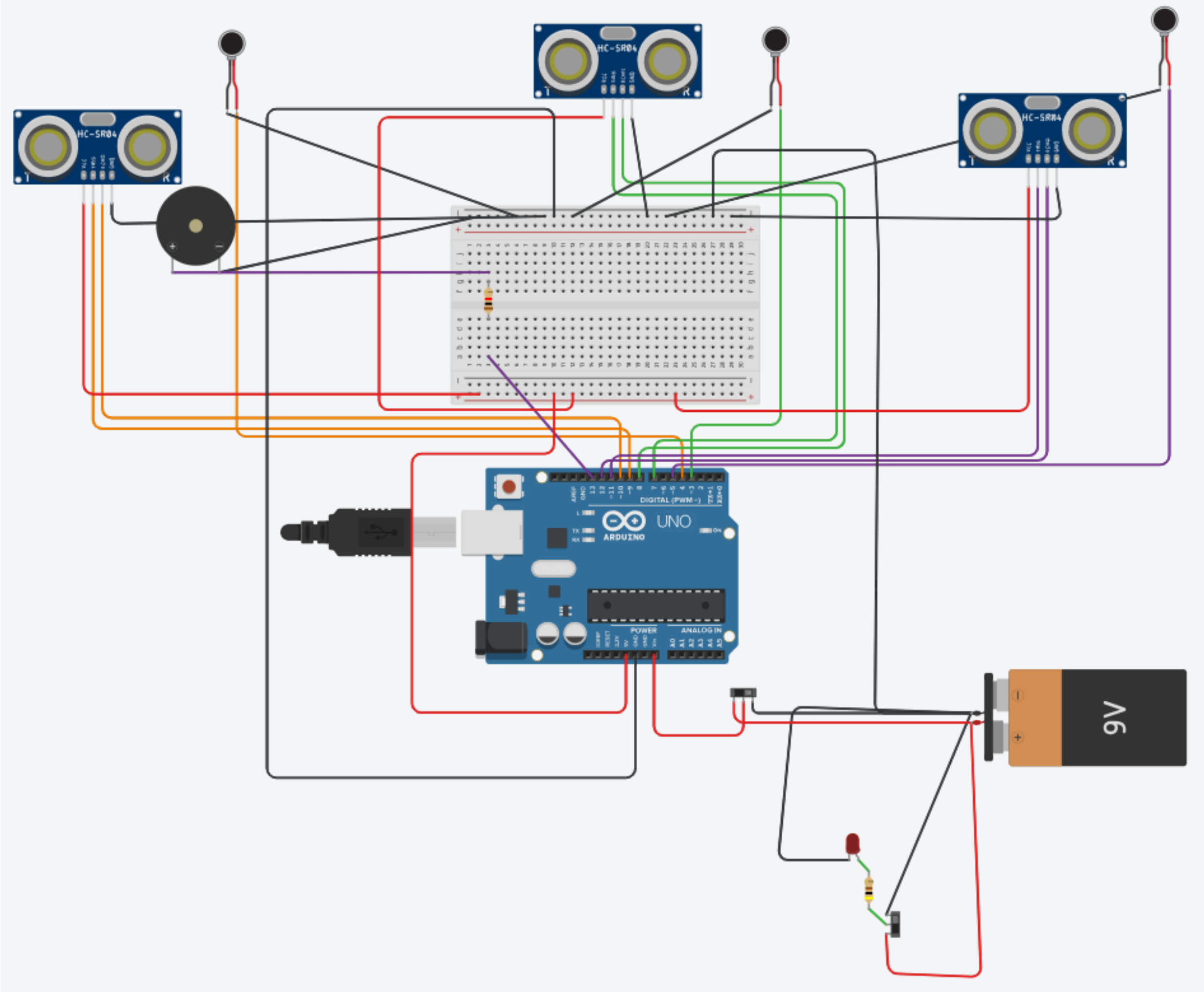
### 3.2 Software tools

Throughout the project's lifespan, a variety of simulation platforms, programming environments, and 3D modeling applications were employed to properly design and create a smart cane for visually impaired. Each instrument performed a particular role, depending on the stage of development and the nature of the work at hand.

#### 3.2.1 Tinkercad

Tinkercad, a browser-based electronics simulator, served as the project's foundation throughout the early prototype phase. Tinkercad was chosen because of its simplicity, accessibility, and simple drag-and-drop interface, which allows for rapid experimentation with components and circuit configurations. Given the limited physical resources at the time, Tinkercad allowed for the visualization and testing of circuit behaviour, such as how ultrasonic sensors responded to changes in object distance and how components like buzzers and vibration motors activated correspondingly. However, while Tinkercad is great for basic testing, it presented one important difficulty that slowed work. When the vibration motor was turned on, its visual depiction was extremely modest. It just showed weak wavy patterns that were scarcely discernible without zooming in. As a result, even though the programming and connections were operationally accurate, the motor seemed unresponsive throughout the simulation. This resulted in hours of useless troubleshooting, rechecking wire configurations, modifying pin settings, and rewriting code, all while the real problem was just the simulation's visual feedback limits. This episode, while disappointing, demonstrated the significance of challenging simulation constraints and considering real-world behaviour.

Fig 9. Tinkercad Design




3.2.2 Arduino IDE

Once the fundamental circuit behaviour was established in simulation, work progressed to the Arduino IDE, the official programming environment for Arduino microcontrollers. This platform was important for getting the real firmware onto the Arduino Uno board used in the physical prototype. The IDE had serial monitoring, a simple C++-like language, and seamless code uploading. It enabled real-time testing with serial output to debug distance measurements and control logic. During development, the Arduino IDE's simplicity proved useful, particularly for calibrating ultrasonic sensor sensitivity and optimizing logic for vibration feedback and buzzer alerts. Frequent usage of the Serial Monitor gave real-time input on sensor data, which was critical in matching theoretical distance thresholds with real-world obstacle detection accuracy.

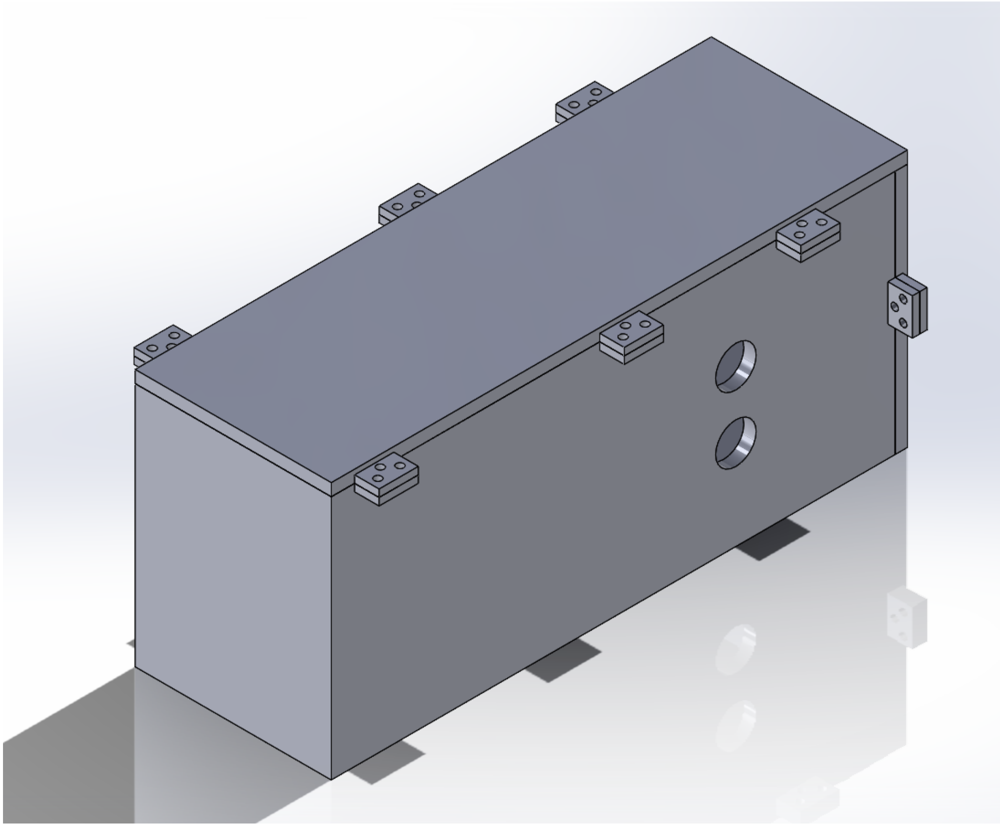
Fig 10. Arduino IDE code

```
// Define ultrasonic sensor pins
const int trigFront = 7, echoFront = 8;
const int trigLeft = 9, echoLeft = 10;
const int trigRight = 11, echoRight = 12;

// Define output components
const int buzzer = 13;
const int vibrationFront = 3;
const int vibrationLeft = 4;
const int vibrationRight = 5;

// Function to get distance from ultrasonic sensor
int getDistance(int trigPin, int echoPin) {
    digitalWrite(trigPin, LOW);
    delayMicroseconds(2);
    digitalWrite(trigPin, HIGH);
    delayMicroseconds(10);
    digitalWrite(trigPin, LOW);
    int duration = pulseIn(echoPin, HIGH);
    return duration * (0.034 / 2);
}

void setup() {
    Serial.begin(9600);

    // Set pin modes
    pinMode(trigFront, OUTPUT); pinMode(echoFront, INPUT);
    pinMode(trigLeft, OUTPUT); pinMode(echoLeft, INPUT);
    pinMode(trigRight, OUTPUT); pinMode(echoRight, INPUT);

    pinMode(vibrationFront, OUTPUT);
    pinMode(vibrationLeft, OUTPUT);
    pinMode(vibrationRight, OUTPUT);
    pinMode(buzzer, OUTPUT);
}

void loop() {
    // Read distances
    int distanceFront = getDistance(trigFront, echoFront);
    int distanceLeft = getDistance(trigLeft, echoLeft);
    int distanceRight = getDistance(trigRight, echoRight);

    Serial.print("Front: "); Serial.print(distanceFront);
    Serial.print(" cm | Left: "); Serial.print(distanceLeft);
    Serial.print(" cm | Right: "); Serial.println(distanceRight);

    // Vibration motors logic
    digitalWrite(vibrationFront, distanceFront < 40 ? HIGH : LOW);
    digitalWrite(vibrationLeft, distanceLeft < 40 ? HIGH : LOW);
    digitalWrite(vibrationRight, distanceRight < 40 ? HIGH : LOW);

    // Buzzer logic: gradual intensity
    if (distanceFront < 20 || distanceLeft < 20 || distanceRight < 20) {
        int buzzerFrequency = map(min(distanceFront, min(distanceLeft, distanceRight)), 0, 20, 1000, 200);
        tone(buzzer, buzzerFrequency);
    } else {
        noTone(buzzer);
    }

    delay(100);
}
```

### 3.2.3 SolidWorks

The third significant software tool used was SolidWorks, which is a professional-grade 3D modelling and CAD program. Its involvement was critical in developing the physical container that held all the electrical components. However, unlike Tinkercad or the Arduino IDE, SolidWorks required a significant learning curve, particularly as it was being utilized for the first time in this project. A large amount of time was spent self-learning via tutorials and trial-and-error to create a practical and ergonomic cane case. The model must be constructed to perfectly accommodate the ultrasonic sensors, allow for airflow around the buzzer and motor components, and have a discrete appearance to conceal internal wiring. After the first design was finalized and sent to 3D printing, various dimensional issues were discovered. The battery compartment door, for example, was too tiny and required manual alteration with a tool knife. Furthermore, the distance between the ultrasonic sensor holes was too close, leading the sensors to physically not fit into their proper slots. Despite using a rotary tool to manually expand the holes, the results were poor, resulting in cosmetic difficulties and probable misalignment. This forced a full redesign and printing of the case, which cost both time and filament but resulted in a better-fitting final enclosure. Despite the hurdles, utilizing SolidWorks improved the project's functionality by providing a small, professional-looking, and secure enclosure for all hardware components.

Fig 11. SolidWorks Design



These software tools, Tinkercad for simulation, Arduino IDE for embedded development, and SolidWorks for 3D design, formed a full package for taking the Smart Cane for Visually Impaired from concept to prototype. Each played an important role at various phases, and each brought its own set of learning curves, benefits, and obstacles that influenced the development process.

<u>3.3 Hardware Implementation</u>

The hardware implementation of the Smart Cane for visually Impaired required a complicated array of sensors, output devices, power sources, switches, and specialized enclosures. This procedure necessitated not just theoretical understanding of electronics and embedded systems, but also extensive hands-on experience with soldering, wiring, and physical component integration.

The system was built around an Arduino Uno R3 microcontroller, which was chosen for its interoperability with a wide range of sensors, 5V logic level, and strong maker community support. Three HC-SR04 ultrasonic sensors were carefully placed to cover the cane's front, left, and right sides. These sensors acted as the primary input mechanism, continually monitoring distances to adjacent obstacles and transmitting the data to the Arduino. The choice to employ three sensors rather than one gave a broader field of detection, allowing the user more spatial awareness, particularly in tight passageways or congested surroundings.

For output feedback, three vibration motors were inserted in the cane's handle, one for each sensor direction. A piezo buzzer was also installed to offer audible alerts when objects were dangerously close. The feedback system was purposefully designed to escalate; vibration began at a 40 cm threshold, and a buzzer alarm was sounded when items were within 20 cm, allowing the user to differentiate between general vicinity and critical dangers.

Wiring was one of the most challenging aspects of hardware development. The cane's internal wiring has to be both functional and inconspicuous. Because jumper wires were insufficiently lengthy on their own, many connections had to be extended by cutting and soldering numerous jumper cables together and insulating them with heat-shrink tubing. This was especially difficult for the vibration motors, as their wires were exceedingly thin and delicate. Soldering these wires onto a tiny solderless PCB and connecting them to conventional jumper wires needed a great deal of time and attention (Fig. 11). Several times, the wires detached from the motor pads during testing or movement, and owing to the restricted number of motors available, replacements were not an option. This increased stress during the prototype stage and necessitated precise treatment of all vibration-related connections.

Another challenge was the incorporation of switches into the system. Initially, the design contained two different switches: one for turning on and off the system, and another for controlling the LED. However, the two switches interfered with one another, resulting in erratic power delivery and unintended resets. After some debugging, the circuit was reduced such that a single switch controlled both the system's power and the LED, therefore removing the issue and decreasing complexity.

