ASSISTANCE SYSTEM FOR PEOPLE WITH VISUAL IMPAIRMENTS USING IoT

A Project report submitted in partial fulfillment of the requirements for

the award of the degree of

BACHELOR OF TECHNOLOGY

IN

INFORMATION TECHNOLOGY

Submitted by

K.Vishnu Vardhini (320126511081)

B.Saketh (320126511064)

S.Kavitha (320126511104)

K.Akash (320126511078)

Under the guidance of

Mr.M.V Kishore

Assistant Professor

DEPARTMENT OF INFORMATION TECHNOLOGY

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES

(UGC AUTONOMOUS)

(*Permanently Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade*)

Sangivalasa, Bheemili Mandal, Visakhapatnam dist.(AP)

2023-2024

DEPARTMENT OF INFORMATION TECHNOLOGY

ANIL NEERUKONDA INSTITUTE OF TECHNOLOGY AND SCIENCES

(UGC AUTONOMOUS)

(*Affiliated to AU, Approved by AICTE and Accredited by NBA & NAAC with 'A' Grade*)

Sangivalasa, Bheemili Mandal, Visakhapatnam dist.(A.P)

CERTIFICATE

This is to certify that the project report entitled "**ASSISTANCE SYSTEM FOR PEOPLE WITH VISUAL IMPAIRMENTS USING IoT"** submitted by **K.Vishnu Vardhini(320126511081),B.Saketh(320126511064),S.Kavitha(320126511104),K.Akash(320126511078)** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology** in **Information Technology**of Anil Neerukonda Institute of technology and sciences (A), Visakhapatnam is a record of bonafide work carried out under my guidance and supervision.

Project Guide	**Head of the Department**
Mr.M.V KISHORE	**Dr.M.REKHA SUNDARI**
Assistant Professor	Professor
Department of IT	Department of IT
ANITS	ANITS

DECLARATION

We, **K.Vishnu Vardhini(320126511081), B. Saketh(320126511064),S. Kavitha(320126511104), K. Akash(320126511078)** of final semester B.Tech., in the department of Information Technology from ANITS, Visakhapatnam, hereby declare that the project work entitled **"ASSISTANCE SYSTEM FOR PEOPLE WITH VISUAL IMPAIRMENTS USING IoT"** carried out by us and submitted in partial fulfillment of the requirements for the award of **Bachelor of Technology in Information Technology**, under Anil Neerukonda Institute of Technology & Sciences(A) during the academic year 2023-2024 and has not been submitted to any other university for the award of any kind of degree.

K.Vishnu Vardhini	320126511081
B.Saketh	320126511064
S.Kavitha	320126511104
K.Akash	320126511078

ACKNOWLEDGEMENT

We would like to express our deep gratitude to our project guide **Mr.M.V KISHORE,** Department of Information Technology, ANITS, for his guidance, unsurpassed knowledge, and immense encouragement. We are grateful to **Dr.M.REKHA SUNDARI**, Head of the Department of Information Technology, for providing us with the required facilities for the completion of the project work.

We are very much thankful to the **Principal and Management, ANITS, Sangivalasa,** for their encouragement and cooperation in carrying out this work.

We thank all the **teaching faculty and non-teaching staff** of the Department of IT, whose suggestions during reviews helped us in the accomplishment of our project.

We would like to thank our parents, friends, and classmates for their encouragement throughout our project period. At last, but not the least, we thank everyone for supporting us directly or indirectly in completing this project successfully.

PROJECT STUDENTS

K.Vishnu Vardhini (320126511081),

B.Saketh (320126511064),

S.Kavitha (320126511104)

K.Akash (320126511078),

CONTENTS

ABSTRACT

The "Assistance system for people with visual impairments using IoT" is a groundbreaking assistive technology designed to revolutionize the daily experiences of visually impaired individuals. By integrating state-of-the-art technology into wearable devices such as smart glasses and a smart stick capable of detecting obstacles, this innovative kit offers a comprehensive solution to enhance independence and confidence.

One of its most remarkable features is its ability to detect currency and provide audible feedback regarding its denomination. This functionality, powered by artificial intelligence and computer vision, empowers visually impaired individuals to effortlessly handle money in their day-to-day lives, offering a level of autonomy previously unimaginable.

In this article, we will explore the extraordinary capabilities and advantages of the Assistance system for visually impaired individuals using IoT. From its advanced detection of obstacles to its seamless currency recognition, we will delve into how this transformative technology is reshaping the way visually impaired individuals interact with currency, providing them with newfound freedom and security.

LIST OF FIGURES

LIST OF ABBREVIATIONS

IoT	Internet of things
SVM	Support Vector Machine
SIFT	Scale-invariant feature
ORB	Oriented FAST and Rotated BRIEF
AI	Artificial Intelligence
ML	Machine Learning
DL	Deep Learning
HOG	Histogram of Oriented Gradients
OpenCV	Open-Source Computer Vision
RAM	Random Access Memory
GB	Giga Bytes
UML	Unified Modeling Language
IDE	Integrated Development Environment

CHAPTER 1
INTRODUCTION

1. INTRODUCTION

1.1 INTRODUCTION

The "Assistance system for people with visual impairments using IoT" represents a groundbreaking innovation poised to revolutionize the daily experiences of individuals with visual impairments. This cutting-edge technology integrates wearable devices such as smart glasses and a smart stick equipped with IoT capabilities, specifically designed to enhance independence and facilitate navigation for visually impaired individuals. A standout feature of this assistive technology is its remarkable ability to detect currency and provide audible feedback regarding its denomination. Through the utilization of artificial intelligence and computer vision, the system empowers visually impaired users to confidently handle money in their day-to-day activities, thereby fostering greater autonomy and convenience. In this article, we explore the extraordinary features and advantages of the Assistance system for visually impaired individuals using IoT, illustrating how it is fundamentally reshaping the way individuals with visual impairments interact with currency. By leveraging advanced IoT technologies, this innovative solution not only enhances accessibility but also promotes inclusivity and empowerment for individuals with visual impairments, ultimately enriching their quality of life.

1.2 MOTIVATION

The "Assistance system for people with visual impairments using IoT" was conceived from a real-life incident that underscored the challenges faced by visually impaired individuals in their daily lives. Observing numerous blind individuals urgently requiring medication and experiencing challenges in accurately receiving the required change at a nearby store specifically during my visits underscored the need for a theoretical solution to help the visually impaired make safe and independent currency transactions. The idea of smart glasses incorporating advanced technology to recognize currency and offer audio feedback was developed through this experience. Furthermore, the smart stick, which was used to detect obstacles and improve navigation among the visually impaired, was the hallmark of the plan.

Combining these technologies was designed to solve the unique problems of visually impaired people, especially in financial areas and movement.

The primary technology that fits this novel solution is the implementation of IoT technologies in wearable glasses, like smart glasses and a stick with intelligent sensors. These devices powered by IoT and computer vision algorithms are utilized for real-time visualization of both visual data. Smart glasses use a camera to capture currency note images, and IoT algorithms process images and recognize the correct currency note through denomination. Then, system provides, By providing real-time assistance in currency recognition and obstacle detection, the Assistance system for visually impaired individuals using IoT not only addresses immediate challenges but also fosters inclusivity and autonomy in daily activities.

1.3 PROBLEM DEFINITION

The problem statement for the above abstract would focus on the challenges faced by visually impaired individuals in managing currency transactions independently and safely. It would highlight the prevalence of incidents where visually impaired individuals encounter difficulties in accurately receiving change during financial transactions, leading to potential exploitation or scams. The problem statement would underscore the critical need for a technological solution to empower visually impaired individuals with the ability to recognize currency denominations and navigate financial transactions with confidence and autonomy. It would emphasize the importance of addressing these challenges to enhance the daily lives and financial independence of visually impaired individuals, ultimately contributing to their overall well-being and inclusion in society.

1.4 OBJECTIVE OF THE PROJECT

The objective of the project is to develop an assistive technology solution, leveraging IoT technology, to enhance the daily lives of visually impaired individuals. Specifically, the project aims to provide a wearable device in the form of smart glasses and a smart stick equipped with advanced features such as obstacle detection and currency recognition. By incorporating artificial intelligence and computer vision algorithms, the system seeks to empower visually impaired individuals with newfound independence and confidence in handling currency during their day-to-day activities. The primary goal is to address the challenges faced by visually impaired individuals in managing money, thereby improving their overall quality of life.

1.5 LIMITATIONS OF THE PROJECT

1.5.1 Bulkiness

Despite efforts to streamline the design, the wearable components of the system, such as the smart glasses and smart stick, may still possess a certain degree of bulkiness. While strides have been made to minimize the size and weight of these devices for improved comfort and convenience, the inherent technology and functionality may contribute to a level of bulkiness that could potentially affect user comfort and mobility.

1.5.2 Battery life

Despite efforts to make the system user-friendly and portable, one limitation is the need for frequent battery recharges. Wearable devices, such as smart glasses and smart sticks, rely on battery power to function, which may require regular charging sessions, potentially causing inconvenience to users. The current iteration of the project operates with constrained memory capacity. This limitation may restrict the amount of data or updates that can be stored and processed by the system.

1.5.3 Can't Detect Fake Currency Notes

Assistance system for people with visual impairments using IoT can't detect fake currency notes. It was the main drawback of this project unless everything was fine and the problem statement was achieved.

1.6 ORGANIZATION OF DOCUMENTATION

1. Introduction:

Brief overview of the Assistance system for visually impaired individuals using IoT. Explanation of its purpose and significance in enhancing the lives of visually impaired individuals.

2. Background:

Discussion on the challenges faced by visually impaired individuals in handling currency. Introduction to existing assistive technologies and their limitations.

3. Overview of the Assistance System

Description of the components, smart glasses, and smart stick. Explanation of how the system utilizes IoT technology.

4. Features of the Assistance System

Detailed explanation of each feature and obstacle detection capability of the smart stick. Currency detection and denomination feedback are provided by the system. Utilization of artificial intelligence and computer vision for enhanced functionality.

5. Benefits for Visually Impaired Individuals

Exploration of the newfound independence and confidence offered by the system. Real-life scenarios showcasing how the system improves daily activities related to currency management.

6. Technical Implementation

Overview of the technology stack used in developing the Assistance system. Explanation of how IoT, ML, and computer vision are integrated into the system.

7. User Experience

Discussion on the ease of use and user-friendliness of the system. Feedback from visually impaired individuals who have used the system.

8. Future Developments

Potential enhancements and updates to address current limitations. Opportunities for further research and innovation in assistive technology for the visually impaired.

9. Conclusion

Recap of the key features and benefits of the Assistance system. Final thoughts on its impact on reshaping the interaction of visually impaired individuals with currency.

1.6.10 References

Citations for any research papers, articles, or resources referenced throughout the documentation.

CHAPTER 2

LITERATURE SURVEY

2. LITERATURE SURVEY

2.1 INTRODUCTION

The literature surrounding assistive technologies for individuals with visual impairments has long emphasized the need for innovative solutions to enhance independence and quality of life. Visual impairment presents unique challenges in various aspects of daily living, including mobility, communication, and financial transactions. Traditional assistive devices, while helpful, often lack the sophistication and integration of cutting-edge technologies like IoT and Machine learning. The "Assistance system for people with visual impairments using IoT" represents a significant advancement in the field, offering a comprehensive solution to address the specific challenges faced by visually impaired individuals, particularly in handling currency transactions. This system combines wearable devices—smart glasses and a smart stick—with advanced IoT capabilities to provide real-time assistance and feedback.

2.2 EXISTING SYSTEM

Essentially, existing systems can detect specific currency notes using separately connected modules to provide voice feedback on the denomination. However, it cannot recognize human faces or detect obstacles. Additionally, the current setup is not ideal, as it requires the proximity of the camera to the currency note for detection to be successful. To enhance the system, we may consider integrating facial recognition and obstacle detection capabilities, as well as reducing the bulkiness of the detection modules.

R. Radhika et al .[1] if someone is visually handicapped, a smart stick featuring infrared, ultrasonic, and water sensors has been developed to assist in obstacle detection. These sensors can identify obstacles up to roughly 3 meters.

M.H. Mahmud et al .[4] reduce the effects of blindness by building automated technology based on microcontrollers that can confirm a blind person's ability to instantaneously notice impediments in front of them. The hardware is a proximity sensor, wet detector, ping sonar sensor, micro pager motor, and other components.

R. Sheth et al. [19] designed the stick to keep it structurally similar i.e. thin, lightweight, and easy to handle, yet give active feedback to the operator regarding hazards in his walking path.