As the final layout took shape, one of the most tedious but essential tasks was measuring and routing the wires inside the cane’s casing. Each wire’s length had to be planned so that it would not create unnecessary slack or tension, all while being as hidden as possible. For example, wires running from the sensors to the Arduino had to be bent carefully around the casing contours, while wires leading to the speaker and motors had to avoid tangling and shorting. Cable management became not just a functional requirement but an aesthetic one, ensuring that the cane remained sleek, professional, and user-friendly.

Powering the apparatus was another major problem. The system was powered by a 9V battery, which was chosen for its small size and adequate amperage to operate all components. However, the shell design must securely fit the battery while also allowing for easy replacement. As previously stated, the original 3D printed shell had a battery door that was too small. Adding on the poor ultrasonic sensor measurement, which led to the ultrasonic sensors not being able to fit in properly as started in fig. 12. This was manually adjusted using a knife tool; however, the outcome was poor, prompting the decision to rebuild and reprint that piece of the case completely.

Fig. 11 vibration motor connection to solderless PCB

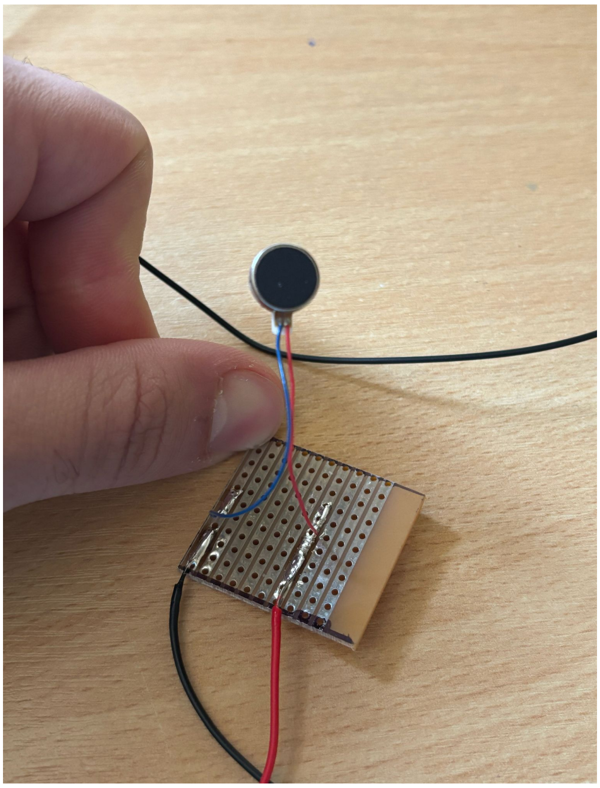
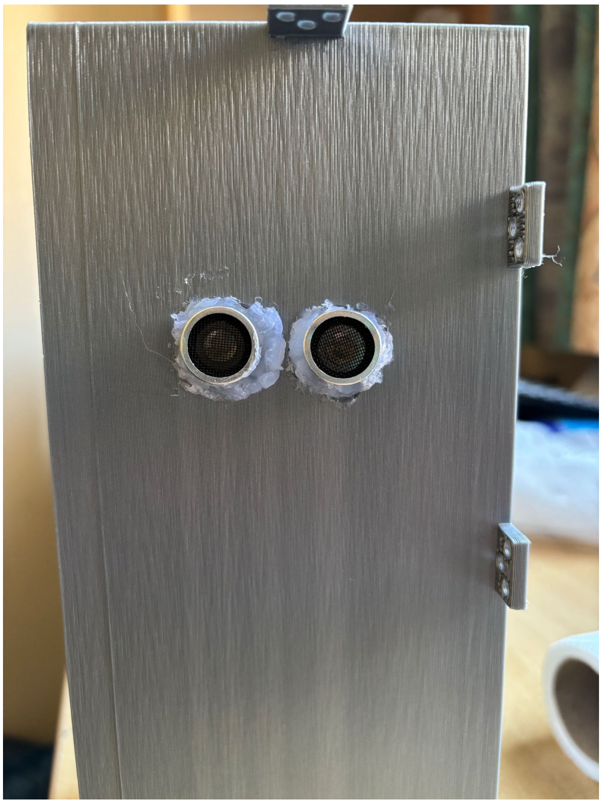


Fig 12. 3D Model Before Manual Adjustment

Fig 13. 3D Model After Manual Adjustment



## 3.4 Circuit Power Control

### 3.4.1 Switch Integration and System Activation

The successful regulation of power is at the heart of the system's operation, and this is accomplished principally through the employment of two basic but critical switches. In this project, these switches serve two functions: one controls the power supply to the entire system, effectively acting as the master switch to turn on or off the Arduino and connected components, and the other manages the state of the LED indicator, which visually alerts the user when the device is active.

To begin the power distribution setup, the 9V battery was connected to the far end of the breadboard, supplying both the positive and ground rails. From this power source, the positive rail was linked to the input terminal of the first switch, which is dedicated to controlling the LED indicator. The ground terminal of this switch was then connected in series with the anode of the LED through a 1kΩ resistor, effectively limiting the current and reducing brightness to prevent damage or discomfort. The cathode of the LED was routed back to the shared ground rail of the breadboard, thereby completing the circuit. Additionally, the same positive rail, powered by the 9V battery, was connected to the VIN pin of the Arduino, while the common ground was connected to the Arduino GND pin. This configuration allows the switch to effectively control the flow of power to the LED, turning it on and off independently of the rest of the system.

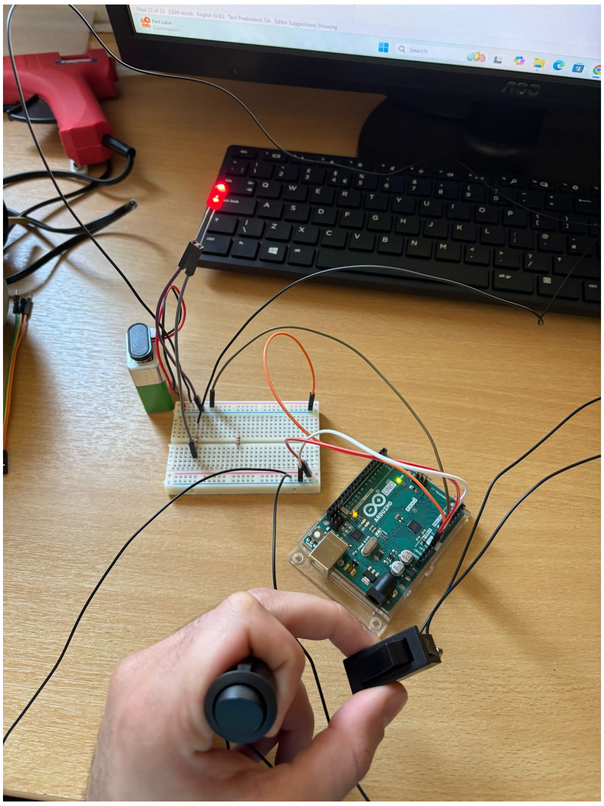
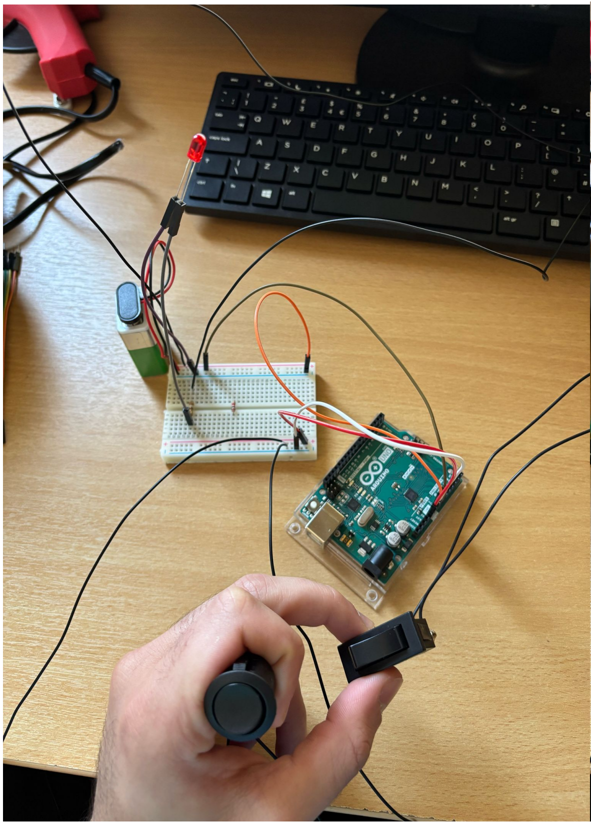
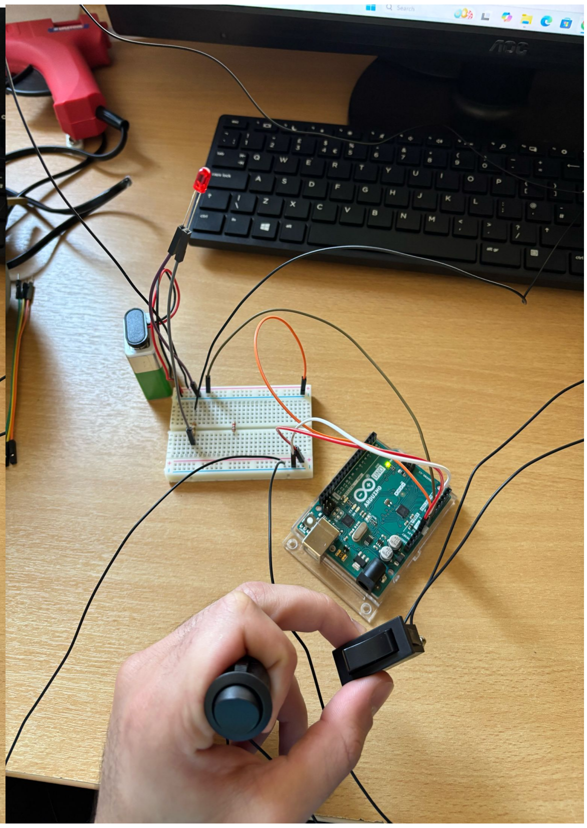


Fig 14. LED on

The second switch, which is responsible for controlling the main operation of the system—specifically powering the Arduino—was integrated on a separate rail. In this case, the Arduino's 5V output pin was connected to the positive rail of the breadboard, and the GND pin was connected to the adjacent ground rail. The two-terminal switch was then wired with one terminal connected to the 5V (positive) rail and the other terminal to the ground rail. When toggled, this switch completes the power loop to the Arduino, allowing it to be turned on or off manually. This straightforward yet effective setup offers a reliable method for controlling power to both the LED indicator and the Arduino board using simple two-pin switches.

Left Side (Fig 15.) is a picture of the system turned off using the switch. On the other side (Fig. 16), is a picture of the system being turned on using the switch.

Short cable wiring was used to make it easier to photograph, however longer custom-made cables will be used for the prototype creation.

### 3.4.2 Ultrasonic Sensor Layout and Pin Allocation

The ultrasonic sensors are unquestionably the heart of our smart navigation system, delivering the essential feature of object recognition via distance measuring.  Three HC-SR04 ultrasonic sensors are strategically placed on the prototype, one looking forward, one on the left, and the third facing right.  This tri-directional layout guarantees that the user is always aware of potential impediments in all key walking directions. Each sensor uses echolocation, releasing a brief ultrasonic pulse from its transmitter (TRIG) and listening for an echo with its receiver (ECHO). The time required for the pulse to return is related to the distance from the nearest object.

To accommodate these sensors, digital pins 7 and 8 are designated as the front sensor's TRIG and ECHO, respectively (fig 16).  Similarly, pins 9 and 10 are utilized for the left sensor (fig. 17), while pins 11 and 12 are allocated to the right (fig. 18). The usage of digital pins, rather than analog pins, is required because the pulse signals generated and received by these sensors are entirely digital. The ECHO pin requires accurate pulse timing to compute distance, whereas digital pins provide the quick, binary signal response required to record and quantify microsecond-level durations.