C.S. Kher et al .[20] the ultrasonic detectors mounted on the smart white cane are situated in a way that can detect potholes and pits. Android handsets, RFID, and infrared sensors are utilized, they present a system of navigating for the visually handicapped that focuses on giving speech output for preventing obstacles and navigation. There are infrared proximity sensors on the device. Both public buildings and the walking sticks employed by the visually impaired are equipped with RFID tags. Through Bluetooth, this gadget is linked to an Android mobile device. barriers at low lying and knee level, as well as those above the waist, including staircases and downslopes. Pre-recorded sound cues and haptic feedback in vibrations alert the operator to the same.

Ahmed et al .[8] created the using an ORB algorithm. Binary descriptors can be obtained by applying ORB. The improved performance of this new algorithm. The accuracy and outcomes of the prior ROI extraction and template matching are superior. There is also a platform for Android applications designed for people who are blind or visually impaired.

Imad et al.'s proposed study [5] uses deep neural networks (DNNs). The primary features of the Pakistani dollar notes are extracted via live input after the system has been trained using the Alex-net architecture. The master features are classified using the Support Vector Machine (SVM) methodology, while the discriminating features are extracted using the HOG technique.

Snehal et al. [6] have developed and implemented the scale-invariant feature SIFT algorithm, which extracts the primary feature—the color discriminatory feature—among the currency notes data sets. This results in more accurate and reliable results and an audio output feature.

2.3 DISADVANTAGES OF THE EXISTING SYSTEM

2.3.1 Limited Detection Range

Many of the existing systems, such as those employing infrared, ultrasonic, or water sensors, have a restricted detection range. For instance, some systems can only identify obstacles up to a distance of roughly 3 meters (Radhika et al. [1]). This limitation may not provide sufficient advance warning to users, particularly in crowded or fast-paced environments.

2.3.2 Dependency on Specific Environments

Some systems, such as those utilizing RFID tags in public buildings or on walking sticks, are reliant on specific environmental setups (Kher et al. [20]). This dependency may limit the effectiveness of the system when navigating in areas lacking the required infrastructure, such as remote or outdoor locations.

2.3.3 Hardware Complexity

Certain systems involve the integration of multiple sensors and components, such as microcontrollers, proximity sensors, and sonar sensors (Mahmud et al. [4]). The complexity of such hardware setups may increase the risk of technical failures, maintenance issues, and overall system reliability.

2.3.4 Lack of Customization

Some systems rely on pre-recorded sound cues or fixed algorithms for obstacle detection and currency recognition (Kher et al. [20], Snehal et al. [6]). These predetermined features may not cater to the diverse needs and preferences of visually impaired individuals, limiting the adaptability and personalization of the assistive technology.

2.3.5 Performance Variability

The accuracy and reliability of certain systems may vary depending on factors such as lighting conditions, currency variations, or the presence of background clutter (Imad et al. [5], Snehal et al. [6]). Inconsistent performance may undermine user confidence and satisfaction with the assistive technology, particularly in critical situations requiring precise obstacle detection or currency identification.

2.4 PROPOSED SYSTEM

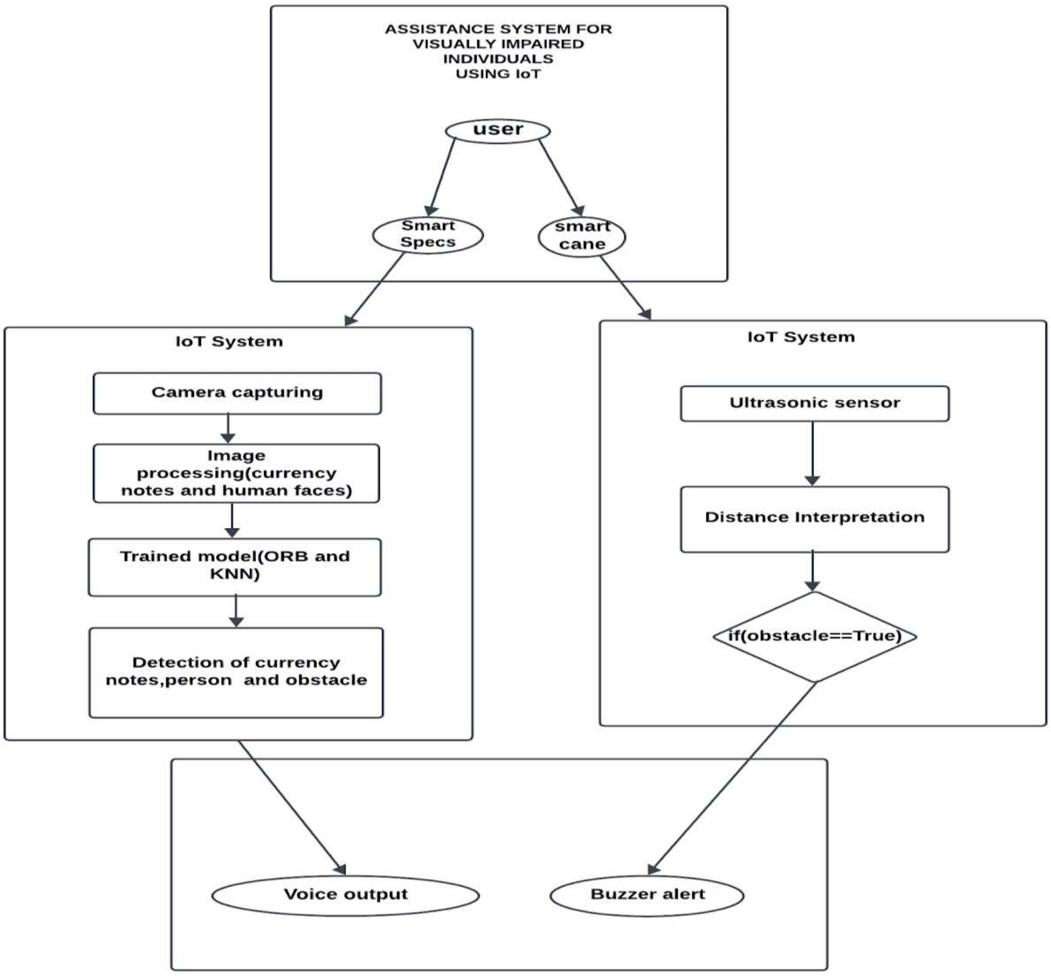


Fig 1: proposed system

2.4.1 Reading Images

This Step contains reading images from the camera. There are various methods to obtain an image, such as using a scanner or a digital camera. All features must take part in the final image, which will be converted to RGB. We have used OpenCV-python(cv2) for reading images.

2.4.2 Image preprocessing

Pre-processing is done to boost an image's visual expression and enhance the impact of a dataset. Pre-processing images is typically the first step needed to extract relevant information and data. Image pre-processing, also known as image restoration, comprises the introduction of noise, degradation, and distortion during the imaging process. Pre-processing images can improve optical inspection accuracy.

2.4.3 Feature Extraction

Reads the captured image using OpenCV. Uses the ORB feature detector and descriptor to extract key points and descriptors from the image. Therefore, it is expected that when the features that are extracted is well-selected; then, the total collection of these features will complete its function of input data by swapping the input data for the associated information.

2.4.4 Data for testing

Then, you should Get pictures of a variety of denominations of the currency you want the code to be able to recognize. Prioritize taking accurate, well-lit images that conform to practical circumstances. In the section of the real-time users utilizing the camera element of the IoT, the data was retrieved.

2.4.5 Algorithm for matching

Considering the already extracted several attributes, this phase recognizes or distinguishes the currencies of dissimilar denominations. The text-to-audio conversion will be sent to the visually impaired individual based on their classification.

2.4.6 Dataset details

Fig 2: currency notes

2.4.7 Model Training

2.4.7.1 KNN Algorithm

1. Data Preparation

This Step contains reading images from the camera. There are various methods to obtain an image, such as using a scanner or a digital camera. All features must take part in the final image, which will be converted to RGB. We have used OpenCV-python(cv2) for reading images.

2. Data Splitting and Preprocessing

Pre-processing is done to boost an image's visual expression and enhance the impact of a dataset. Pre-processing images is typically the first step needed to extract relevant information and data. Image pre-processing, also known as image restoration, comprises the introduction of noise, degradation, and distortion during the imaging process. Pre-processing images can improve optical inspection accuracy.

3. Feature extraction

Reads the captured image using OpenCV. Uses the ORB (Oriented FAST and Rotated BRIEF) feature detector and descriptor to extract key points and descriptors from the image. It is anticipated that when the features that are extracted are appropriately chosen, the collection of these features will carry out the intended function by substituting the input data's associated information for the total input. Here KNN is used in person detection.

4. Testing and Algorithm for Matching

Gather high-quality images of various currency denominations that you want the code to recognize and persons to detect. Ensure the images are clear, well-lit, and representative of typical conditions. The data is selected from the real-time users using the camera module of the IoT. In light of the many attributes extracted, this phase detects or classifies the currencies of dissimilar denominations. The text-to-audio conversion will be sent to the visually impaired individual based on their classification.

2.4.7.2 Feature matching using ORB algorithm in Python-OpenCV

ORB is a fusion between FAST key point detector and BRIEF descriptor with many modifications to improve the performance. FAST is Features from Accelerated Segment Test, which is used to detect features from the given image. It also uses a pyramid to produce multiscale features. Now it doesn't compute the orientation and descriptors for the features, so this is where BRIEF comes in the role.

ORB uses BRIEF descriptors, but the BRIEF performs poorly with rotation. So, what ORB does is rotate the BRIEF according to the orientation of key points. Using the orientation of the patch, its rotation matrix is found and rotates the BRIEF to get the rotated version. ORB is an efficient alternative to SIFT or SURF algorithms used for feature extraction, computation cost, matching performance, and mainly patents. SIFT and SURF are patented, and you are supposed to pay them for their use. But ORB is not patented.

Algorithm

1. Take the query image and convert it to grayscale.
2. Now Initialize the ORB detector and detect the key points in the query image and scene.
3. Compute the descriptors belonging to both the images.
4. Match the key points using Brute Force Matcher.
5. Show the matched images.

ORB performs as well as SIFT on the task of feature detection (and is better than SURF) while being almost two orders of magnitude faster. ORB builds on the well-known FAST key point detector and the BRIEF descriptor. Both techniques are attractive because of their good performance and low cost. ORB's main contributions are as follows:

- The addition of a fast and accurate orientation component to FAST
- The efficient computation of oriented BRIEF features
- Analysis of variance and correlation of oriented BRIEF features
- A learning method for decorrelating BRIEF features under rotational invariance, leading to better performance in nearest-neighbour applications.

Fast (Features from Accelerated and Segments Test)

Given a pixel p in an array, fast compares the brightness of p to the surrounding 16 pixels that are in a small circle around p. Pixels in the circle are sorted into three classes lighter than p, darker than p, or similar to p . If more than 8 pixels are darker or brighter than p then it is a key point. So the FAST key point gives the information of the location of determining edges in an image.

Brief(Binary robust independent elementary feature)

The brief takes all key points found by the fast algorithm and converts it into a binary feature vector so that together they can represent an object. Binary features vector also known as binary feature descriptor is a feature vector that only contains 1 and 0. In brief, each keypoint is described by a feature vector which is 128–512 bits string.

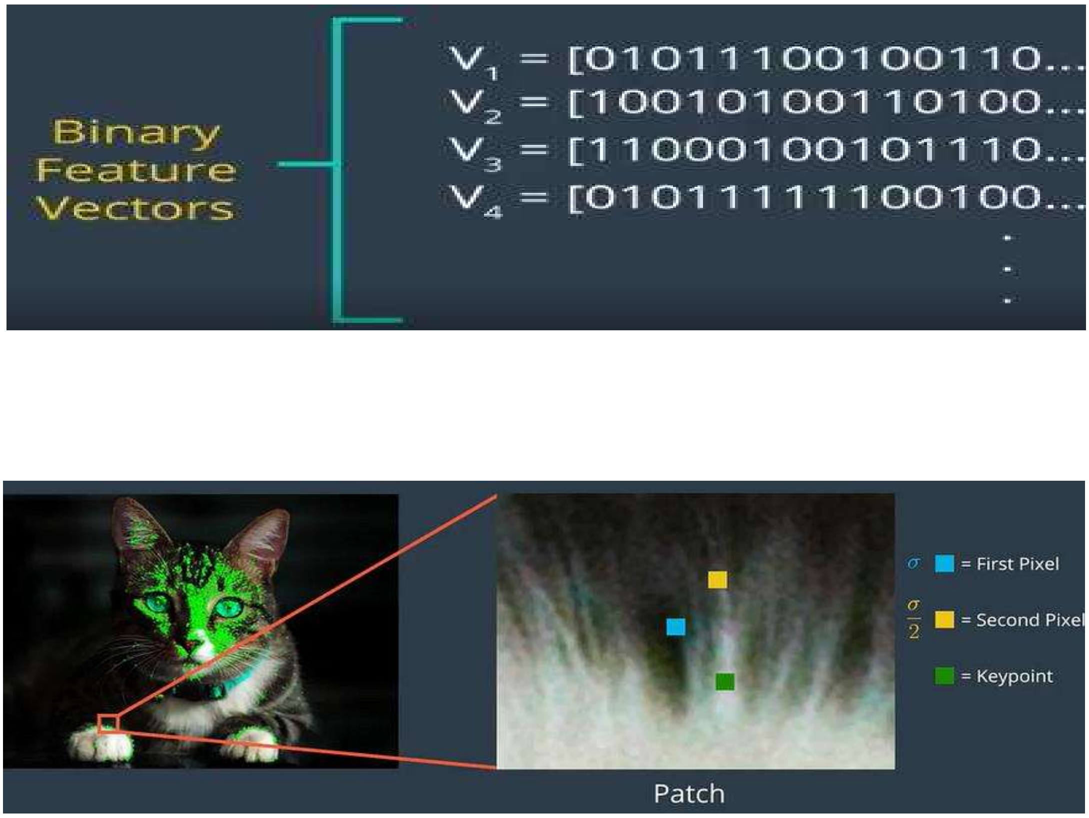


Fig 3: BRIEF

2.5 CONCLUSION

In conclusion, the existing IoT-based methods for assisting visually challenged individuals exhibit notable advancements in enhancing mobility, obstacle detection, and currency recognition. However, these systems are not without limitations, such as limited detection range, dependency on specific environments, hardware complexity, lack of customization, and performance variability. The proposed "Assistance system for people with visual impairments using IoT" seeks to address these shortcomings by integrating cutting-edge technologies like artificial intelligence, computer vision, and machine learning. Through image preprocessing, feature extraction, and algorithmic matching, the proposed system aims to accurately identify currency denominations and provide real-time auditory feedback, thereby empowering visually impaired individuals with increased independence and confidence in managing financial transactions. By leveraging the capabilities of IoT and machine learning, this system represents a significant step forward in reshaping the way visually impaired individuals interact with currency and navigate their daily lives, offering a comprehensive and sophisticated solution to address their specific needs and challenges.

CHAPTER 3
ANALYSIS

3. ANALYSIS

3.1 INTRODUCTION

Requirement Engineering is a fundamental process that plays a vital role in software development. It involves the identification, definition, and management of the system requirements necessary to meet the client's expectations. These requirements comprise of features, functions, and limitations that the system must meet. The process of Requirement Engineering comprises two key activities, namely requirement elicitation and analysis. Requirement elicitation involves the collection and documentation of the client's requirements using various techniques such as surveys, interviews, observations, and workshops. The main objective is to produce a detailed and accurate system specification that the client can understand and approve.

3.2 SOFTWARE REQUIREMENT SPECIFICATION

3.2.1. Introduction

1.purpose

The purpose of this document is to outline the software requirements for the development of the "Assistance System for People with Visual Impairments Using IoT."

2. Scope

The system comprises two main components: Smart spectacles and Smart cane. Leveraging Raspberry Pi, Arduino, and camera technology, it integrates sophisticated algorithms for computer vision, including currency note recognition, person detection, and obstacle detection.

2.1 Definitions, Acronyms, and Abbreviations

IoT: Internet of Things

SRS: Software Requirement Specification

Pi: Raspberry Pi single-board computer

Arduino: Microcontroller board used for hardware interaction.

GUI: Graphical User Interface

2.2 Reference

IEEE Std 830-1998 - IEEE Recommended Practice for Software Requirements Specifications

3. Overall Description

1. Product Perspective

The "Assistance System for People with Visual Impairments Using IoT" is a standalone system designed to operate in conjunction with smart spectacles and a smart cane. It interfaces with various hardware components such as Raspberry Pi, Arduino, Pi Camera module, speaker module, ultrasonic sensor, and buzzer to provide real-time assistance to visually impaired users.

2. Product Functions

Currency note recognition: Identifying currency notes from live camera feed and providing auditory feedback. Person detection: Detecting individuals within the user's vicinity and providing alert signals.

Obstacle detection: Identifying obstacles in the user's path using ultrasonic sensors and alerting the user. Voice-based feedback: Providing auditory cues and voice-based instructions to assist users in navigation and financial transactions.

3. User Classes and Characteristics

The primary users of this system are individuals with visual impairments. They may vary in age, technological proficiency, and degree of visual impairment.

4. Operating Environment

The system operates in indoor and outdoor environments where there is access to the internet and electricity. It should function effectively in various lighting conditions and environmental settings.

5. Design and Implementation Constraints

The system must be designed to be lightweight and portable for easy integration into smart spectacles and a smart cane. It should consume minimal power to ensure prolonged battery life.

6. Assumptions and Dependencies

It is assumed that users have access to a stable internet connection for software updates and data synchronization. The system depends on the proper functioning of hardware components such as Raspberry Pi, Arduino, and camera modules.

4. Specific Requirement

1. External Interface Requirement

1.1 User Interfaces

The system should provide a user-friendly interface with the following features:

Voice-based interaction for navigation and control.

Auditory feedback for currency notes recognition, person detection, and obstacle detection.

Minimalistic graphical user interface (GUI) for configuration and customization.

1.2 Hardware Interfaces

The software should interface with the following hardware components:

Raspberry Pi single-board computer

Arduino microcontroller

Pi Camera module

Ultrasonic sensor

Speaker module

Buzzer

2. Functional Requirements

2.1 Currency Note Recognition

The system shall analyse the live camera feed to recognize currency notes. It shall provide auditory feedback indicating the denomination of the detected currency notes. The software shall be capable of recognizing multiple currencies based on predefined patterns.

2.2 Person Detection

The system shall use computer vision algorithms to detect individuals in the user's vicinity. It shall provide alert signals to notify the user of the presence of other individuals. The software should differentiate between stationary objects and moving individuals.

2.3 Obstacle Detection

The system shall utilize ultrasonic sensors to detect obstacles in the user's path. It shall generate buzzer sounds at varying frequencies to indicate the proximity and nature of obstacles. The software should distinguish between static and dynamic obstacles.

2.4 Voice-based Feedback

The system shall provide voice-based instructions and feedback to guide users in navigation and financial transactions. It shall offer customizable voice settings for language, tone, and volume. The software should adapt its instructions based on user preferences and environmental conditions.

5. Performance Requirements

1. Response Time

The system shall provide real-time feedback with minimal latency (< 500 milliseconds) for currency note recognition, person detection, and obstacle detection.

2. Accuracy

The software shall achieve a high level of accuracy (> 95%) in currency note recognition and person detection under varying lighting and environmental conditions.Obstacle detection accuracy should be sufficient to ensure user safety and navigation efficiency.

3. Software System Attributes

3.1 Reliability

The system shall be reliable and robust, capable of operating continuously without frequent failures or interruptions. It should include mechanisms for error detection, handling, and recovery.

3.2 Security

The software should incorporate security measures to protect user data and privacy.

It shall employ encryption techniques for data transmission and storage.

3.3 Portability

The system should be portable and compatible with different hardware configurations, including smart spectacles and smart canes from various manufacturers.

3.4 Maintainability

The software should be modular and well-documented to facilitate ease of maintenance and future updates.

It should include diagnostic tools for troubleshooting and debugging purposes.

6. Conclusion

This Software Requirement Specification (SRS) document outlines the detailed requirements for the development of the "Assistance System for People with Visual Impairments Using IoT." By adhering to these requirements, the system aims to provide a comprehensive and user-friendly assistive technology solution to enhance accessibility and independence for individuals with visual impairments.

3.2.2 User Requirements

Requirements can be divided into two major categories:
Functional Requirements.
Non-Functional Requirements.

Functional Requirements

Functional requirements are specific actions and behaviors that a software system must perform to meet users' needs. They define the system's capabilities and features and are expressed in terms of input, processing, and output. To ensure accuracy, they must be defined clearly and validated. The functional requirements for the proposed system are as follows:

Data Preprocessing: The system must be able to preprocess images of currency notes and face , including filtering,thresholding,resizing,contouring,scaling and merging.

Feature Extraction: The system must extract relevant features from the preprocessed images to enable accurate classification of faces and send it to the user through audio.

Classification: The system must be able to accurately classify the images into one of the saved faces from the user.

Result Visualization: The system must send the output through the audio jack of raspberry pi.

Non-Functional Requirements

Non-functional requirements are the criteria that define the system's performance, quality, and behavior, rather than its specific functionality. These requirements describe the system's characteristics, such as its reliability, security, usability, performance, scalability, and maintainability, and are essential for ensuring that the system meets the user’s expectations and needs. The non-functional requirements for the proposed system are as follows:

Accuracy: The model should achieve high accuracy in classifying the currency notes and the face.

Performance: The model should perform efficiently and quickly while classifying currency notes and the face.

Robustness: The model should be robust to different lighting and contrast conditions that may occur in real-world scenarios.

Scalability: The model should be able to handle all the currency notes and the saved faces from the user for classification without significant decrease in performance.

TECHNICAL REQUIREMENTS

The technical requirements for this project are mentioned below:

Hardware Requirements

Software Requirements

3.2.3. Hardware Requirements

Processor: A high-performance CPU with multiple cores is essential for handling preprocessing, feature extraction, and model training tasks efficiently. Ensure that the CPU has sufficient processing power to handle the computational demands of training and inference for multiple algorithms simultaneously. Look for CPUs with high clock speeds and multiple cores, such as Intel Core i7 or i9 series, or AMD Ryzen series, to ensure optimal performance.

RAM: Adequate RAM is crucial for storing data, model parameters, and intermediate results during processing and model training. Aim for a minimum of 16 GB of RAM, with the flexibility to upgrade to 32 GB or more for larger datasets and complex ensemble models.

Storage: Utilize fast and reliable storage solutions, preferably SSDs, to store ECG datasets, model checkpoints, and intermediate results. Allocate sufficient storage space (at least 500 GB) to accommodate datasets, model files, and temporary files generated during processing and need external 32gb storage.

Raspberry Pi 4: The Raspberry Pi 4 is a versatile single-board computer renowned for its compact size and robust performance. With a quad-core ARM Cortex-A72 processor and up to 8GB of RAM, it offers ample computing power for various projects, from simple programming tasks to multimedia applications and IoT projects.

Arduino Uno: The Arduino Uno is a popular microcontroller board favored for its simplicity and versatility in electronics prototyping and DIY projects. It features an ATmega328P microcontroller, digital and analog input/output pins, and a USB interface for programming and communication. It's widely used for controlling sensors, motors, and other electronic components.

Ultrasonic Sensor: An ultrasonic sensor is a device that measures distance by emitting ultrasonic sound waves and calculating the time it takes for the waves to bounce back after hitting an object. It typically consists of a transmitter and receiver module and is commonly used in robotics, automation, and distance measurement applications. Ultrasonic sensors offer non-contact distance sensing capabilities with high accuracy and reliability.

Battery: A 9-volt battery is a compact, rectangular power source commonly used in small electronic devices and gadgets. It typically consists of six cylindrical or square cells connected in series, providing a total voltage output of 9 volts. With a standard terminal configuration, it's easy to install and replace in various devices. Despite its compact size, a 9-volt battery offers a relatively high energy density, making it suitable for powering devices with moderate power requirements.

3.2.4 Software Requirements

Python Programming Language: Python is an interpreted, high-level, general- purpose programming language. It is widely used in data science, machine learning, and artificial intelligence. It is necessary for the implementation of the project.

Scikit-Learn Library: Scikit-Learn is a popular machine learning library. It provides a wide range of machine-learning algorithms and tools. The Scikit-Learn library is necessary for the implementation of this method and also for accuracy calculation metrics.

Numpy and Pandas Libraries: Numpy and Pandas are popular libraries for data manipulation. They provide tools for handling arrays, matrices, and data frames. They are necessary for the implementation of the project

CV2: CV2 or OpenCV (Open Source Computer Vision) is a library of programming functions mainly aimed at real-time computer vision. It provides a wide range of functions for processing images and videos, including image filtering, edge detection, object detection, and image **segmentation Matplotlib:** Matplotlib is a Python plotting library used to create static, animated, and interactive visualizations in Python. It can be used for a variety of tasks such as creating line plots, scatter plots, bar plots, histograms, and more. Matplotlib provides a wide range of customization options, including the ability to add titles, labels, and annotations to your plots.