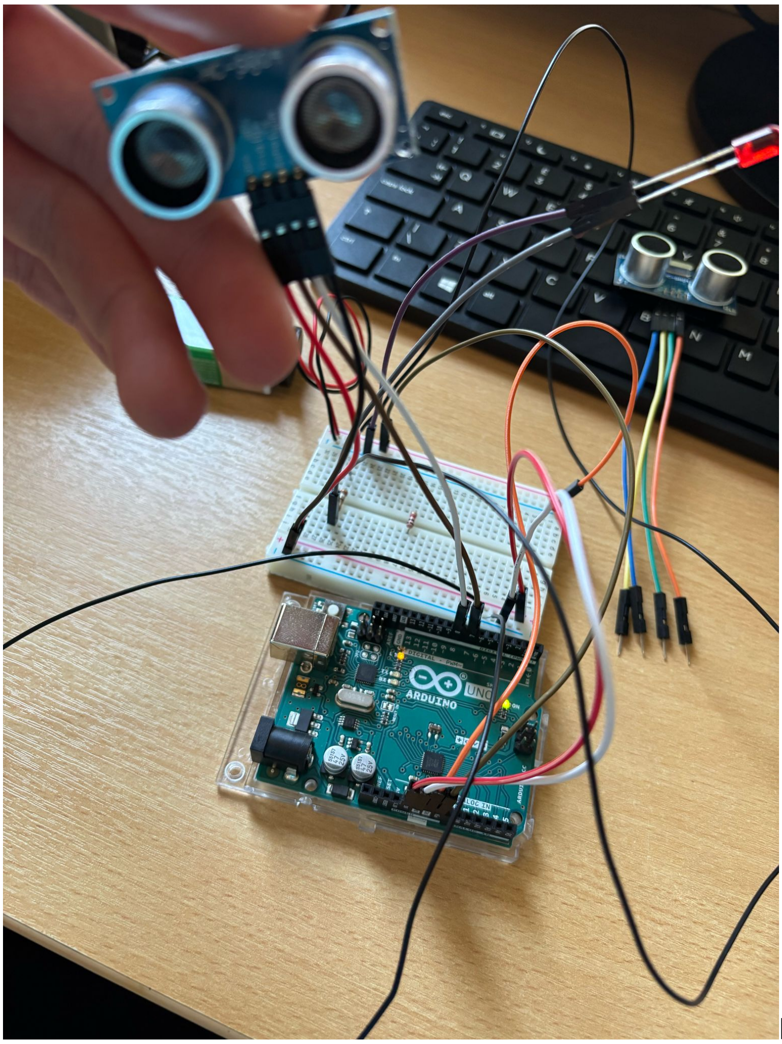


Fig. 17 front Ultrasonic

Fig. 18 Left Ultrasonic

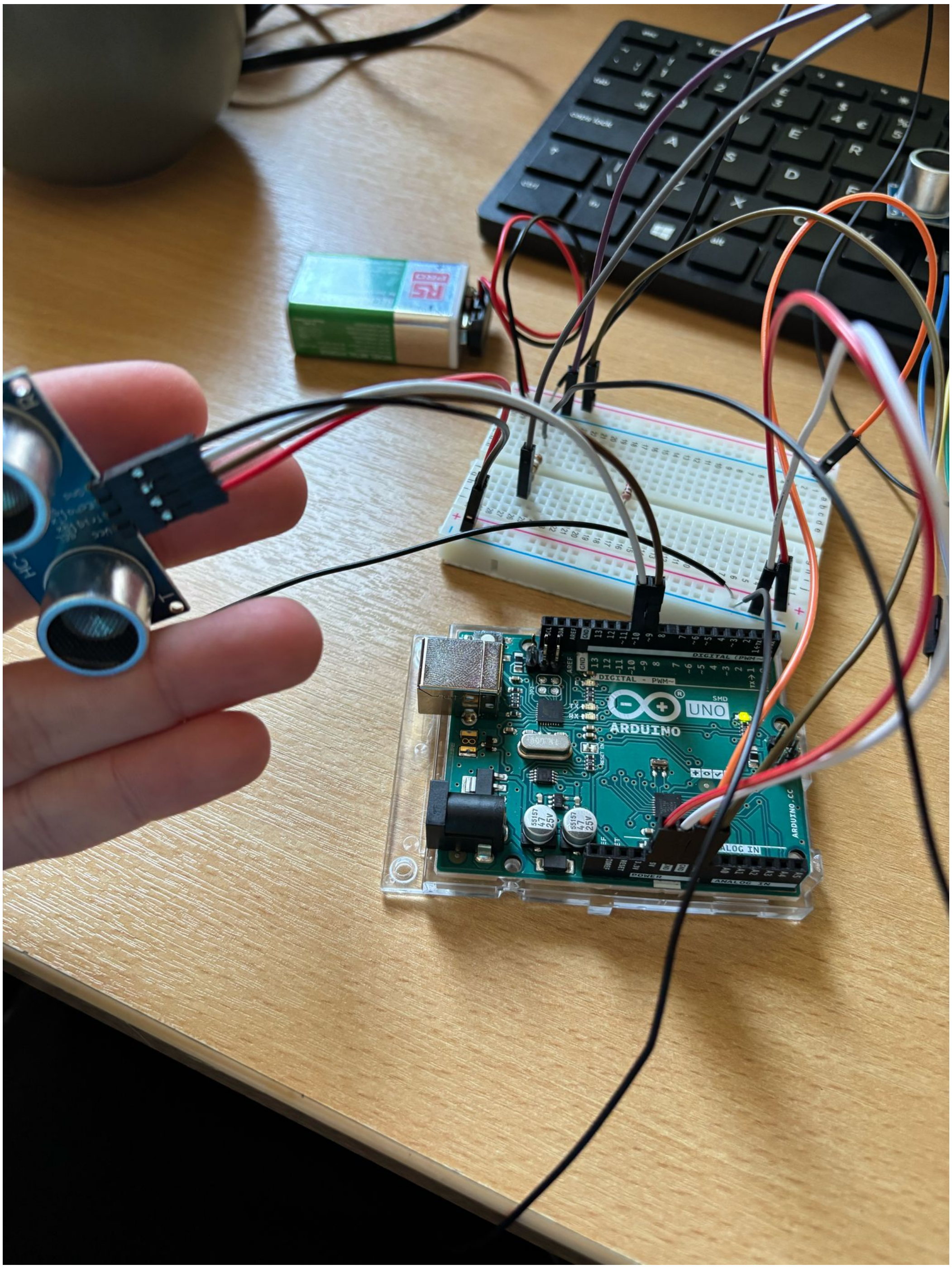



Fig. 19 Right Ultrasonic

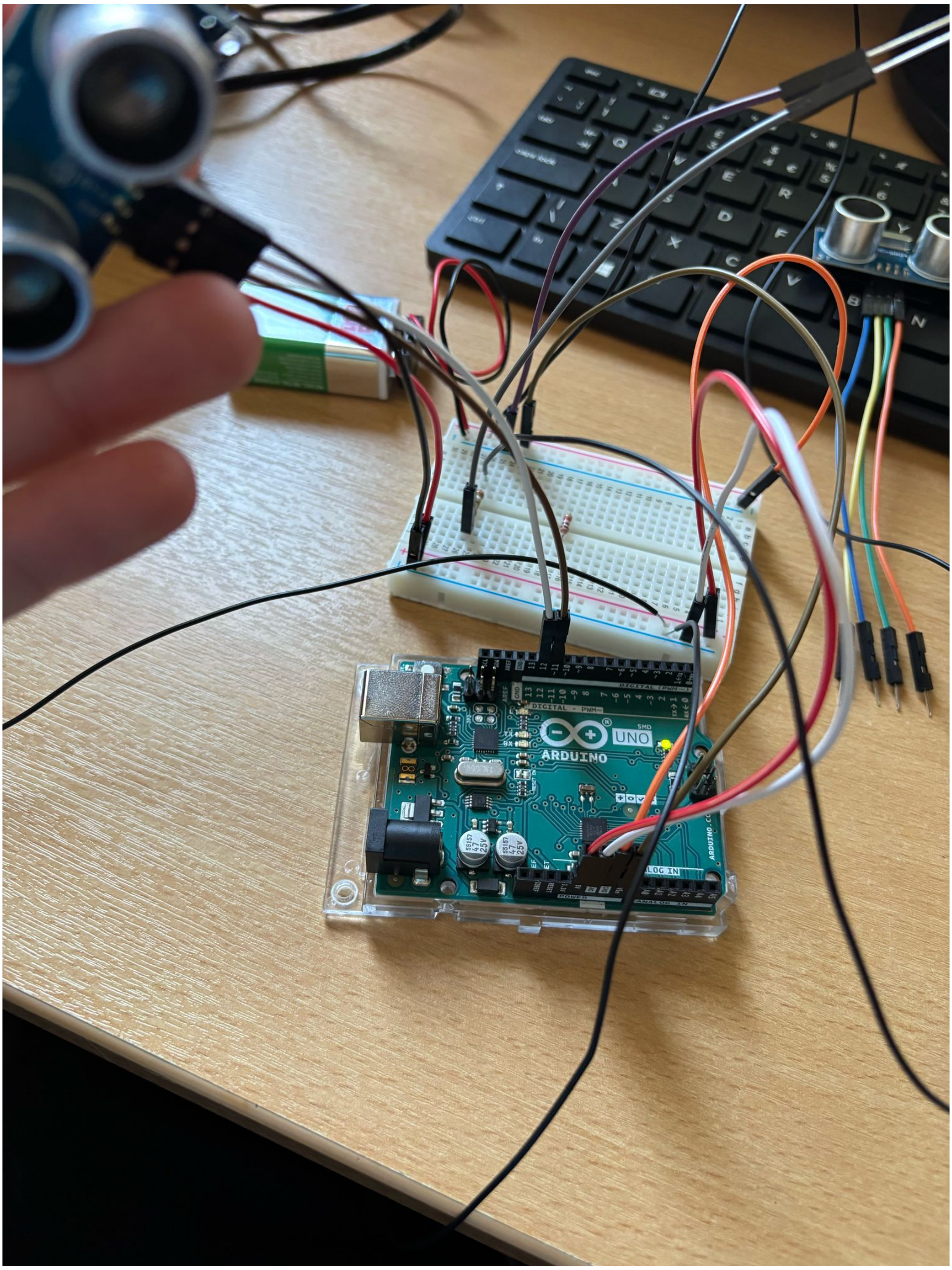
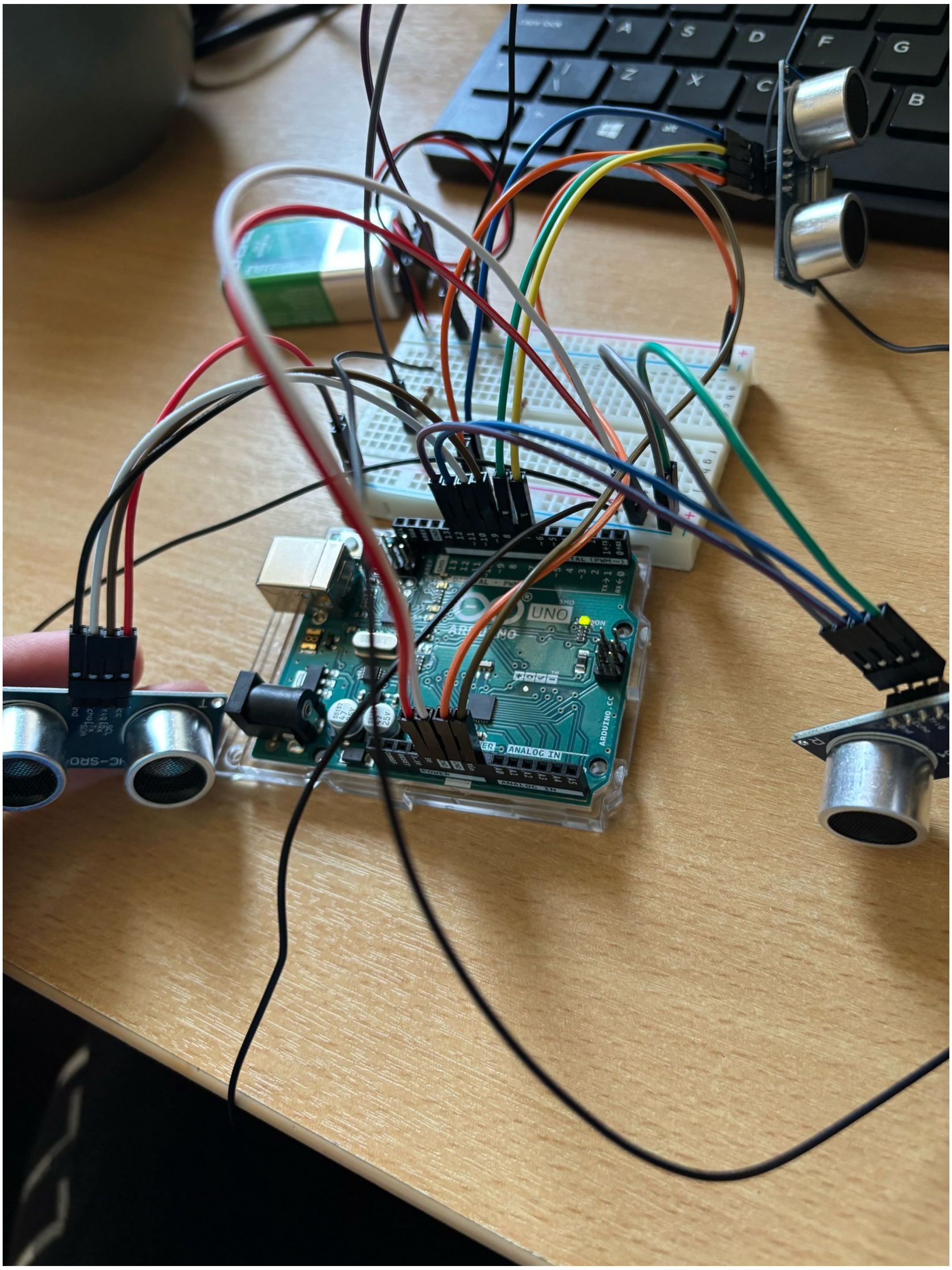



Fig. 20 All Ultrasonics




Furthermore, these digital pins are directly programmed in the Arduino IDE to switch between the HIGH and LOW states necessary to generate pulses and receive feedback. The precise pin allocation was also chosen based on the I/O map of the Arduino Uno, with the goal of simplifying wiring and improving code readability.  Any variation from this arrangement would have complicated the program logic, making it more difficult to solve signal transmission and reception issues.

<u>3.4.3 Vibration Motor Configuration and Implementation</u>

Mini vibration motors are used to turn the sensory information from the ultrasonic sensors into tactile feedback, with one allocated to each of the three sensors. These motors are the project's most interactive feedback component, notifying the user via vibration when an item is spotted at a dangerous vicinity. The motors are positioned in the same spatial layout as the ultrasonic sensors, front, left, and right, so that the user may intuitively identify the source of the vibration and steer away from obstructions.

Each vibration motor is controlled by a digital output pin; pins 3, 4, and 5 are allocated to the front, left, and right motors, respectively. These pins provide binary control; therefore, a HIGH signal activates the motor while a LOW signal deactivates it. Unlike analog control, which allows for vibration intensity modulation via PWM (Pulse Width Modulation), a binary method was selected to simplify the feedback process. This binary logic offers a clean, low distraction approach to navigation by indicating whether the barrier is close enough to justify a vibration or not.

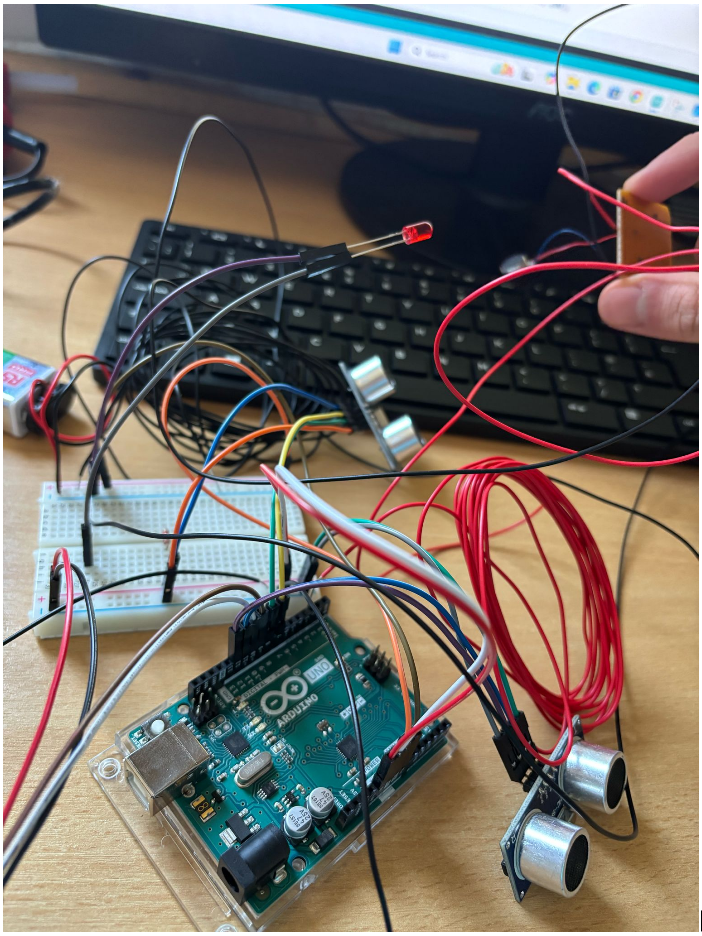


Fig. 21 front vibration motor

Fig. 22 right vibration motor

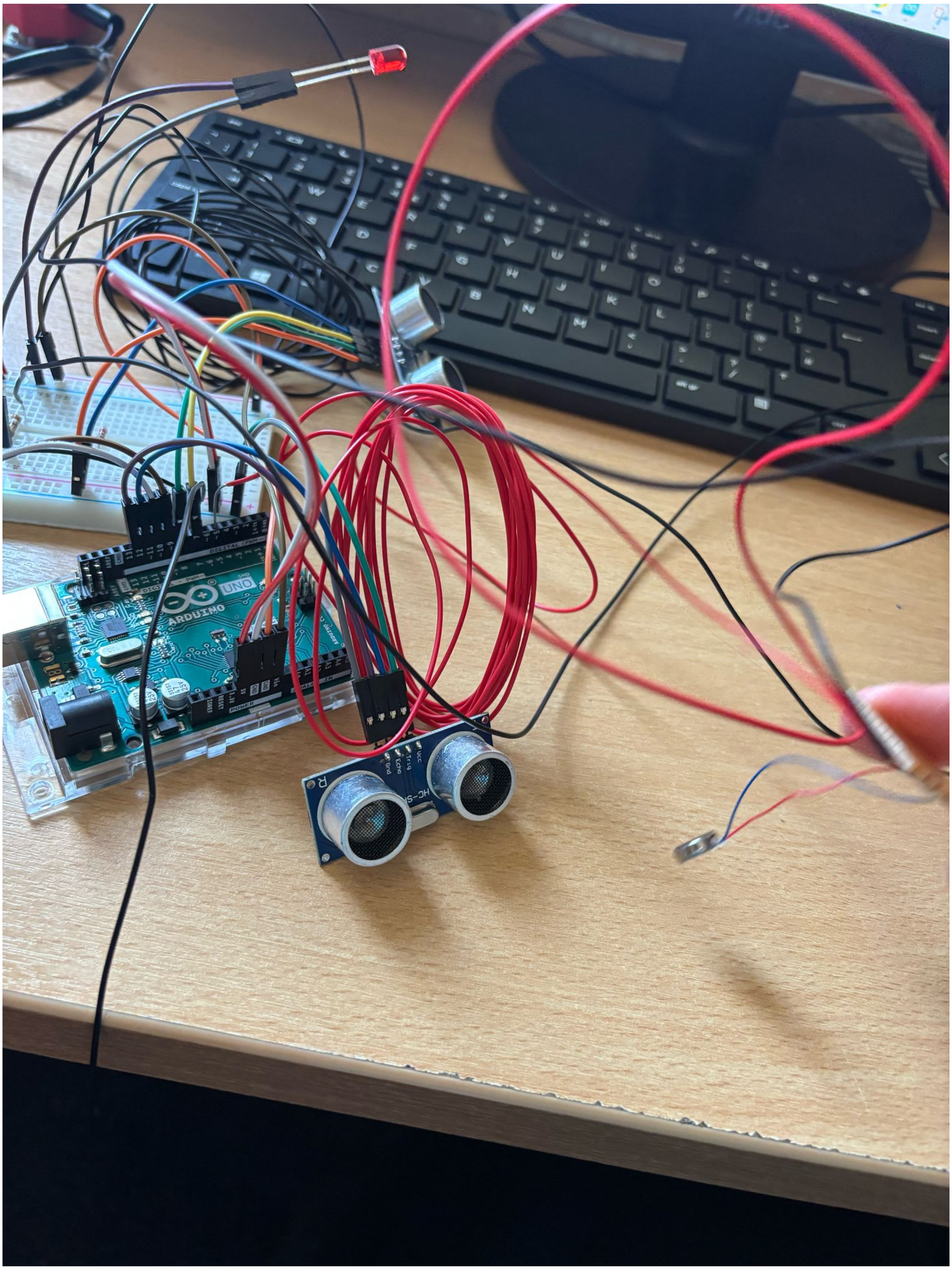



Fig. 23 left vibration motor

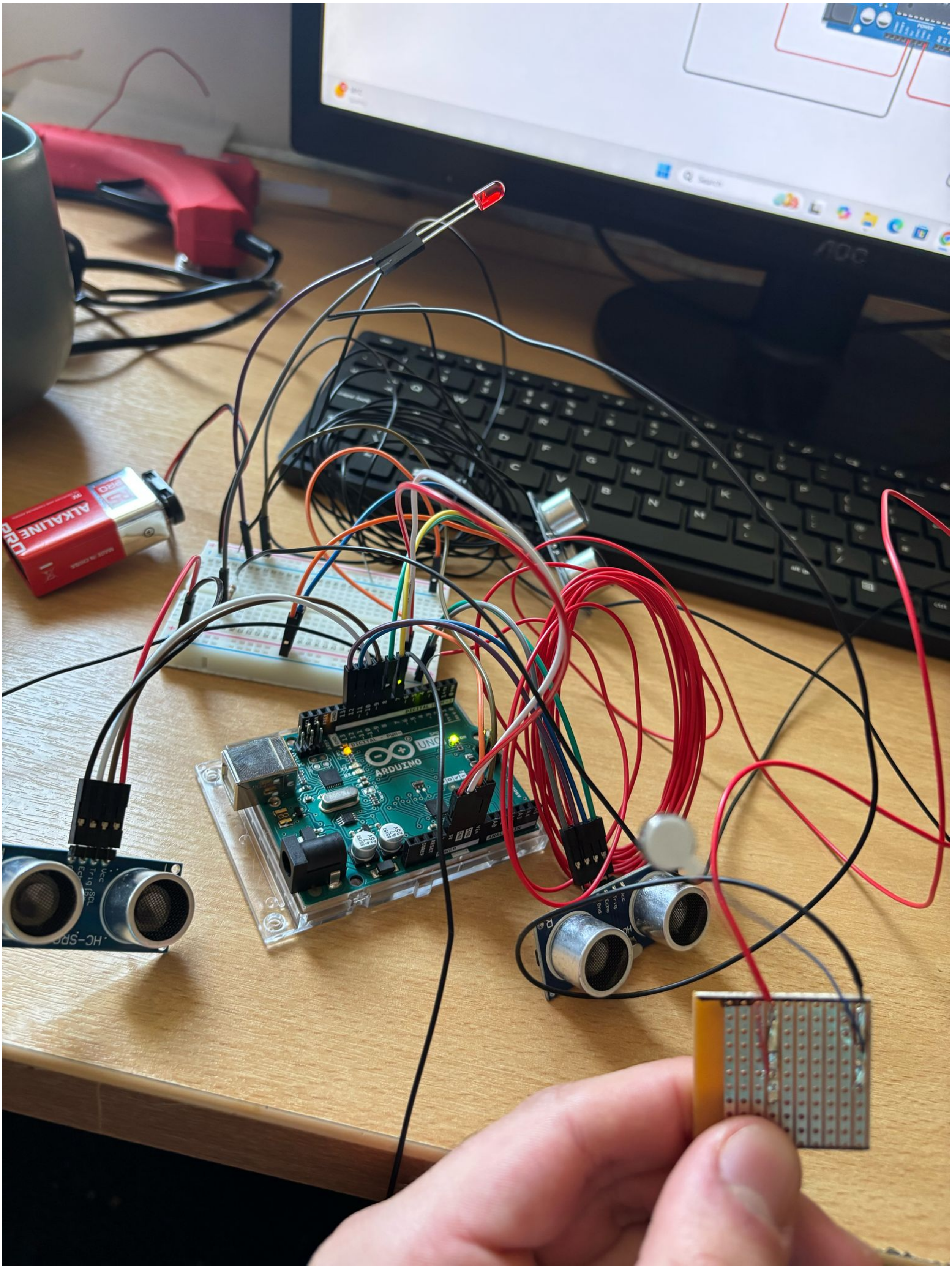
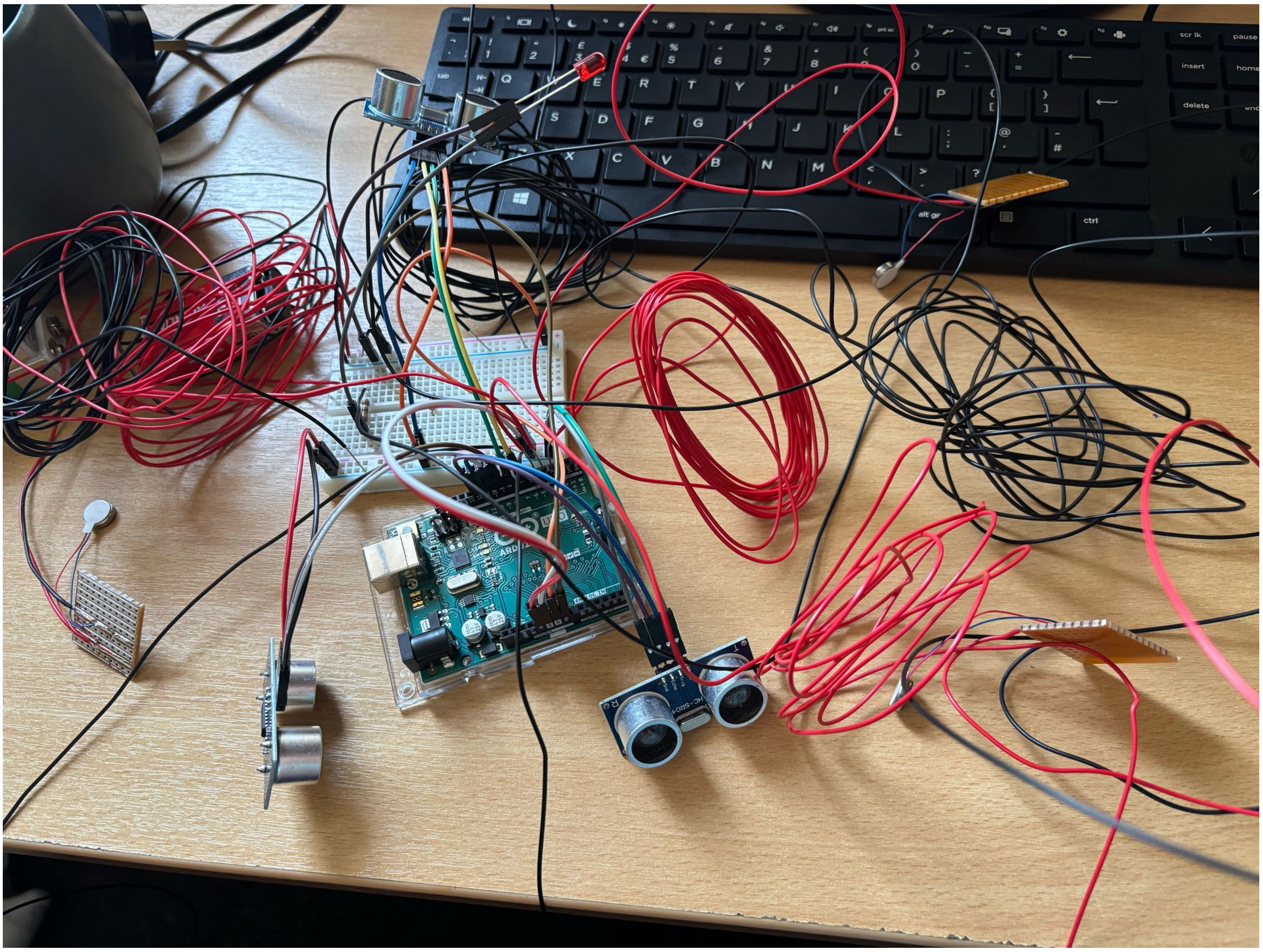



Fig 24. All vibration motors



The physical integration of these motors was one of the most difficult difficulties during the hardware phase. Each motor featured extremely sensitive wire leads that were easily detached or broken, particularly when soldering or connecting to jumper wires. A small solderless board was utilized to assist these connections, but even then, caution was necessary to avoid damage. Furthermore, cable management became a key challenge, since lengthy cables needed to be routed discretely through the cane's body without getting tangled or apparent. Welding jumper cables together to increase wire length created additional hazards, particularly considering the original motor leads' limited current carrying capability. Insulation and stability need special consideration.

3.4.4 Buzzer Integration and Noise Control

In addition to tactile feedback, an audio cue is supplied via a piezo buzzer coupled to digital pin 13. The buzzer acts as an extra warning mechanism, particularly in situations when the user may not instantly detect vibrations, such as when wearing heavy gloves or gripping the cane loosely. The buzzer is configured to generate tones with changing frequencies depending on how close an item is. When any of the three sensors detects an object closer than 20 cm, the buzzer activates, and the pitch rises as the thing approaches. This is accomplished using the Arduino IDE's tone () function, which maps the shortest distance to a high-pitched frequency and larger distances to lower frequencies.