Scipy: SciPy is a comprehensive library for scientific computing in Python. It provides modules for optimization, linear algebra, integration, interpolation, and more. SciPy builds upon NumPy and extends its functionality, offering advanced mathematical algorithms and tools. It is widely used in scientific research, engineering, and data analysis for tasks ranging from signal processing to statistical analysis. With its vast array of functions, SciPy is a fundamental tool for numerical computing in Python.

os library: The os library in Python provides a way to interact with the operating system. It offers functions for performing tasks related to file and directory management, process management, and environment variables, among others. Some common functionalities provided by the os module include file and directory operations, path operations, process management, and handling environment variables.

Joblib: Joblib is a Python library that provides tools for pipelining Python functions, especially those that involve computation-heavy tasks. It is particularly useful for parallel computing and caching. Joblib offers simple and efficient tools for parallelizing computations using threads or processes, as well as for caching function outputs to disk to avoid redundant computations.

pyttsx3: Pyttsx3 is a Python library for text-to-speech (TTS) conversion. It provides a simple and cross-platform interface for converting text into spoken audio. Pyttsx3 supports various TTS engines and can be customized with different voices, rates, and volumes. It is commonly used in applications where Speech synthesis is required, such as in assistive technologies, virtual assistants, and accessibility tools.

subprocess: The subprocess module in Python provides a way to spawn new processes, interact with them, and handle input/output streams. It allows Python programs to run shell commands or other external processes, capture their output, and communicate with them using pipes or other inter-process communication mechanisms. The subprocess module is often used for tasks such as running system commands, launching external programs, and executing scripts in different languages. It provides a flexible and a powerful interface for process management and system integration.

Arduino IDE: The Arduino Integrated Development Environment (IDE) is a software platform used for programming Arduino boards. It provides a user-friendly interface for writing, compiling, and uploading code to Arduino microcontrollers. With a simple and intuitive interface, beginners and experienced developers alike can quickly prototype and develop various electronic projects. The IDE supports a wide range of libraries and examples, making it easier to integrate sensors, actuators, and other components into projects. Additionally, it offers features like syntax highlighting, serial monitoring, and debugging tools, facilitating the development process for Arduino enthusiasts.

3.3 CONCLUSION

The Software Requirement Specification (SRS) document delineates the essential software requirements for crafting the "Assistance System for People with Visual Impairments Using IoT," a pivotal endeavor aimed at bolstering accessibility and independence for individuals with visual impairments. Rooted in Requirement Engineering, this document elucidates the system's purpose, which revolves around identifying, defining, and managing features, functions, and limitations to align with client expectations. Comprising two primary components—Smart spectacles and Smart cane—the system harnesses Raspberry Pi, Arduino, and camera technology, amalgamating sophisticated algorithms for currency note recognition, person detection, and obstacle detection.

CHAPTER 4
DESIGN

4.DESIGN

4.1 INTRODUCTION

Unified Modeling Language (UML) diagrams are a visual representation of software systems, processes, or structures using standardized symbols and notation. UML provides a common language for software developers, designers, and stakeholders to communicate ideas, concepts, and designs throughout the software development lifecycle. Introduced in the mid-1990s, UML has become a widely adopted standard in the software engineering industry, aiding in the analysis, design, and documentation of complex systems.

4.2 UML DIAGRAMS

4.2.1 Use Case Diagram

A use case diagram is used to represent the dynamic behavior of a system. It encapsulates the system's functionality by incorporating use cases, actors, and their relationships. It models the tasks, services, and functions required by a system/subsystem of an application. It depicts the high-level functionality of a system and tells how the user handles a system.

Purpose of Use Case Diagram

The main purpose of a use case diagram is to portray the dynamic aspect of a system. It accumulates the system's requirements, which include both internal as well as external influences. It invokes persons, use cases, and several things that invoke the actors and elements accountable for the implementation of use case diagrams. It represents how an entity from the external environment can interact with a part of the system.

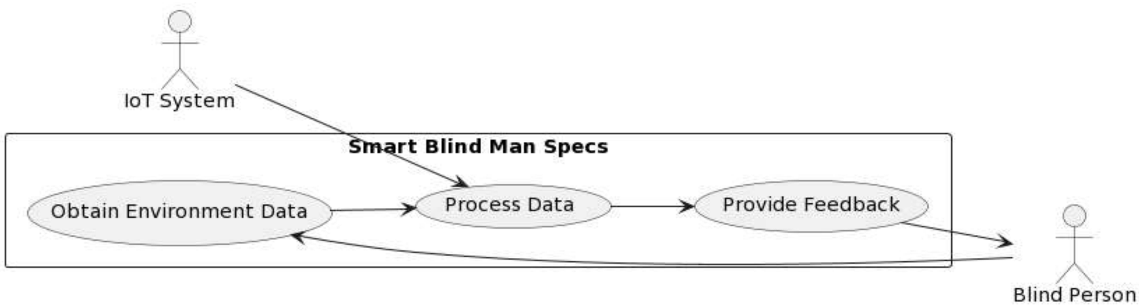


Fig 4: use case diagram for smart specs

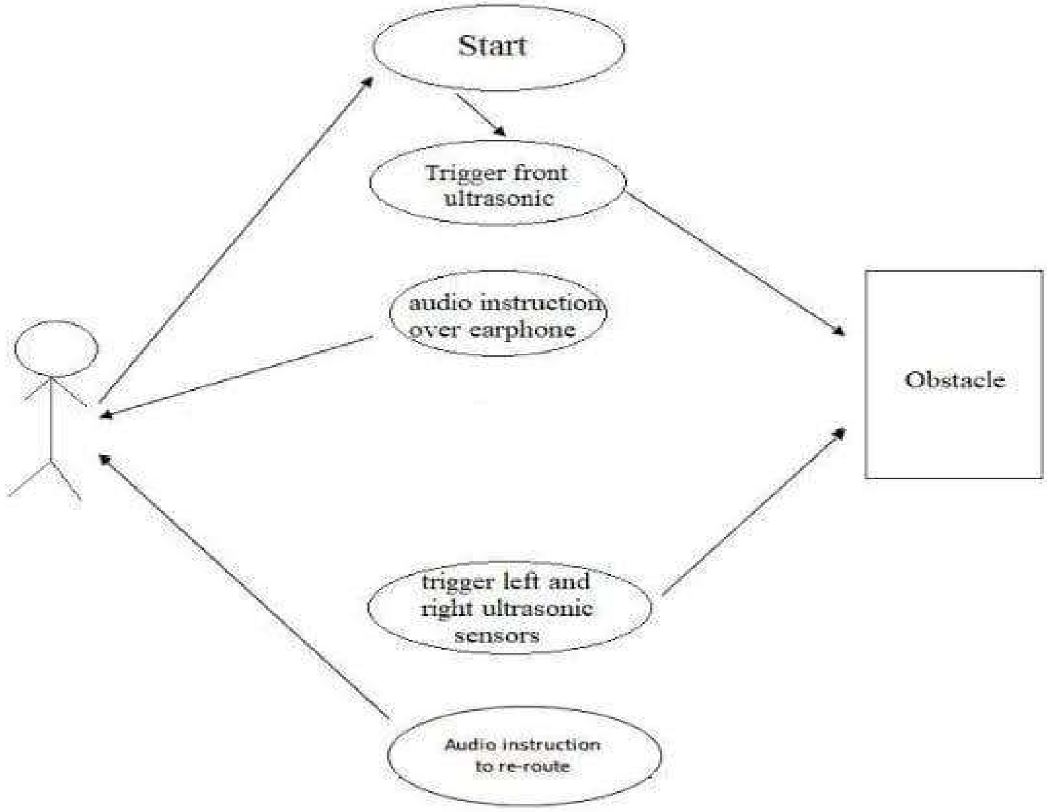


Fig 5: use case diagram of smart stick

4.2.2 Class Diagram

The class diagram is static. It represents the static view of an application. Class diagrams is not only used for visualizing, describing, and documenting different aspects of a system but also for constructing executable code of the software application. A class diagram describes the attributes and operations of a class and the constraints imposed on the system.

Purpose of Class Diagram

The Purpose of the class diagram can be summarized as:

Describe the responsibilities of a system.

The base for component and deployment diagrams.

Forward and reverse engineering.

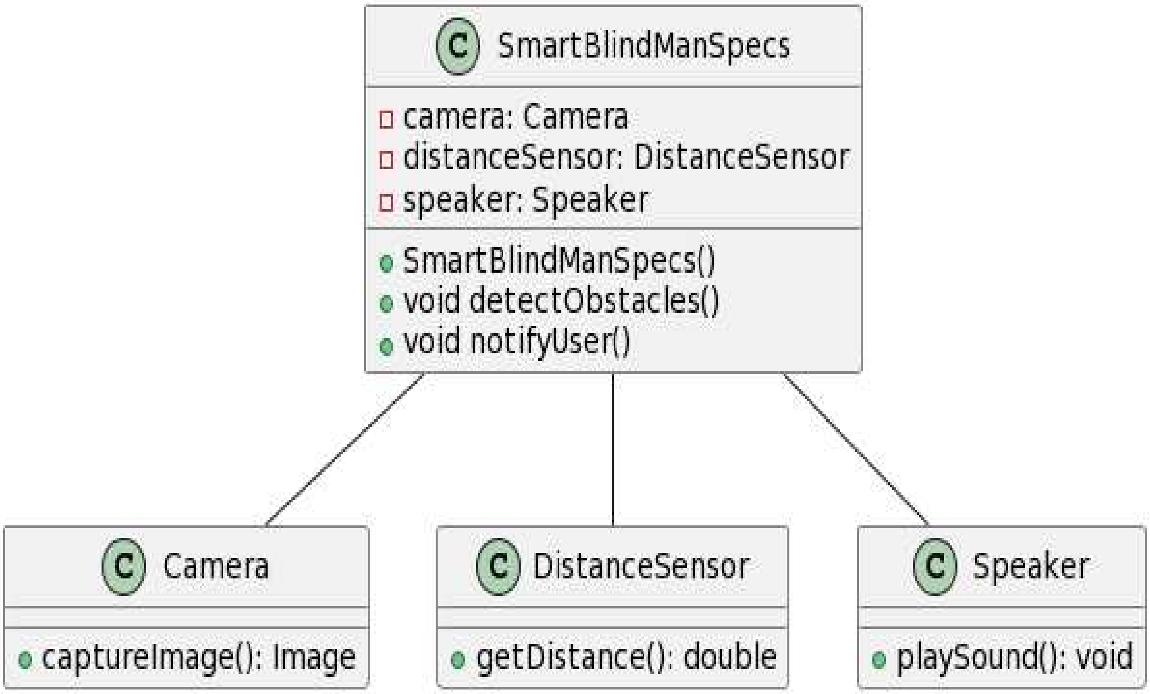


Fig 6: class diagram

4.2.3 Sequence Diagram

A sequence diagram is an event diagram. It shows the flow of messages between different objects or components over time and is often used to model the behavior of a single use case. They are useful for designing and communicating the flow of messages between objects in a software system, and for identifying potential problems or bottlenecks in the system's behaviour.

purpose of a sequence diagram

The purpose of a sequence diagram is to visualize the interactions between objects in a software system over time. It shows the order in which messages are exchanged between objects or components in a system and can be used to model the behaviour of a single use case or scenario.

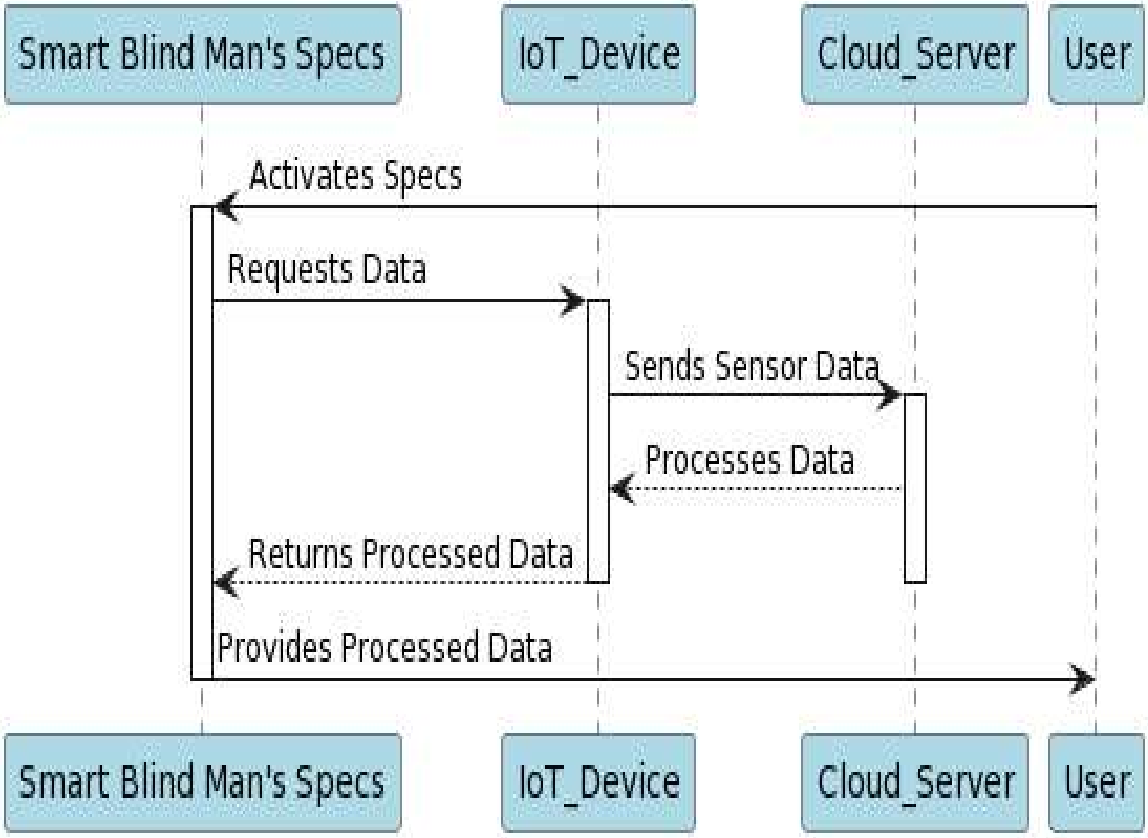


Fig 7: sequence diagram

4.2.4 Activity diagram

Activity diagrams are a type of diagram used in software engineering and business process modelling to visualize the flow of activities involved in a system or process. There are several types of nodes used in activity diagrams, including:

Initial node: Represents the starting point of the process or system.

Activity note: Represents an action or task that is performed as part of the process.

Decision node: Represents a branching point in the process, where a decision is made based on a certain condition or set of conditions.

Merge node: Represents a point where multiple paths in the process converge back into a single path.

Final nod e: Represents the endpoint of the process or system.

Edges in activity diagrams can be either control flows or object flows. Control flows represent the flow of control between nodes, while object flows represent the flow of data or objects between nodes.

Purpose of Activity Diagram

The purpose of an activity diagram is to provide a visual representation of a system or process, making it easier to understand and analyse.

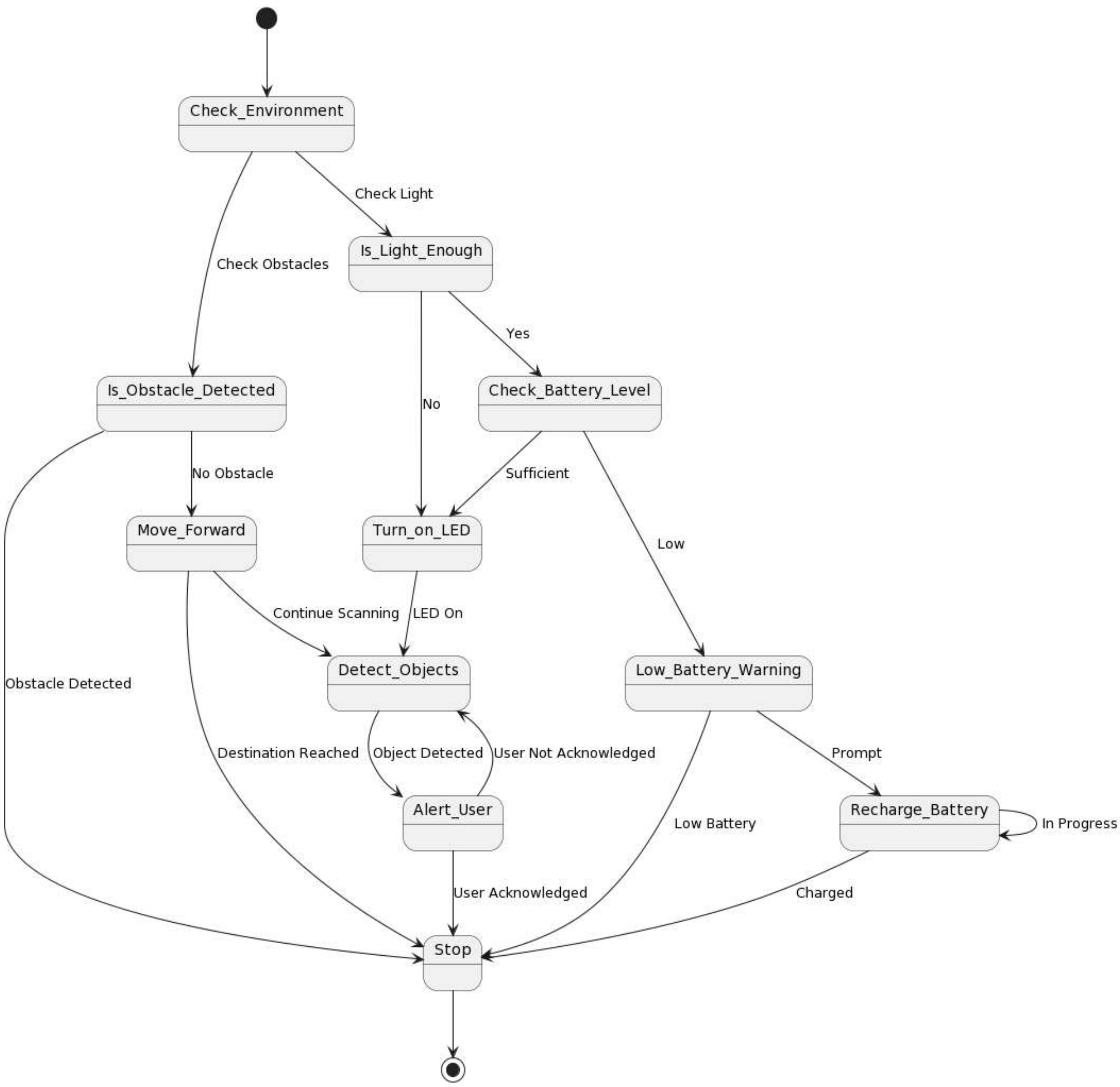


Fig 8: Activity Diagram

4.2.5 Architecture Diagram

An architecture diagram is a visual representation of the structure, components, and interactions within a system or application. It typically includes boxes representing different elements of the system, such as modules, layers, or components, and lines or arrows indicating the connections or relationships between them.

Purpose of Architecture diagram

Communication

Design Visualization

Documentation

Decision Making

System Understanding

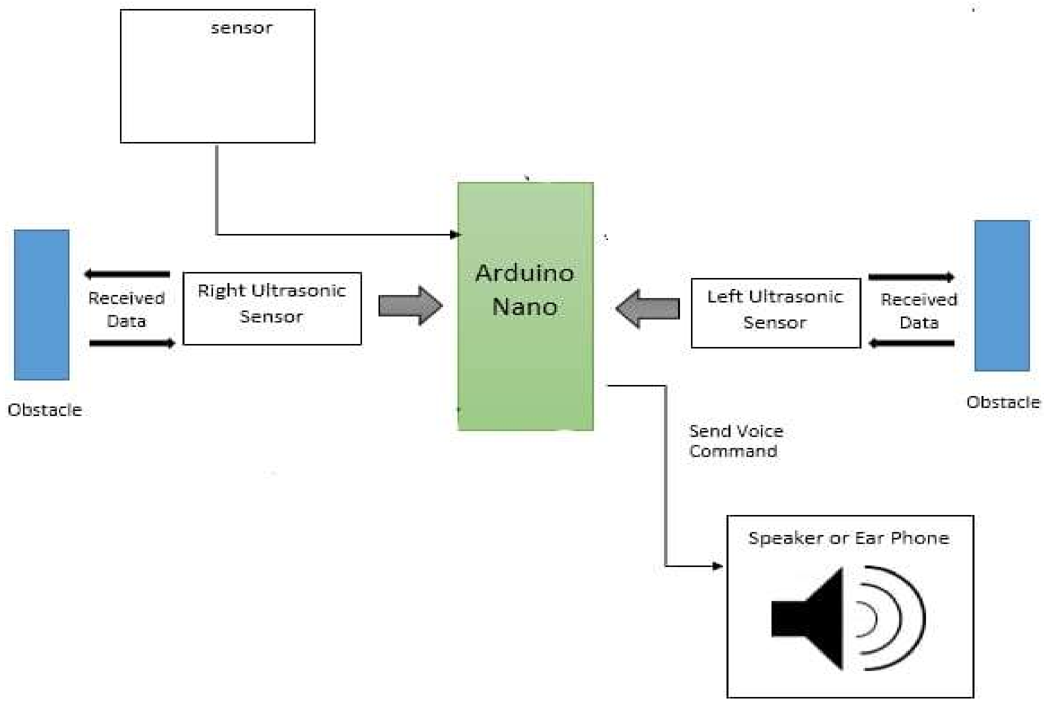


Fig 9: Architecture diagram of smart stick

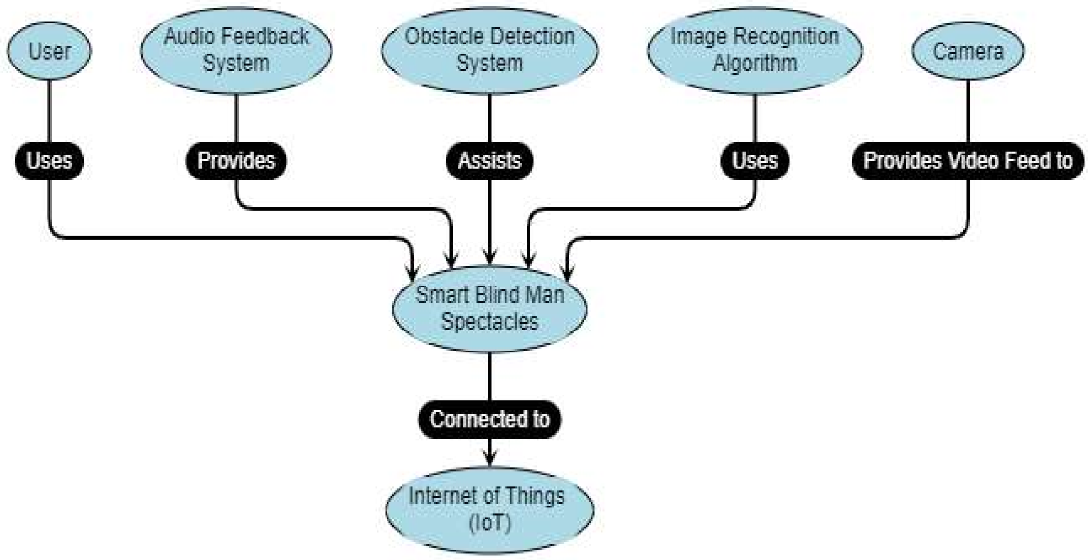


Fig 10: Architecture diagram of smart specs

4.2.6 Data flow diagram

A Data Flow Diagram (DFD) is a graphical representation of the flow of data within a system. It depicts how data moves through various processes, stores, and external entities within the system. In a DFD, processes are represented by rectangles, data stores by parallel lines, data flows by arrows, and external entities by squares.

Purpose of Data flow diagram

The purpose of a DFD is to provide a visual illustration of the system's data flow, allowing stakeholders to understand how data is input, processed, stored, and output within the system.