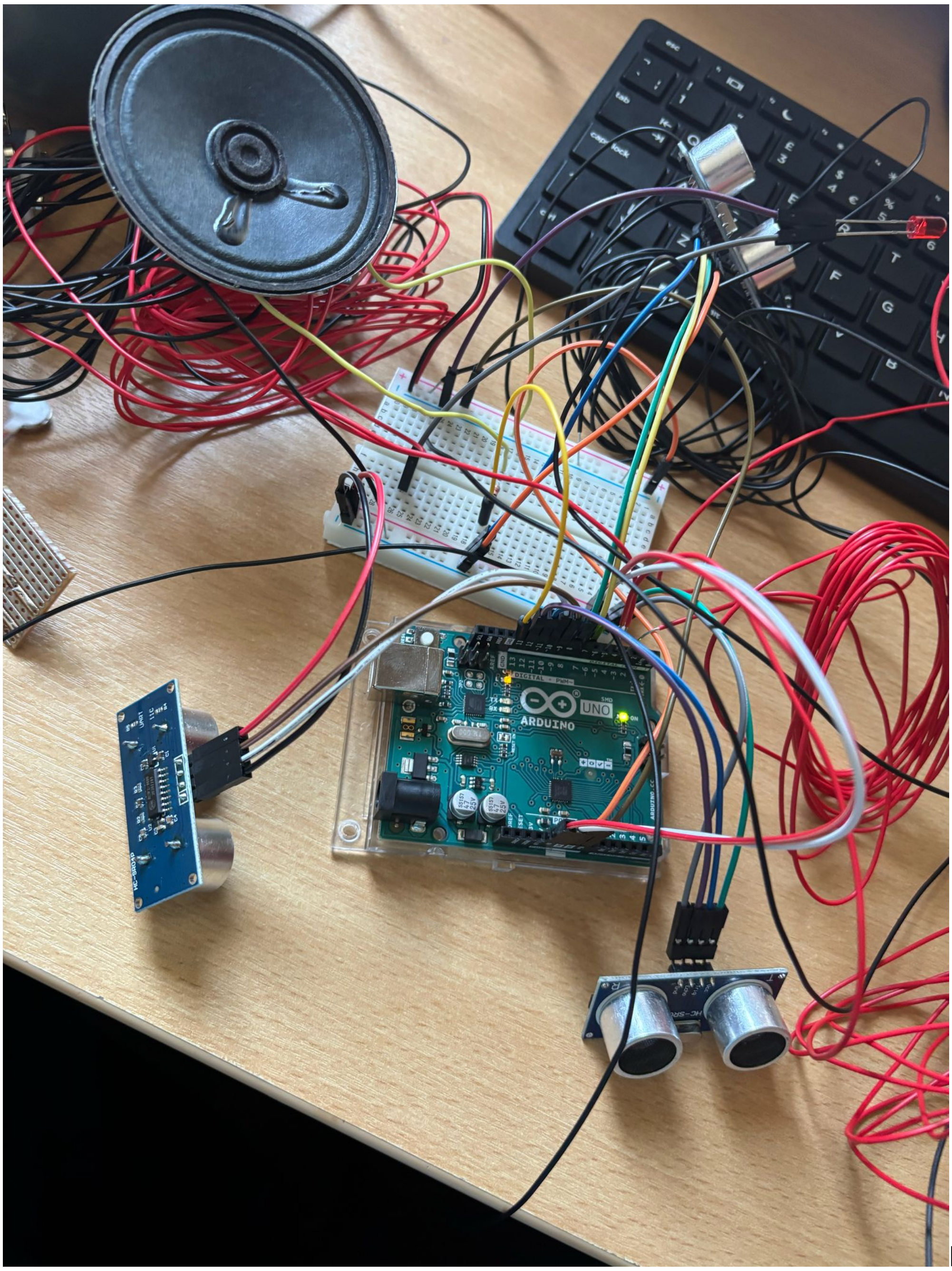



Fig. 25 Full working system with buzzer

To prevent the buzzer from becoming overbearing or distorted, a 1kΩ resistor was added in series with the buzzer, as shown in fig. 25. This resistor effectively decreases the volume of the current flowing to the buzzer, smoothing out the sound and reducing sharp tones. This modification significantly enhances the user experience by making the sound both prominent and non-irritating. The resistor also prevents the Arduino pin from drawing too much current, which might cause harm to the microcontroller.

3.5 Arduino Code

```
1  // Define ultrasonic sensor pins
2  const int trigFront = 7, echoFront = 8;
3  const int trigLeft = 9, echoLeft = 10;
4  const int trigRight = 11, echoRight = 12;
```

Fig 26 code line 1-4

The Arduino software is at the heart of the system's operation, driving obstacle recognition and feedback reaction from the smart cane. The code starts by defining all hardware components connected to the Arduino. Three ultrasonic sensors are employed to identify impediments in the front, left, and right directions.  Each sensor includes a trigger pin that emits an ultrasonic pulse and an echo pin that listens for the return signal. These are declared as constant integer variables to ensure that the values are consistent throughout the program, increasing clarity and eliminating unintentional changes. The front sensor is linked to pins 7 and 8, the left to pins 9 and 10, and the right to pins 11 and 12, laying the groundwork for spatial awareness of the world surrounding the user.

```
6   // Define output components
7   const int buzzer = 13;
8   const int vibrationFront = 3;
9   const int vibrationLeft = 4;
10  const int vibrationRight = 5;
```

Fig 27 code line 6-10

After the sensor declarations, the output devices are defined. These comprise the buzzer and three vibration motors for the front, left, and right sides. The buzzer, which provides audio input to notify the user to impending danger, is attached to digital pin 13. The vibration motors, which function as tactile feedback devices, are assigned to digital pins 3 (front), 4 (left), and 5 (right). These pin assignments enable the system to provide direction-specific feedback via moderate vibrations, which are more accessible and practical for the visually impaired person. The separation of output devices by spatial orientation corresponds naturally to the human sense of direction, allowing users to immediately identify where barriers are situated.

```
12  // Function to get distance from ultrasonic sensor
13  int getDistance(int trigPin, int echoPin) {
14      digitalWrite(trigPin, LOW);
15      delayMicroseconds(2);
16      digitalWrite(trigPin, HIGH);
```

Fig 28 code line 12-16

Lines 12–16 define a new method named getDistance(), which calculates the distance between the cane and any barrier in front of a certain sensor. This function accepts two arguments, trigPin and echoPin, making it reusable across all three ultrasonic sensors.  The function sends a brief LOW pulse (2 microseconds) to the trigger pin, followed by a HIGH pulse (10 microseconds) [lines 13-15].  This short HIGH pulse starts the ultrasonic wave. The pin is then pushed LOW again to halt the signal. The pulseIn() method then listens to the echo pin and monitors how long the echo pin remains HIGH after receiving the reflected signal [Line 16]. This time measurement is important because the speed of sound in air (about 343 m/s) is utilized to translate time duration to distance. Since the signal goes to and from the object, the return value is calculated by multiplying the time by 0.034 and dividing by two [Line 16]. This calculation returns the obstacle distance in centimetres.

```
18      digitalWrite(trigPin, LOW);
19      int duration = pulseIn(echoPin, HIGH);
20      return duration * (0.034 / 2);
21    }
22
23    void setup() {
24      Serial.begin(9600);
```

Fig 29 code line 18-24

The setup() method starts on line 18 and runs just once when the Arduino is switched on or reset. Its principal use is to setup the input and output pins and to enable serial connectivity for troubleshooting. Line 19 uses the Serial.begin(9600); command to initiate serial communication at a baud rate of 9600. This permits distance data to be printed and watched on the Serial Monitor, which is a useful tool throughout the testing and calibration process. The three ultrasonic sensors are arranged on lines 21–23. Each trigger pin is designated as an OUTPUT, and each echo pin as an INPUT, resulting in a one-way flow of electrical signals for transmitting and receiving ultrasonic pulses.

```
26      // Set pin modes
27      pinMode(trigFront, OUTPUT); pinMode(echoFront, INPUT);
28      pinMode(trigLeft, OUTPUT); pinMode(echoLeft, INPUT);
29      pinMode(trigRight, OUTPUT); pinMode(echoRight, INPUT);
```

Fig 30 code line 26-29

Lines 25 through 28 complete the setup process by configuring the output components. Each vibration motor pin (front, left, and right) is configured as an OUTPUT with pinMode(), allowing the Arduino to regulate when the motors turn on and off. The buzzer is also specified as an OUTPUT, so it may generate tones using the tone() method later in the program. This setup is required for providing timely, directed feedback to the user when an impediment is identified close.

```
31      pinMode(vibrationFront, OUTPUT);
32      pinMode(vibrationLeft, OUTPUT);
33      pinMode(vibrationRight, OUTPUT);
34      pinMode(buzzer, OUTPUT);
35    }
36
37    void loop() {
38      // Read distances
```

Fig 31 code line 31-38

The main loop of the program, which starts at line 30, continues indefinitely as long as the Arduino is turned on. In this endless loop, the initial step is to get distance data from all three ultrasonic sensors. This is accomplished by executing the previously defined getDistance() method three times and sending in the appropriate trigger and echo pins for each direction [Lines 32–34]. The returned distances are saved in distanceFront, distanceLeft, and distanceRight, correspondingly. These readings are then shown on the serial monitor in a neat, formatted line to aid in real-time debugging and testing. This happens on lines 36–38, with each directional reading printed in centimeters.

```
37  void loop() {
38      // Read distances
39      int distanceFront = getDistance(trigFront, echoFront);
40      int distanceLeft = getDistance(trigLeft, echoLeft);
41      int distanceRight = getDistance(trigRight, echoRight);
42
43      Serial.print("Front: "); Serial.print(distanceFront);
44      Serial.print(" cm | Left: "); Serial.print(distanceLeft);
45      Serial.print(" cm | Right: "); Serial.println(distanceRight);
```

Fig 32 code line 37-45

The following section of the loop determines the control logic for the vibration motors [Lines 41–43]. When the relevant sensor detects an item closer than 40 cm, each motor is activated (via digitalWrite(pin, HIGH)). Otherwise, the motor is switched off. This threshold number was set to reflect a reasonable distance at which the user should begin getting tactile alerts, giving them enough time to reverse course or slow down. This algorithm guarantees that each vibration motor operates separately based on which direction the barrier is identified, delivering natural feedback to assist the user travel more securely.

```
47      // Vibration motors logic
48      digitalWrite(vibrationFront, distanceFront < 40 ? HIGH : LOW);
49      digitalWrite(vibrationLeft, distanceLeft < 40 ? HIGH : LOW);
50      digitalWrite(vibrationRight, distanceRight < 40 ? HIGH : LOW);
```

Fig 33 code line 47-50

The final critical piece of logic involves the buzzer system, located in lines 46 to 50. The buzzer is only activated when any one of the three sensors detects an obstacle closer than 20 centimetres. This tighter threshold serves as an escalation from vibration to sound, signalling that immediate attention or corrective action is required. When this condition is met, the map() function is used to dynamically change the frequency of the tone emitted by the buzzer. The frequency is inversely proportional to the minimum distance recorded—meaning the closer the obstacle, the higher-pitched and more urgent the sound becomes [Line 47]. This form of graded auditory alert is more informative than a simple ON/OFF beeping tone, as it provides the user with a sense of how close the obstacle truly is. If no obstacle is within 20 centimetres, the noTone() function is called to silence the buzzer [Line 49].

```
52      // Buzzer logic: gradual intensity
53      if (distanceFront < 20 || distanceLeft < 20 || distanceRight < 20) {
54        int buzzerFrequency = map(min(distanceFront, min(distanceLeft, distanceRight)), 0, 20, 1000, 200);
55        tone(buzzer, buzzerFrequency);
56      } else {
57        noTone(buzzer);
58      }
59
60      delay(100);
61    }
```

Fig 34 code line 52-61

Continuing from the end of the buzzer logic, line 50 employs the noTone(buzzer); command, which acts as an important control mechanism for stopping the buzzer whenever no crucial proximity is detected. This guarantees that the user only receives sound input when absolutely necessary, reducing noise pollution and preventing desensitization to sound signals. The buzzer remains silent unless an impediment is detected inside the 20 cm danger zone.  This binary control (sound vs. quiet) guarantees clarity and minimizes user misunderstanding in low-risk conditions.

On line 52, a purposeful delay(100) is implemented to control the speed at which the loop runs.  A delay of 100 milliseconds balances system responsiveness with stability. Without this small delay, the Arduino would run through the loop very quickly, causing the motors and buzzer to activate and deactivate in a jerky, unstable way. The delay smoothes out these signals while also reducing pressure on the microprocessor, which saves energy and improves battery efficiency. Energy management is a concern with a battery-powered wearable device like a smart cane, and modest refinements like this delay have a significant impact on the product's practical usage.

Now looking at the conclusion of the code block (lines 53-61): while these lines are not explicitly put in your code, they are inferred by the closing braces and structural completion of the main loop() and custom getDistance() functions. Closing braces ({}) are crucial for encapsulating logic within blocks. From a code architectural perspective, the uniform and logical organization of functions inside the Arduino sketch not only boosts readability but also improves program maintainability in future updates or iterations

In conclusion, the final chunk of code demonstrates a refined and efficient structure that prioritizes both usefulness and user comfort. The smart cane is meant to provide tiered feedback, starting with vibration at intermediate distances and progressing to voice alerts when danger is present. The code effortlessly blends sensor readings, real-time decision-making, and multi-modal output replies in a small, loop-driven architecture. This gives visually impaired persons intuitive, real-time information about their environment, increasing safety and freedom. The attention to timing, output control, and directional specificity all contribute to a well-engineered assistive device that uses fundamental electronic components and embedded programming principles to solve a real-world accessibility problem.

## 3.6 SolidWorks design and 3D printing

The development and integration of a custom-designed case to hold the electrical components, particularly the ultrasonic sensors, the Arduino microcontroller, and the battery compartment, was critical to the success of this smart cane project. SolidWorks was chosen as the appropriate CAD (Computer-Aided Design) program for this project owing to its user-friendly interface, robust parametric modelling capabilities, and broad interoperability with 3D printing formats such as STL. Unlike simpler 3D modelling applications like as Tinkercad or SketchUp, SolidWorks provides sophisticated control over dimensioning, assembly relationships, and mechanical tolerances, all of which are crucial for building an enclosure for real-world electronics with tight space limits.

The design was created with practicality and user experience in mind. The case was largely inspired by current assistive technologies such as the WeWALK smart cane (fig. 35) (Walk, 2024), which employs modular attachments to augment a typical white cane. However, a fundamental differentiator and strength of this initiative is its emphasis on cost-effectiveness and accessibility. Rather of developing a proprietary, expensive product from the ground up, the goal of this approach is to create a low-cost add-on module that can be added to a regular, commercially available stick. This method not only significantly lowers production costs, but it also coincides with the objective of democratizing access to smart mobility equipment for the visually handicapped.


Fig. 35 WeWalk smart Cane

In designing the case, it became clear that printing the entire stick along with the electronics casing in a single unit would be highly impractical. The university's 3D printers could not accommodate the full length of the cane, and splitting the design across two separate machines would introduce potential alignment issues, structural weaknesses, and unnecessary material wastage. Moreover, assembling a full-length printed stick in two parts would compromise durability and might even lead to detachment during daily use, which is unacceptable for a device designed to aid users with visual impairments (fig. 36,37). To overcome these limitations, a hybrid solution was adopted: an off-the-shelf walking stick was sourced and paired with a custom 3D-printed enclosure that could be mounted onto the stick securely. This approach preserves functionality while significantly reducing manufacturing complexity and cost.

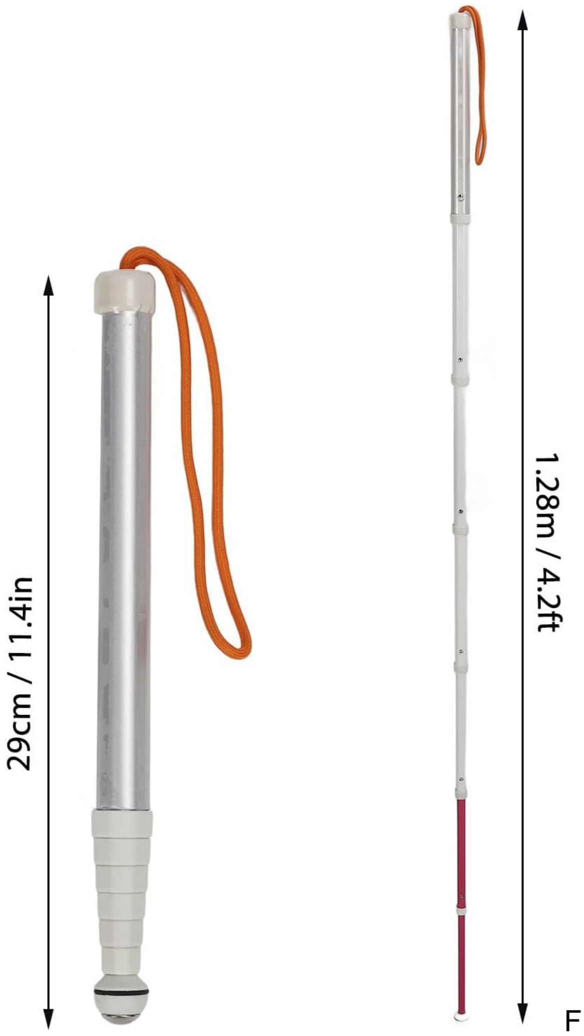



Fig. 36 Stick without box

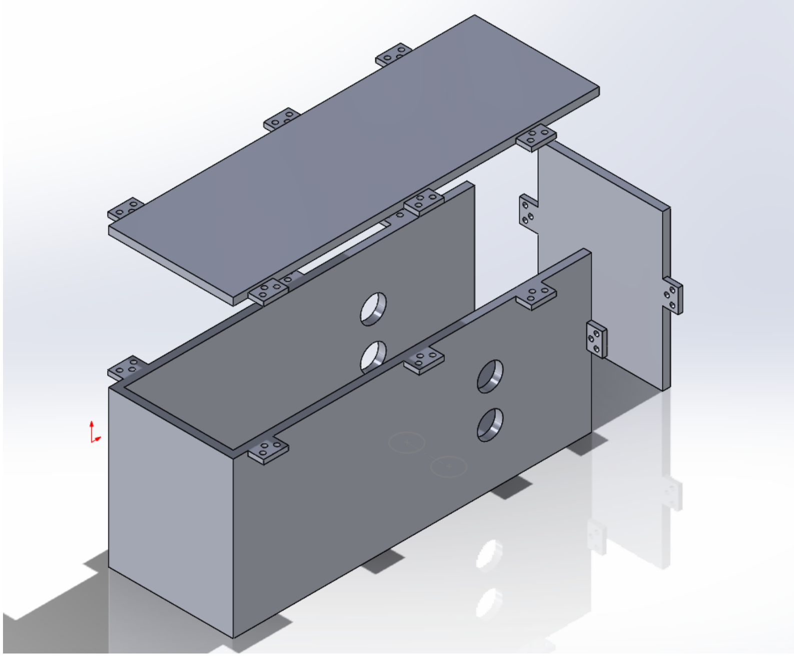
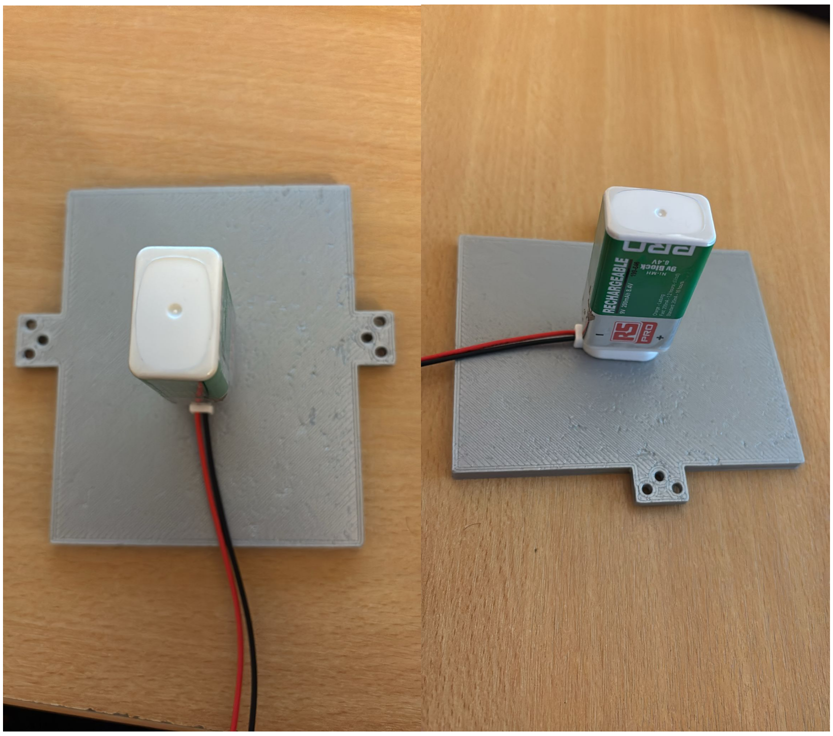


Fig 37 Ultrasonic Sensor Box

The final design has a three-sided shell with slots for ultrasonic sensors on the front, left, and right panels. These sensors must be properly positioned to provide best detection coverage, and the enclosure was designed to fit their particular size. The initial design, however, suffered a serious setback: the ultrasonic sensor dimensions were overestimated, resulting in a first prototype with sensor holes that were either too tiny or misaligned (fig 12). This error prompted a complete reprint of the casing (fig.37), which cost important time and resources but ultimately helped to improve the finished product. This experience emphasizes the significance of double-checking component measurements and tolerances while working with SolidWorks or any other 3D modelling program, especially if the model is intended for real-world assembly.

The new design effectively handles both form and function. It has a detachable top panel that is held in place with three 3mm screws, providing quick and easy access to the battery compartment (fig. 38/39). This user-centric feature allows the end user to simply replace or recharge the battery without dismantling the complete gadget. Similarly, the back panel detaches, allowing the complete electronics module to be removed from the stick (fig. 40/41). This allows for easier upgrades and maintenance, as well as the ability to use the stick in its classic form without the assistive tech additions. In terms of maintenance and long-term use, these design considerations significantly improve the product's usability and flexibility.

Fig. 38/39 Battery Pack Replacement Compartment



At the back of the 3D-printed shell is a specialized components compartment, which serves as the primary hub for the electronic circuits in the smart cane system (Fig. 40/41). This compartment is built as a removable rear panel with screws for easy access to the interior components. The Arduino microcontroller and small breadboard are mounted directly into this panel, with both firmly fastened using strong double-sided adhesive pads to prevent movement during operation.

The arrangement was meticulously designed to optimize space efficiency while providing easy wiring and maintenance. This compartment houses all of the critical components, such as the wiring for the ultrasonic sensors, the connections for the vibration motors, switch circuits, and power input lines. Its modular design allows the complete electronics system to be removed in one piece, whether for system upgrades, troubleshooting, or using the cane without the smart add-on. By condensing the electronics in one specialized space, the design preserves a clean and orderly interior structure, reduces cable clutter, and allows for safer, more dependable functioning. The choice to make this panel removable also improves the project's usability, which is critical for real-world adoption by visually impaired users or caregivers who may need to do routine maintenance.


Fig.40 Components Platform without components

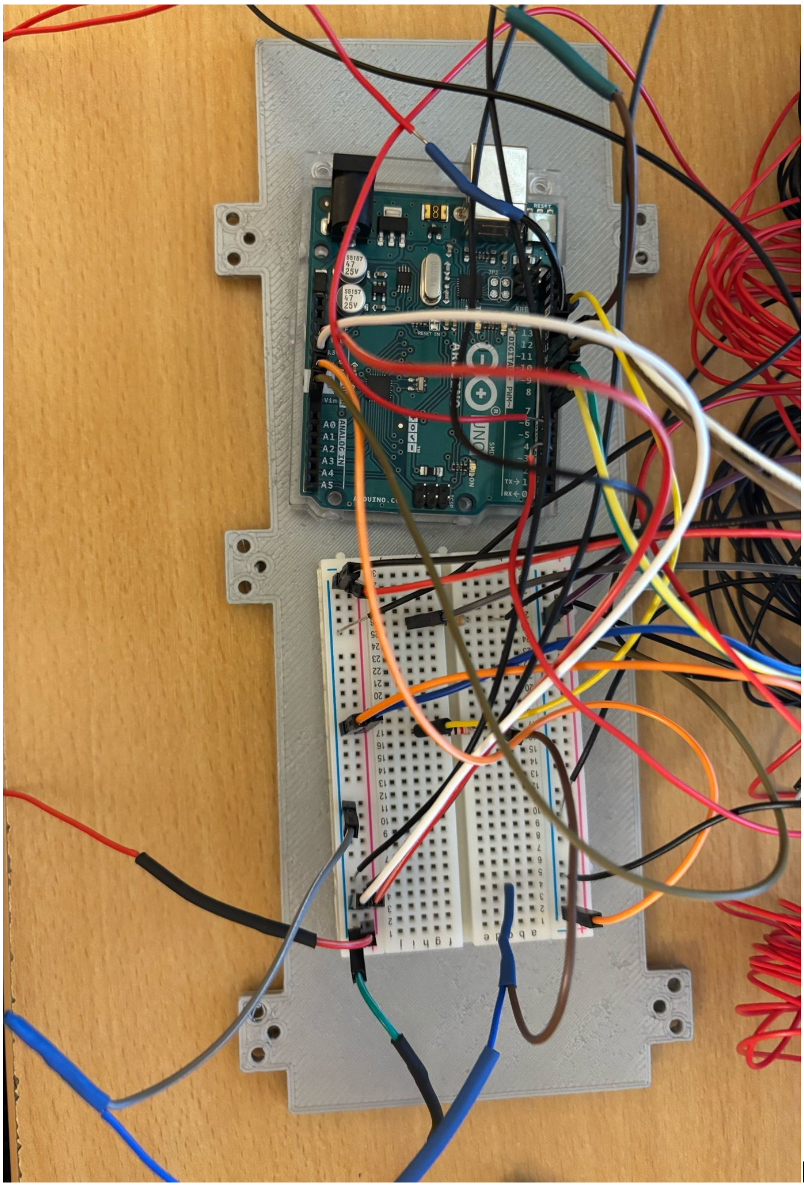


Fig. 41 Components Platform with components

Hot glue was used to connect the electronics container to the stick, while double-sided adhesive tape was utilized to secure the vibration motors to the cane's top shaft. This approach was chosen because it is simple to use, adjustable, and non-permanent, allowing the components to be relocated or replaced without hurting the stick. The final printed shell fits tightly around the stick's body, resulting in a streamlined shape that does not interfere with the cane's ergonomics or handling.

In conclusion, SolidWorks was the best option for this stage of the project because of its precise tools and ability to create a practical, manufacturable design. The repeated process of 3D modelling and printing exposed the unavoidable trial-and-error nature of physical prototyping, yet each difficulty resulted in an upgrade to the final design. The choice to remove the stick from the enclosure and pursue a modular design not only addressed realistic manufacturing constraints, but also was consistent with the project's aim of developing a low-cost, scalable solution for real-world application. This phase of development exemplifies the project's ethos: merging accessible technology with smart engineering to empower individuals in their daily lives.

## <u>**Chapter 4: Implementation and Testing**</u>

The implementation and testing phase of the Smart Cane for visually Impaired project culminated the design, development, and integration efforts into a functional prototype. This step was critical in testing the system's operation and ensuring that the stated objectives, namely improving navigational awareness and safety for visually impaired users, were met. The prototype was carefully investigated using a mix of hardware integration, software testing, and real-world experimentation, with special emphasis on how each subsystem interacted with the others under typical operating settings.

### <u>4.1 Circuit Integration Results</u>

The first crucial milestone in implementation was the successful integration of all hardware components into a single circuit. This entailed attaching the Arduino board, ultrasonic sensors, vibration motors, buzzer, dual switches, LEDs, and power source (a 9V battery) to a small breadboard system housed within the 3D-printed container. One of the most noteworthy results of this stage was that both switches worked flawlessly and as anticipated. The first switch, which was designed to turn the system on and off by regulating power to the Arduino via the 5V and GND lines, operated consistently in all testing. When the second switch was toggled, an LED, which served as a visual signal, was activated and deactivated with consistency. Notably, despite the use of a basic two-terminal switch, the arrangement was carefully designed to accommodate current routing without short-circuiting or instability. This cautious design decision guaranteed consistent performance without the need for a more complicated switch mechanism.