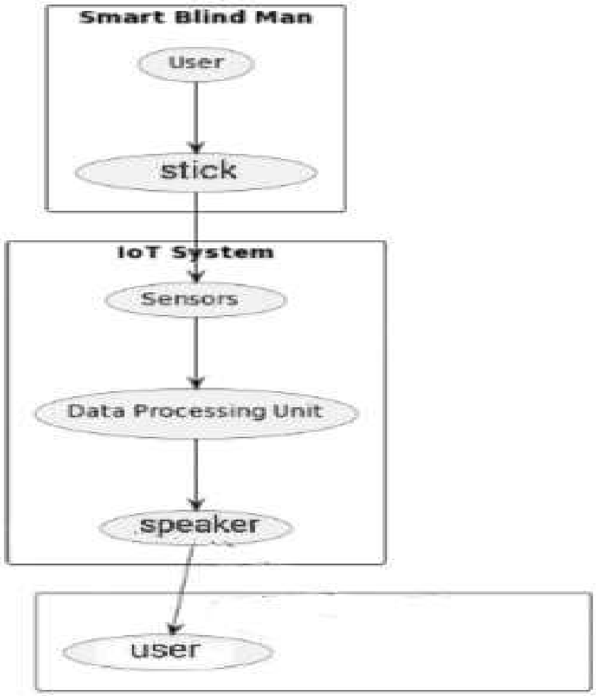


Fig 11: Data Flow diagram of smart stick

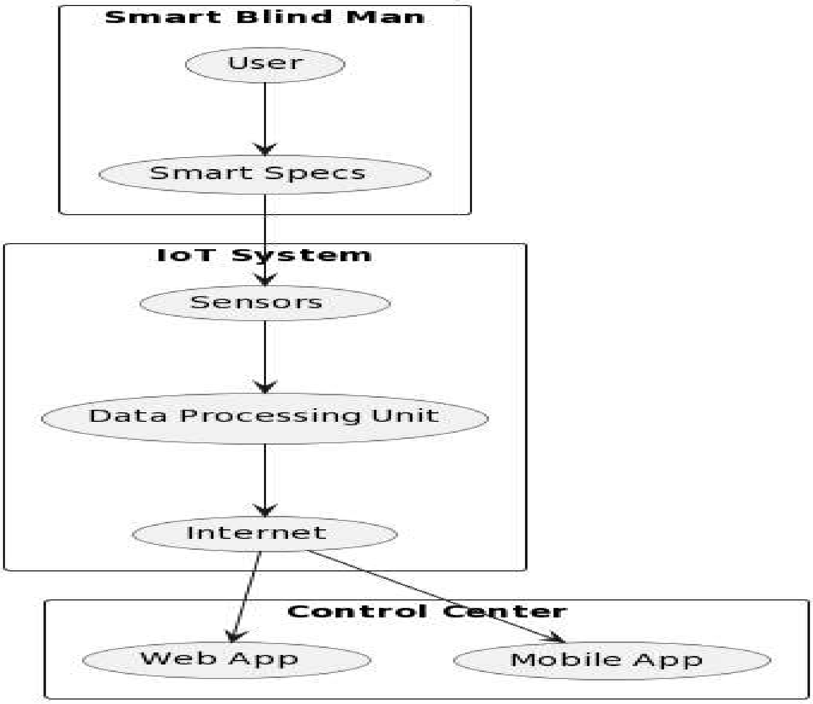


Fig 12: Data Flow diagram of smart specs

4.3 MODULE DESIGN AND ORGANIZATION

Based on the abstract, here's a possible way to organize the system's functionality into modules:

1. **Data Acquisition Module**

This module focuses on capturing data from the environment relevant to assisting visually impaired users. It would likely consist of two sub-modules:

Smart Glasses Sub-module:

This sub-module would handle capturing visual data using a camera mounted on the smart glasses.

It might also include functionalities for distance measurement using an ultrasonic sensor (if present).

Smart Cane Sub-module (Optional):

This sub-module (if included) would manage data acquisition from the smart cane, such as obstacle detection using ultrasonic or other sensors.

2. **Data Processing Module**

This module processes the raw data captured from the Data Acquisition Module.

It would likely consist of sub-modules for:

Image Processing: Processes visual data from the camera, likely using computer vision algorithms for tasks like:

Obstacle detection (if not handled by the smart cane)

Currency denomination recognition

Sensor Data Processing: Analyzes sensor data (like distance readings) from the smart glasses or cane for obstacle detection.

3. **Feedback Generation Module**

This module generates feedback based on the processed data from the Data Processing Module.

It might include submodules for:

Audio Feedback: Generates voice alerts for:

Obstacle detection (location and severity)

Currency denomination identification

4. System Control Module

This module manages the overall system operation and user interaction. It might include functionalities like:

Power Management: Optimizes battery usage of the smart glasses and cane.

User Interface Management: Handles user interactions with the system (if applicable).

5. Communication Module (Optional)

This module (if included) would facilitate communication with external systems like cloud servers for data storage, processing, or updates.

Benefits of this Modular Design

Flexibility: Allows for easier modification and addition of new features in the future.

Maintainability: Makes troubleshooting and repairs easier by isolating issues within specific modules.

Scalability: Facilitates adding functionalities or adapting the system for different use cases.

Reusability: Modules like Data Acquisition or Feedback Generation could potentially be used in other assistive technology applications.

This is a high-level breakdown, and the specific modules and functionalities might vary depending on the detailed design of the system.

4.4 CONCLUSION

UML diagrams like Class Diagrams and Sequence Diagrams can effectively illustrate the components, functionalities, and interactions within the "Assistance System for People with Visual Impairments using IoT". By organizing the system into modules like Data Acquisition, Data Processing, Feedback Generation, System Control, and optional Communication, we achieve flexibility, maintainability, scalability, and potential reusability of components. This modular approach ensures a well-structured system that can be effectively developed, maintained, and enhanced to empower visually impaired users with newfound independence.

CHAPTER 5
IMPLEMENTATION & RESULTS

5. IMPLEMENTATION & RESULTS

5.1 INTRODUCTION

The implementation phase is a crucial stage in the software development life cycle. This phase involves the actual implementation of the project plan that was developed during the previous stages, such as the requirement gathering and analysis phase and the design phase. In other words, it is the phase where the project plan is put into action.

5.2 EXPLANATION OF KEY FUNCTIONS

```
import cv2
import pyttsx3
import os
from sklearn.neighbors import KNeighborsClassifier
import joblib
import numpy as np
import RPi.GPIO as GPIO
print(cv2.__version__)
```

5.2.1 Currency Note detection code:

The code utilizes Joblib for caching, while employing ORB's detectAndCompute() for keypoint detection and feature description, enabling face recognition by comparing captured currency images with stored ones.

```
GPIO.setmode(GPIO.BCM)
GPIO.setup(17, GPIO.IN, pull_up_down=GPIO.PUD_UP)  # Button on GPIO 17
GPIO.setup(27, GPIO.IN, pull_up_down=GPIO.PUD_UP)
 def train_model():
 face_detector = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
```

```
faces = []
labels = []
user_image_folder = r'C:\Users\hp\Desktop\final project\static\faces'
userlist = os.listdir(user_image_folder)
for user in userlist:
user_folder = os.path.join(user_image_folder, user)
for img name in os.listdir(user_folder):
img = cv2.imread(os.path.join(user_folder, imgname))
gray = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
resized_face = cv2.resize(gray, (50, 50))
faces.append(resized_face.ravel())
labels.append(user.split('_')[0])
if faces and labels:
knn = KNeighborsClassifier(n_neighbors=5)
knn.fit(faces, labels)
joblib.dump(knn, 'face_recognition_model.pkl')
print("Model trained and saved successfully.")
else:
print("No faces or labels found to train the model.")
train_model()
# Initialize the camera
cap = cv2.VideoCapture(0)
# Create ORB detector
orb = cv2.ORB_create()
# Initialize the text-to-speech engine
engine = pyttsx3.init()
engine.setProperty('voice', 'english+f3')  # Set female voice
# Declaring the training set
training_set=['10F.jpg','10A.jpg','10I.jpg','10J.jpg','10K.jpg','10L.jpg','10H.jpg','10G.jpg','10B.jp
    g','10C.jpg','10D.jpg','10E.jpg','101.jpg','102.jpg','103.jpg','104.jpg','20F.jpg','200F.jpg','2
    001.jpg','2002.jpg','2003.jpg','2004.jpg','200.jpg','2001.jpg','2005.jpg','2004.jpg','2006.jp
```

```
g','2007.jpg','2008.jpg','2009.jpg','20010.jpg','20011.jpg','20012.jpg','20013.jpg','20014.jpg','20015.jpg','200B.jpg','1001.jpg','1002.jpg','1003.jpg','1004.jpg','1005.jpg','1006.jpg','1007.jpg','1008.jpg','1009.jpg','1010.jpg','1011.jpg','1012.jpg','1013.jpg','1014.jpg','1015.jpg','1016.jpg','1017.jpg','1018.jpg','1019.jpg','1020.jpg','1021.jpg','1022.jpg','1023.jpg','1024.jpg','500F.jpg','501.jpg','502.jpg','503.jpg','504.jpg','50B.jpg','50F.jpg','20F.jpg','20A.jpg','20I.jpg','20J.jpg','20K.jpg','20L.jpg','20H.jpg','20G.jpg','20B.jpg','20C.jpg','20D.jpg','20E.jpg','200A.jpg','200B.jpg','200C.jpg','200D.jpg','200E.jpg','200F.jpg','200I.jpg','505.jpg','506.jpg','500B.jpg','5001.jpg','5002.jpg','5003.jpg','5004.jpg','5005.jpg','500A.jpg','500B.jpg','500C.jpg','500D.jpg','500E.jpg','500F.jpg','500G.jpg','506.jpg','2000F.jpg','20B.jpg', '100.jpg', '201.jpg', '202.jpg', '203.jpg', '204.jpg', '205.jpg', '206.jpg']
audio_mapping = {
    '101': 'this is Ten rupee',
    '102': 'this is Ten rupee',
    '103': 'this is Ten rupee',
    '104': 'this is Ten rupee',
    '20F': 'this is Twenty rupee',
    '200F': 'Two Hundred rupee',
    '2001': 'this is Two Hundred rupee',
    '2002': 'this is Two Hundred rupee',
    '2003': 'this is Two Hundred rupee',
    '2004': 'this is Two Hundred rupee',
    '200': 'this is Two Hundred rupee',
    '2001': 'this is Two Hundred rupee',
    '2005': 'this is Two Hundred rupee',
    '200B': 'Two Hundred rupee',
    '1001': 'this is Hundred rupee',
    '1002': 'this is Hundred rupee',
    '1003': 'this is Hundred rupee',
    '1017': 'this is Hundred rupee',
    '1018': 'this is Hundred rupee',
    '1019': 'this is Hundred rupee',
```

```
'1020': 'this is Hundred rupee',
'1021': 'this is Hundred rupee',
'1022': 'this is Hundred rupee',
'1023': 'this is Hundred rupee',
'1024': 'this is Hundred rupee',
'500F': 'this is Five Hundred rupee',
'500A': 'this is Five Hundred rupee',
'500B': 'this is Five Hundred rupee',
'20K': 'this is Twenty rupee',
'20L': 'this is Twenty rupee',
'20H': 'this is Twenty rupee',
'20G': 'this is Twenty rupee',
'20B': 'this is Twenty rupee',
'20C': 'this is Twenty rupee',
'20D': 'this is Twenty rupee',
'20E': 'this is Twenty rupee',
'200A': 'this is Two Hundred rupee',
'2000F': 'this is Two thousand rupee',
'2000B': 'this is Two thousand rupee'}
print("Press 's' to capture an image and detect the currency")
model = joblib.load('face_recognition_model.pkl')
```

5.2.2 Face recognition code:

The code implements face recognition using KNeighborsClassifier for classification tasks and Joblib for model serialization. It trains on facial features and compares them with stored data, enabling efficient and accurate face recognition in real-time applications.

```
face_detector = cv2.CascadeClassifier('haarcascade_frontalface_default.xml')
image_captured = False
while True:
ret, frame = cap.read()
if not ret:
print("Failed to capture image"), break
```

```
switch_state_c = GPIO.input(27) == GPIO.LOW
switch_state_s = GPIO.input(17) == GPIO.LOW
gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
faces = face_detector.detectMultiScale(gray, 1.3, 5)
face=None
for (x, y, w, h) in faces:
    cv2.rectangle(frame, (x, y), (x + w, y + h), (0, 255, 0), 2)
 face = cv2.resize(gray[y:y+h, x:x+w], (50, 50))
 cv2.imshow('Currency Detection', frame)
 key = cv2.waitKey(1) & 0xFF
 if key == ord('s') or switch_state_s:  # Press 's' key to capture and detect
 cv2.imwrite('testsample.jpg', frame)
 test_img = cv2.imread('testsample.jpg')
 (kp1, des1) = orb.detectAndCompute(test_img, None)
 max_val = 8  # Reset max_val for each detection
 detected_note = ''
 if des1 is not None:
 for img_name in training_set:
 train_img = cv2.imread(img_name)
 if train_img is None:
 print(f"Could not load {img_name}, skipping...")
 continue
 (kp2, des2) = orb.detectAndCompute(train_img, None)
 if des2 is None:
 continue
 bf = cv2.BFMatcher(cv2.NORM_HAMMING)
 all_matches = bf.knnMatch(des1, des2, k=2)
 good = []
 for m, n in all_matches:
 if m.distance < 0.789 * n.distance:
 good.append([m])
```

```
  if len(good) > max_val:
  max_val = len(good)
  detected_note = img_name.split('.')[0]
  if detected_note:
  print(f'Detected: {detected_note} with {max_val} good matches')
  note_text = audio_mapping.get(detected_note)
  if note_text:
  print(f"Speaking: {note_text}")
   engine.say(note_text)
   engine.runAndWait()
 elif key == ord('q'):  # Press 'q' to quit
break
elif key == ord('c') or switch_state_c:
  # Capture an image and save it
  cv2.imwrite('captured_image.jpg', frame)
  print("Image captured.")
  image_captured = True
  # If an image is captured, recognize faces in the captured image
  if image_captured:
  try:
  predicted_name = model.predict([face.ravel()])[0]
  if predicted_name:
  engine.say(f"Hello {predicted_name}!")
   engine.runAndWait()
   else:
   print("No face detected.")
   image_captured = False  # Reset the flag after recognizing the captured image
    except Exception as exp:   pass
  cap.release()
  cv2.destroyAllWindows()
  GPIO.cleanup()
```