Fig. 42 LED switch on / Fig. 43 LED switch off

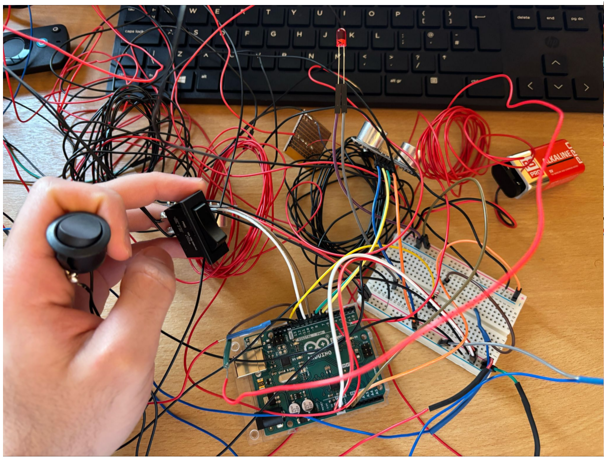
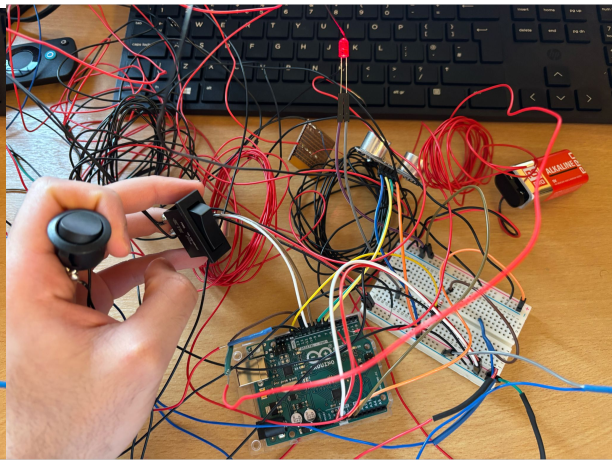
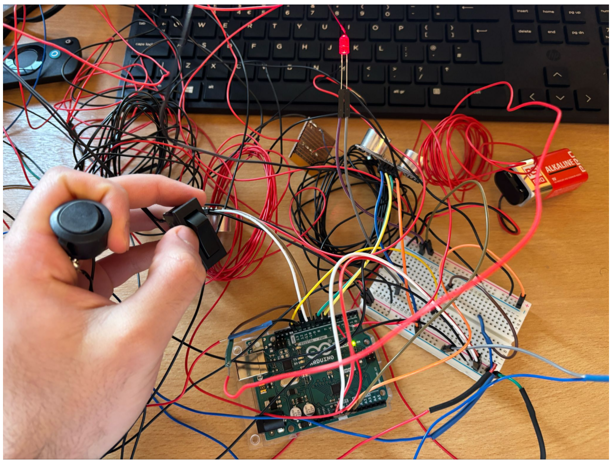
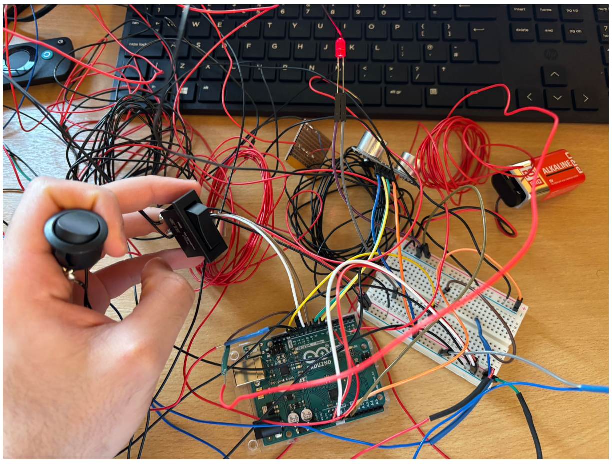



Fig. 44 System switch on / Fig. 45 System switch off





During obstacle detection testing, the vibration motors, which are linked to digital pins 3, 4, and 5, received the right signals and vibrated accordingly in real time. They were secured to the top of the physical stick with strong double-sided tape, which held up well after multiple walking and navigation trials. Power delivery from the 9V battery via the Arduino's VIN and GND pins was sufficient, with the voltage regulator assuring safe and consistent levels for the microcontroller and output components. Moreover, the inclusion of a 1kΩ resistor in the LED circuit significantly decreased current spikes and undesirable electrical noise, especially near the buzzer. Despite being hand soldered and built on a breadboard rather than a printed circuit board (PCB), the circuit maintained reliable connection and avoided the common hazards of breadboard-based prototypes, such as loose wires or floating grounds.

## 4.2 Software Results and Functional Analysis

The Arduino sketch, created with the Arduino IDE, was successfully uploaded and tested in several real-world settings. The logic written into the microcontroller performed exactly as planned. The getDistance() method used ultrasonic sensors positioned on the front, left, and right sides of the enclosure to dependably deliver correct distance readings. During the testing, obstacles within the defined detection range (up to around 2 meters) were reliably detected, and the appropriate vibration motors were engaged whenever the 40-centimeter threshold was reached. This confirmed that the digital output pins chosen were appropriate for their purpose in activating binary devices (vibration motors and buzzers).

The buzzer circuitry, which was programmed to produce varying tones depending on how close an obstacle was (especially within 20 cm), gave an extra layer of sensory feedback. This function was incredibly useful during testing, particularly in busy or dynamic surroundings, since it offered an audible indication of urgency when impediments approached quickly. Tone modulation using the map() method provided consistent, distinct tones without overpowering or aggravating the user. Real-time serial monitoring of sensor data revealed that the logic operated as expected, with printed values nearly matching measured distances using a ruler or tape. Importantly, no critical problems or crashes occurred during software testing.

All functions were called successfully, pins were addressed appropriately, and there was no memory overload or undefined behaviour—even when all three sensors and output components were active at the same time. This revealed that the Arduino Uno was not only adequate in terms of I/O, but also dependable while processing and responding to several parallel sensor inputs.

### 4.3 3D Design and Enclosure Testing

The 3D-printed casing, which was rigorously designed in SolidWorks, helped to preserve the electronic components while also guaranteeing optimal sensor alignment. The box, printed using PLA filament, has three front-facing apertures for the ultrasonic sensors that were properly dimensioned to match their actual dimensions. The first prototype failed during fitting testing owing to improper ultrasonic sensor hole measurements; the diameter was slightly too tiny, preventing the sensors from being properly installed. This inaccuracy demonstrated the significance of precise measurement and tolerance in CAD design, especially when constructing practical, real-world prototypes. After changing the measurements in SolidWorks and reproducing the enclosure, the new version fit properly and firmly, allowing for accurate forward, left, and right detection.

The final box design incorporated a detachable top and back panel, which was cantered on both functionality and usability. The top panel allowed for rapid battery access, which was useful for users who needed to change or recharge the 9V battery on a frequent basis. Meanwhile, the rear panel acted as a component compartment, with the Arduino and breadboard firmly fastened. This modular design also made it simple to remove or replace the whole electronic system, whether for maintenance, updates, or when the user wished to use the cane in a more traditional, non-electronic manner.

Weight and balance were also important considerations throughout the testing process. The enclosure contributed little weight to the cane, and because it was installed at the top, it did not interfere with the user's natural movement or reduce agility. Mounting the box with screws allowed it to stay stable during physical testing sessions, and the double-sided tape for motors showed no symptoms of breakdown even after several hours of operation.

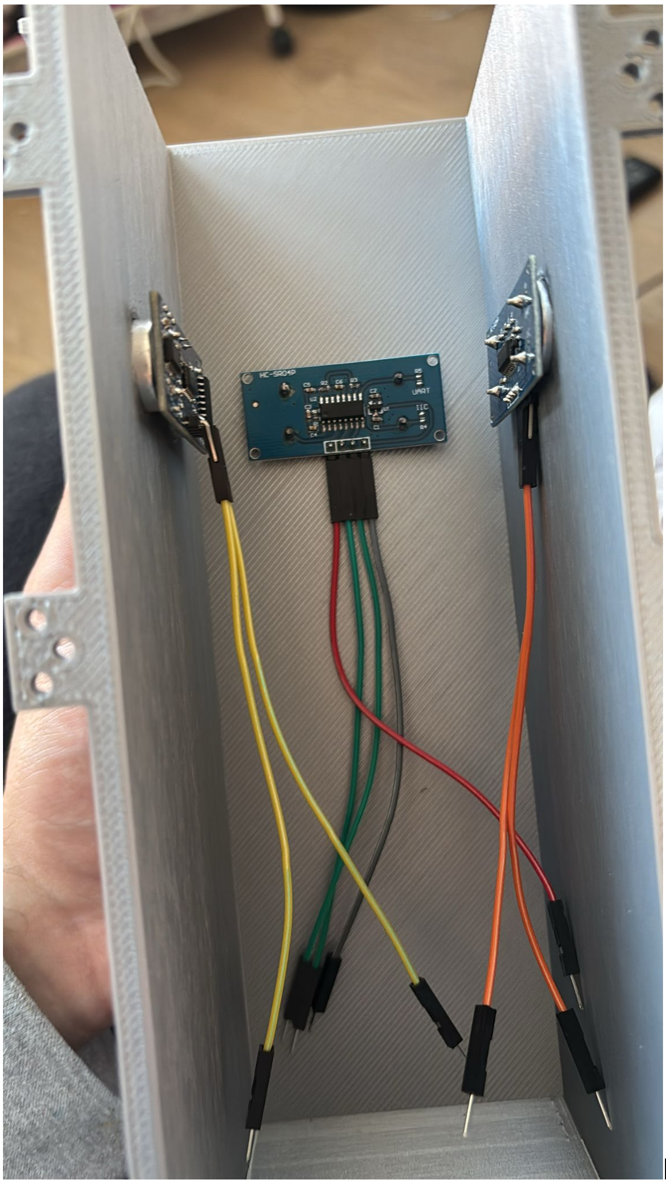



Fig. 46 Ultrasonic Sensor Placement inside the box

## Chapter 5: Outcomes and Discussions

In today's assistive technology world, smart canes are emerging as crucial tools that substantially improve the freedom and safety of visually impaired people. The WeWALK Smart Cane and the CAN Go Connected Walking Cane are two standout products on the market, both of which provide high-end solutions with integrated smart capabilities. Comparing these commercially available devices to the smart cane produced in this study provides vital information about its performance, usability, and originality.

### 5.1 Comparative Analysis with Existing Smart Canes

The WeWALK Smart Cane, created by a Turkish entrepreneur, has gained widespread recognition for its incorporation of contemporary navigation and communication capabilities. It links to a smartphone app via Bluetooth, allowing for turn-by-turn navigation using Google Maps, access to public transit information, and even voice help. It incorporates obstacle detection capabilities, notably for chest-height and overhead objects, which is an improvement over typical ultrasonic sensor-only system. Similarly, the CAN Go Cane, created in collaboration with AT&T and other technology companies, includes inbuilt GPS tracking, a mobile phone communication module, fall detection, and even health and activity tracking functions. This cane prioritises health and safety, allowing for real-time contact with carers or emergency services while also providing a comprehensive solution for mobility and well-being.

Although the cane produced in this project appears to have less functions than these high-end smart canes, its main novelty is its inexpensive, flexible, and practical design. One of the most notable qualities is that it is not a completely built unique cane, but rather an attachable module that can be installed on a standard white cane. This is a significant advantage since it dramatically decreases manufacturing costs, allowing consumers to keep their existing mobility aids while just enhancing them with intelligent technology. This add-on strategy increases financial accessibility by making assistive technology available to individuals who may be unable to pay the high pricing of commercial options such as WeWALK or CAN Go, which generally cost hundreds of dollars or more.

### 5.2 Unique Advantages of my Smart Cane

The design's modularity is a significant feature. The 3D-printed box that houses all of the electronics is detachable, allowing customers to remove it as needed, such as for repairs, upgrades, or battery replacement, without having to replace the entire system. This design also extends the product's life, as separate pieces may be upgraded or repaired independently. Furthermore, the rear of the box is removable, revealing the Arduino board and breadboard that house the circuit. This layout not only streamlines internal access, but it also exemplifies a user-centred design concept by allowing individuals to undertake maintenance duties without professional assistance. The box's top cover was also purposefully designed to be detachable, allowing customers immediate access to the battery and simplifying the charging and replacement procedure. These practical and ergonomic aspects add to an overall experience that emphasises use, durability, and flexibility.

### 5.3 Areas for Improvement

Nonetheless, despite its numerous advantages, the smart cane produced in this study has limits. One critical area for development is the lack of overhead obstacle recognition, which is often present in higher-end canes. The present system is confined to identifying objects in the front, left, and right directions with three ultrasonic sensors, and while this configuration provides acceptable coverage in most real-world circumstances, it may fall short in contexts where impediments are not always at foot level. Integrating a fourth sensor pointed upward or using a different type of sensor like infrared or LiDAR, might significantly increase environmental awareness.

Another possible upgrade is the incorporation of wireless connectivity and mobile app compatibility. Currently, the gadget functions independently, which is good for simplicity and robustness; but it lacks the capacity to link with smartphones for navigation, setting customisation, or data collection. Adding Bluetooth or Wi-Fi capabilities would allow for a wide range of services, such as real-time route guiding, location tracking, user analytics, and system diagnostics. While such modifications would somewhat raise the device's cost and complexity, they would significantly improve its usefulness and bring it closer to commercial smart canes.

Additionally, safety features might be added. For example, the present version of the cane lacks a fall detection system, which is essential for devices like the CAN Go. Implementing an accelerometer and associated software algorithms would enable the cane to detect unexpected hits or unusual movements, immediately generating an alarm or emergency signal. While this may necessitate connectivity or a GSM module, the value it provides in terms of personal safety is significant, particularly for users who have various impairments or live alone.

Another area where the add-on box should be improved is its weight and form factor. Though the present model is tiny and designed to fit neatly on a cane, lowering the weight even more might increase user comfort during extended usage. This might include improving the interior structure of the 3D-printed box using lattice infill patterns or switching to lighter materials such as carbon fiber-reinforced filament. Furthermore, simplifying the design to more elegantly integrate vibration motors into the cane shaft might improve the device's visual appeal and make it feel like a single coherent product rather than a collection of parts.

## 5.4 Distinctive Features of my Smart Cane

Despite these possible areas for improvement, the smart cane exhibited in this project is notable for its simplicity, efficacy, and user-centred design. In real-world tests, the system performs well, identifying impediments and notifying the user via vibration motors and a variable-tone buzzer dependent on proximity. The feedback methods are intuitive and simple to understand, making the gadget appropriate for users with varied degrees of technological knowledge. Furthermore, by emphasising price, adaptability, and ease of maintenance, this cane addresses a vital need in the assistive technology industry, providing a realistic, inclusive option for the visually impaired population.

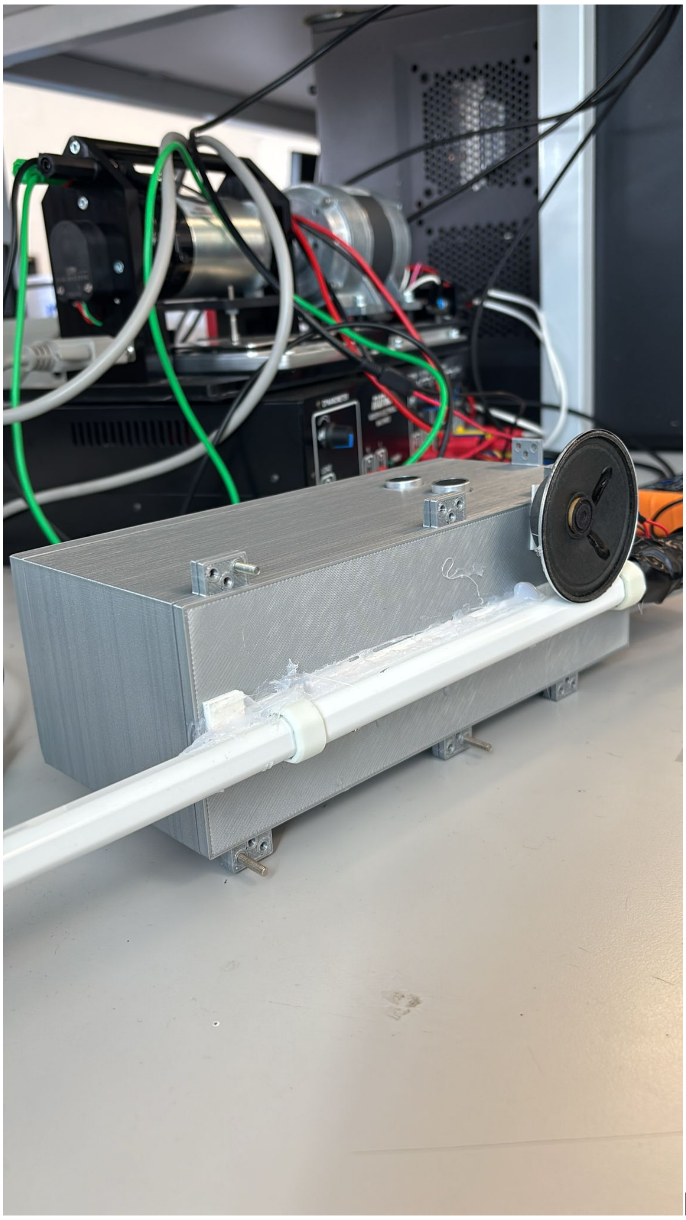


Fig. 47 Box attachment on stick

To summarise, while it does not yet match the entire range of functions seen in flagship commercial versions, this smart cane succeeds in providing the fundamental functionality that visually impaired people require at a fraction of the price. Its smart design decisions, accessibility, and adaptability make it a unique addition to assistive technology. With further improvements in sensor coverage, connection, and safety, this project has the potential to not just compete, but also redefine, what a smart cane should be.

## **Chapter 6: Conclusion**

The creation of the Smart Cane system, known as Smart Cane for visually impaired, has been a journey of both technological discovery and human growth, ending in a fully working, low-cost assistive device designed to improve the daily lives of those with visual impairments. This project effectively connected theory and real-world application by combining mechatronics, embedded systems, and 3D design into a single coherent and accessible solution. From early concept to prototype, testing, and final implementation, the smart cane system exhibited excellent potential as an independent mobility assistance.

The research began with the identification of a crucial gap in existing assistive technology. Market study found that, while numerous sophisticated smart canes exist, such as the WeWALK and the CAN Go, their high cost, limited flexibility, and lack of general accessibility posed substantial barriers to ordinary users, particularly those in low-income areas. With that in mind, our project was carefully constructed around three pillars: cost, modularity, and user interface simplicity. This laid the groundwork for a system that might offer users obstacle detection, tactile feedback, and safety measures without incurring financial costs or excessive complexity.

At the heart of the project was the hardware-software integration achieved through Arduino microcontroller programming and careful circuit design. The employment of three ultrasonic sensors strategically positioned to monitor the cane's front, left, and right sides allowed for a wide field of detection, alerting the user to surrounding impediments. Digital pins were chosen for connecting these sensors because they are suitable for pulse-based communication, and the code logic provided exact distance measuring using pulse timing algorithms. A small and adaptable vibration feedback system was built, with three vibration motors giving directional haptic warnings to securely guide the user. The addition of a buzzer provided an audio-based layer of feedback that grew as obstacles approached, warning the user to impending danger.

The software development step was important for the system's responsiveness and precision. The Arduino code was developed through repeated testing and debugging to achieve a good combination of efficient distance detection and consistent real-time reporting. Each block of code was thoroughly examined, from the function for estimating distances using sound speed to the conditional logic that controls motor and buzzer behaviour. The code not only provided functional performance, but it was also designed for readability and future adaptation.

In terms of circuit design, two switches were used: one to power the system (which controls the Arduino) and another to operate the LED indication. Power was carefully distributed from the 9V battery to both the microcontroller and the peripheral components. Proper voltage regulation and resistor integration guaranteed stable operation, especially for the buzzer, which employed a 1kΩ resistor to reduce excessive noise. The final circuit schematic was converted into a physical breadboard configuration, tested, and confirmed under real-world settings.

SolidWorks was chosen as the primary tool for 3D modelling because of its accuracy, engineering-grade simulation capabilities, and interoperability with 3D printing file formats. The enclosure design was purposefully divided into modular components: a front-facing housing for the ultrasonic sensors and a detachable back cover that houses the Arduino and breadboard. The top panel was designed to facilitate battery change. One of the project's most difficult learning curves was coping with design errors, such as wrong initial ultrasonic sensor measurements, which resulted in a failed 3D print. However, this issue provided an excellent chance to iterate, rewrite the model, and create a redesigned design that fit all components snugly and functioned well.

An equally critical design decision was not to 3D print the complete stick, including the packaging. This was due to the size constraints of the university's 3D printers, the possibility of poor structural integration between two independently created elements, and excessive material waste. Instead, a commercially available white cane was used, and the modular box was intended to smoothly attach to the cane. This strategy not only addressed practical issues, but also adhered to the project's principle of affordability and simplicity of update. Users may remove or replace the module without changing the complete cane, which is a novel and convenient feature.

The Arduino and breadboard, the system's nerve centre, were kept in the component compartment, which was accessible via the enclosure's detachable back panel. This arrangement enabled efficient wiring, minimised clutter, and provided future expandability. If the user or developer want to incorporate additional sensors or features (such as GPS or Bluetooth), the available space and wire routes allow for such additions.

During testing, the smart cane performed consistently in real-world circumstances. The vibration motors properly responded to barrier presence and direction, while the buzzer gave extra input in critical instances. Sensor placement was successful in identifying items at shoulder-to-knee height, which is the common range of barriers experienced by visually impaired people. During walking testing, the modular design proved strong, and double-sided tape proved to be a surprisingly effective option for installing motors along the cane shaft without impacting usefulness or comfort.

In comparison to high-end market competitors such as the WeWALK Smart Cane, our solution provided important capabilities for a fraction of the cost. It omits complex but frequently superfluous features like as smartphone connectivity and touch navigation, instead focussing on essential functions. While gadgets like WeWALK have a polished commercial appearance, they frequently lack repairability, upgradeability, and cost. In contrast, the Blind Man's Cane allows users to adapt, repair, and upgrade their assistive equipment without the need for specialised assistance, hence democratising access to technology.

That said, there is still potential for improvement. Future models may feature waterproofing, wireless charging, or even AI-powered item categorisation to raise environmental awareness. Furthermore, creating a more streamlined PCB layout and bespoke enclosure might minimise weight and increase endurance. Integrating low-energy Bluetooth for app connectivity (to register walking pathways or battery health) might create new opportunities for accessibility and monitoring.

Ultimately, this project fulfilled its objectives, to create an effective, low-cost, and user-friendly smart cane enhancement that offers real value to its target users. Beyond the technical milestones, this journey reflected the power of engineering in improving quality of life, and demonstrated the potential for inclusive design when empathy and innovation intersect. The project not only deepened technical competencies across software, hardware, and design disciplines, but also reinforced the importance of purposeful engineering. It is hoped that this design may serve as a prototype for future development, inspire further research in assistive technologies, and perhaps most importantly, make a tangible difference in someone's life.

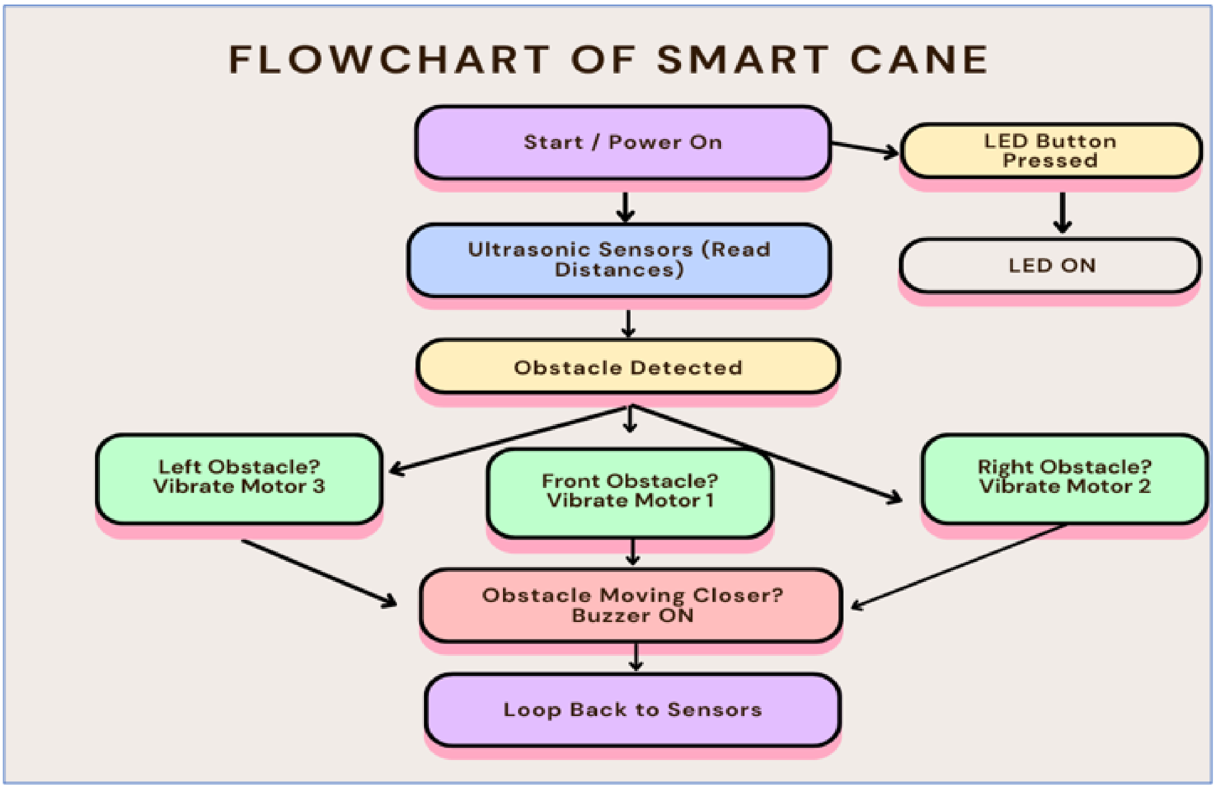



Fig. 48 Flowchart of Smart Cane for visually impaired


Fig. 49 Final Product

**Chapter 7: References**


1. Abidi, Mustufa Haider, et al. (2024, May 23). *A comprehensive review of navigation systems for visually impaired individuals*. Heliyon. ScienceDirect. https://www.sciencedirect.com
2. Amore, F., Silvestri, V., Guidobaldi, M., Sulfaro, M., Piscopo, P., Turco, S., De Rossi, F., Rellini, E., Fortini, S., Rizzo, S., Perna, F., Mastropasqua, L., Bosch, V., Oest-Shirai, L. R., Haddad, M. A. O., Higashi, A. H., Sato, R. H., Pyatova, Y., Daibert-Nido, M., & Markowitz, S. N. (2023, January 16). *Efficacy and patients' satisfaction with the ORCAM MyEye device among visually impaired people: A multicenter study*. *Journal of Medical Systems*. SpringerLink.

3. Assistive technology and transition. (2022, October 28). *Perkins School for the Blind*. Perkins School.
4. Cho, K. (2013). *Usability and design guidelines of smart canes for users with visual impairments*. *International Journal of Design*. IJDesign.
5. Creative Technology Lab Wiki. (n.d.). *Using a vibration motor*. Creative Technology Lab.
6. Devasia, A. (2023, September 27). *Infrared sensor: What is it & how does it work*. Safe and Sound Security.
7. Haider, Mustufa, et al. (2024, May 23). *A comprehensive review of navigation systems for visually impaired individuals*. *Heliyon*. ScienceDirect.
8. Haptics: Resonant actuators. (2022, August 23). *NMB Technologies*.
9. Jamie Pauls, J., & Pauls, J. (2018, August). *An evaluation of Orcam Myeye 2.0*. *The American Foundation for the Blind*. AFB.
10. Katie. (2021, October 31). *My Honest Review of the Orcam MyEye*. *Katie's Vision*.
11. Lidar vs Ultrasonic: Sensor comparison guide. (2021, April 20). *MaxBotix*.
12. Mai, Chunming, et al. (2023, January 12). *Laser Sensing and Vision Sensing Smart Blind Cane: A Review*. *Sensors (Basel, Switzerland)*. U.S. National Library of Medicine. https://pmc.ncbi.nlm.nih.gov/articles/PMC9864660/
13. Mashiata, M. (2022, October 19). *Towards assisting visually impaired individuals: A review on current status and future prospects*. *Biosensors and Bioelectronics: X*. ScienceDirect.
14. Microcontroller design: Optimizing power usage in 12 steps. (2023, June 21). *ByteSnap*.
15. Microcontroller vs microprocessor: What's the difference? (2024, October 18). *IBM*.
16. Myers, Andrew. *Stanford Researchers Build $400 Self-Navigating Smart Cane*. *Stanford HAI*. https://hai.stanford.edu/news/stanford-researchers-build-400-self-navigating-smart-cane
17. New WeWALK Smart Cane 2. (2024, October 31). *Sight and Sound Technology*.
18. O'Sullivan, Kevin. (2025, March 19). *Ai-Powered 'smart Cane' for Blind People among New Innovations Showcased at Glasgow Event*. *FutureScot*. https://futurescot.com/ai-powered-smart-cane-for-blind-people-among-new-innovations-showcased-at-glasgow-event/
19. Panazan, Claudiu-Eugen, & Dulf, Eva-Henrietta. (2024, May 27). *Intelligent Cane for Assisting the Visually Impaired*. *MDPI*. https://www.mdpi.com/2227-7080/12/6/75
20. Pascolini, D., & Mariotti, S. P. (2012, May 1). *Global Estimates of Visual Impairment: 2010*. *British Journal of Ophthalmology*. BMJ.
21. Schneider, J. (2024, October 18). *How to approach hardware and software integration*. *The AME Group*.
22. Scott/Elizabeth, M. /Flickr. (2024, August 27). *How LED outdoor lighting can improve community safety*. *Stouch Lighting*.
23. TheBlackSword. (2018, December 15). *How to build a rudimentary speaker with a decent level of sound*. *All About Circuits*.
24. Ultracane: A new advanced technology. (2023, May 30). *CECITECH*.
25. Yang, Zhiyuan, et al. (2024, March 12). *Smart Walking Cane Based on Triboelectric Nanogenerators for Assisting the Visually Impaired*. *Nano Energy*. Elsevier. https://www.sciencedirect.com/science/article/abs/pii/S2211285524002337