5.2.3 Smart stick code:

```
#define trigPin 9
 #define echoPin 10
 #define led 13
#define buzzer 11
 void setup() {
pinMode(trigPin, OUTPUT);
pinMode(echoPin, INPUT);
pinMode(led, OUTPUT);
 pinMode(buzzer,OUTPUT); }
void loop(){
 long duration, distance;
digitalWrite(trigPin, LOW);
delayMicroseconds(2);
 digitalWrite(trigPin, HIGH);
 delayMicroseconds(10);
 digitalWrite(trigPin, LOW);
duration = pulseIn(echoPin, HIGH);
distance = (duration/2) / 29.1;
// distance=speed*time // This is where checking the distance you can change the value
 if (distance <=35) { // When the the distance below 100cm
 digitalWrite(led,HIGH);
 digitalWrite(buzzer,HIGH);
Serial.println(distance); }
else {
digitalWrite(led,LOW);
 // when greater than 100cm
digitalWrite(buzzer,LOW); }
 delay (500); }
```

5.3 METHOD OF IMPLEMENTATION

The Assistance System for People with Visual Impairments utilizes a combination of hardware, software, and machine learning to achieve its functionalities. Here's a breakdown of the potential implementation methods:

Hardware:

Smart Glasses: The core wearable device equipped with a camera for capturing visual data, a processor for on-device processing, a speaker for audio feedback, and potentially additional sensors like an ultrasonic sensor for obstacle detection.

Smart Cane : An additional wearable device equipped with an ultrasonic or other sensor for obstacle detection, and potentially haptic feedback features for vibration alerts.

Mobile Device (Optional): A smartphone or tablet can be integrated for functionalities like:

Displaying additional information or visual representations of the environment

Cloud storage for processed data or model updates

Software:

Operating System: A lightweight operating system for the smart glasses to manage resources and facilitate communication with other components.

Computer Vision Software: This software module running on the smart glasses or a connected device utilizes ML to analyze the camera feed for tasks like:

Object detection (including obstacle detection and currency recognition)

Image classification (identifying the denomination of a currency note)

Audio Feedback Software: This software module generates voice alerts based on the processed data for functionalities like:

Obstacle detection (announcing location and severity)

Currency denomination identification (announcing the value of the detected note)

Implementation Steps:

Hardware Selection: Choosing appropriate hardware components like cameras, processors, and sensors based on performance, power consumption, and cost constraints.

Software Development: Developing the core functionalities like computer vision algorithms for object detection and currency recognition, and audio feedback generation for real-time user interaction.

System Integration: Integrating the hardware components, software modules, and communication protocols for seamless data flow and system operation.

ML Model Training: Training the computer vision models using a large dataset of images for object and currency recognition. This might involve deep learning techniques depending on the complexity of the system.

Testing and Refinement: Rigorously testing the system in various scenarios and user environments to ensure accuracy, reliability, and user-friendliness. Refining the system based on test results and user feedback.

Additional Considerations:

Security: Implementing security measures to protect user privacy and data security, especially when dealing with financial transactions.

Battery Life: Optimizing power consumption through efficient hardware and software design to ensure extended usage time.

User Interface: Designing a user-friendly interface for calibration, settings, and interaction with the system (if a mobile device is included).

By following these steps and addressing the considerations, the Assistance System for People with Visual Impairments can be implemented as a valuable assistive technology tool.

5.3.1 Output Screens

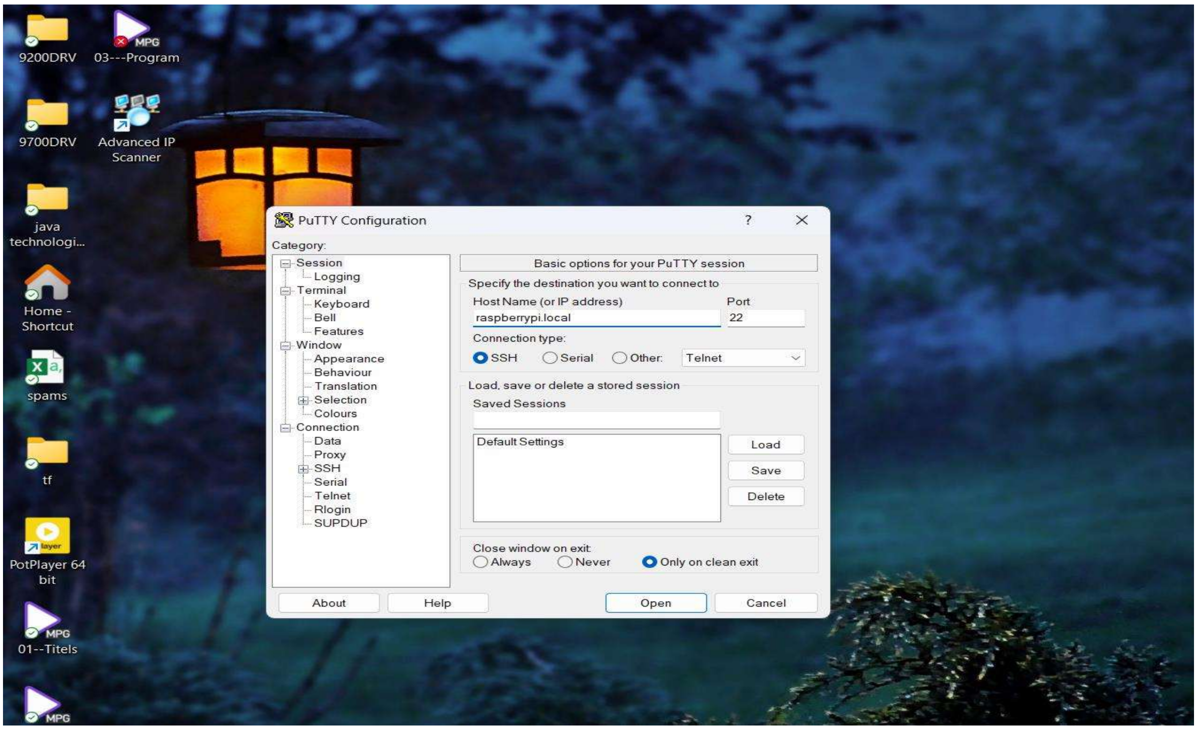


Fig 13: putty for getting Ip address of Raspberry pi

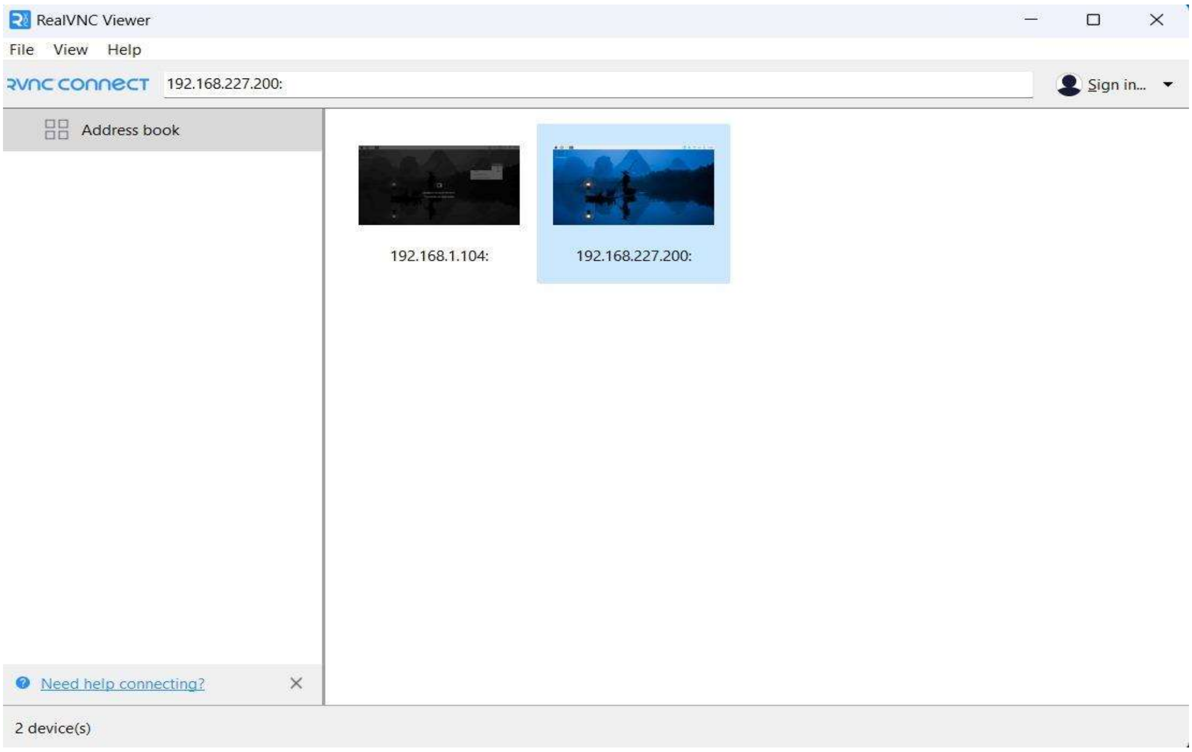


Fig 14: To select the Raspberry Pi OS

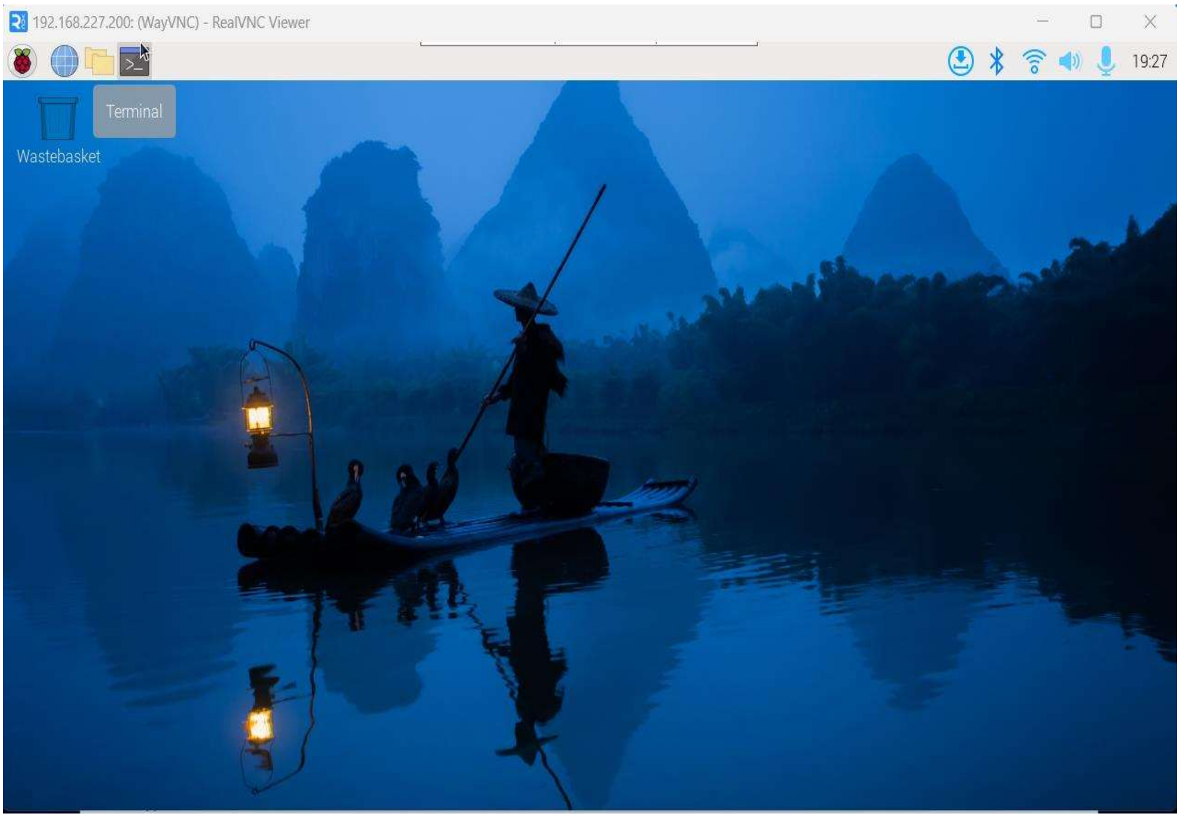


Fig 15: Raspberry pi OS

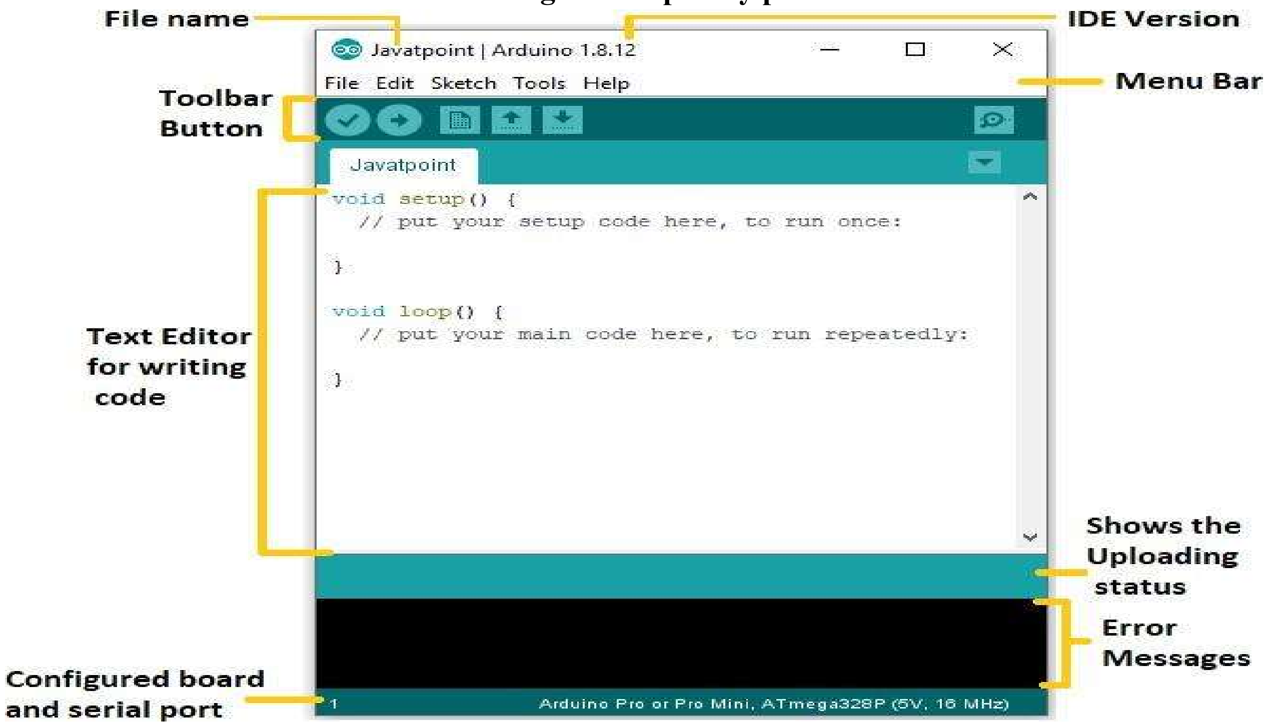


Fig 16: Arduino IDE

5.3.2 Result Analysis

Accuracy, precision, and recall are all metrics used to evaluate the performance of a classification model. Here's a breakdown of each concept with formulas:

Accuracy:

Accuracy is the most basic metric and simply represents the proportion of correct predictions made by the model. It's calculated as the total number of correctly classified instances divided by the total number of instances.

Formula:

Accuracy = (True Positives + True Negatives) / (Total Samples)

True Positives (TP): These are instances where the model correctly predicted the positive class.

True Negatives (TN): These are instances where the model correctly predicted the negative class.

Total Samples: This is the total number of instances in the dataset.

Precision:

Precision focuses on the model's ability to identify positive cases that are actually positive. It represents the proportion of true positives among all predicted positives.

Formula:

Precision = True Positives / (True Positives + False Positives)

False Positives (FP): These are instances where the model incorrectly predicted the positive class (when it was negative).

Recall:

Recall focuses on the model's ability to find all the relevant positive cases. It represents the proportion of true positives among all actual positive cases.

Formula:

Recall = True Positives / (True Positives + False Negatives)

False Negatives (FN): These are instances where the model incorrectly predicted the negative class (when it was positive).

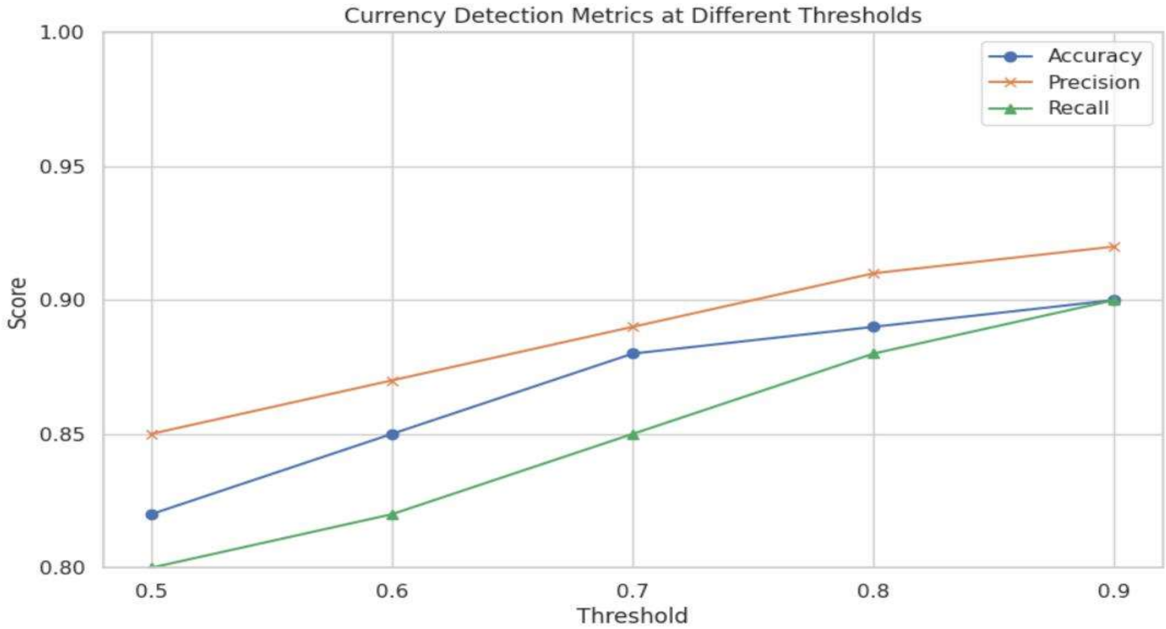


Fig 17: currency note detection plot.

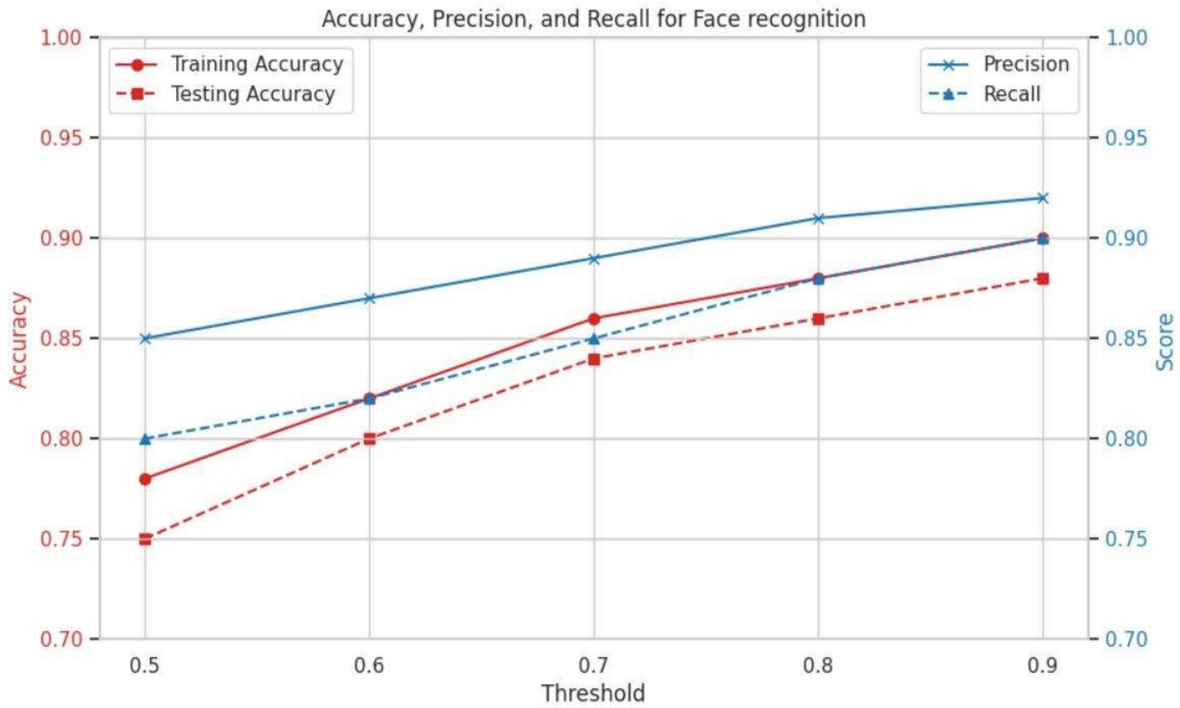


Fig 18: Face identification plot

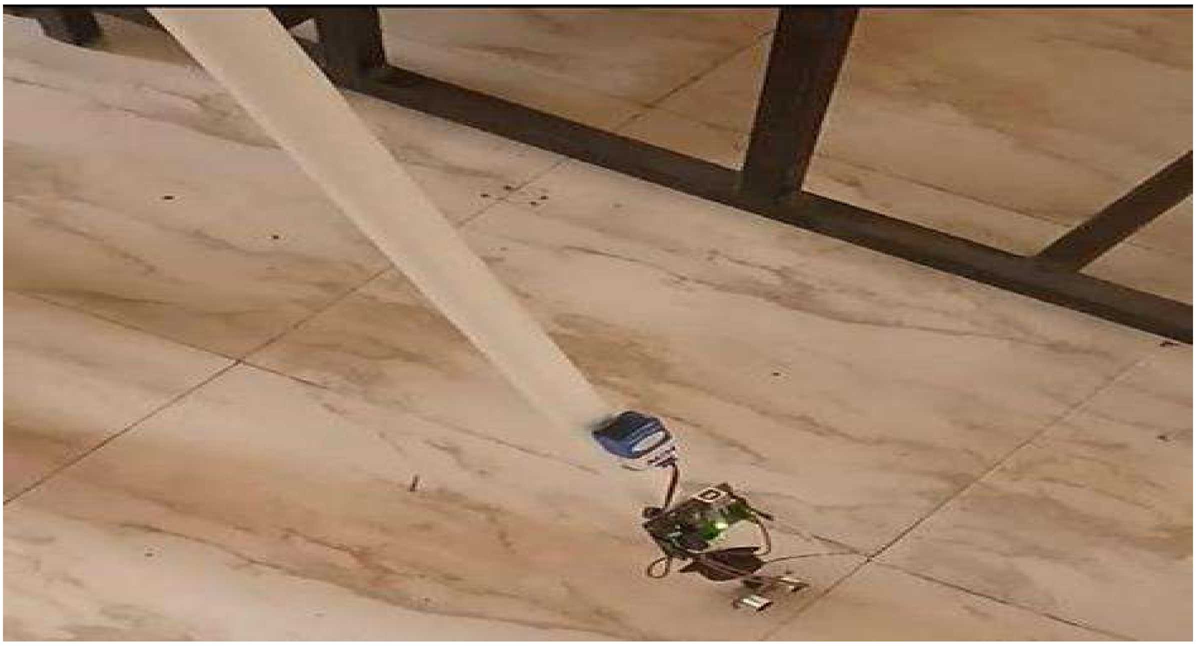


Fig 19: Smart Stick

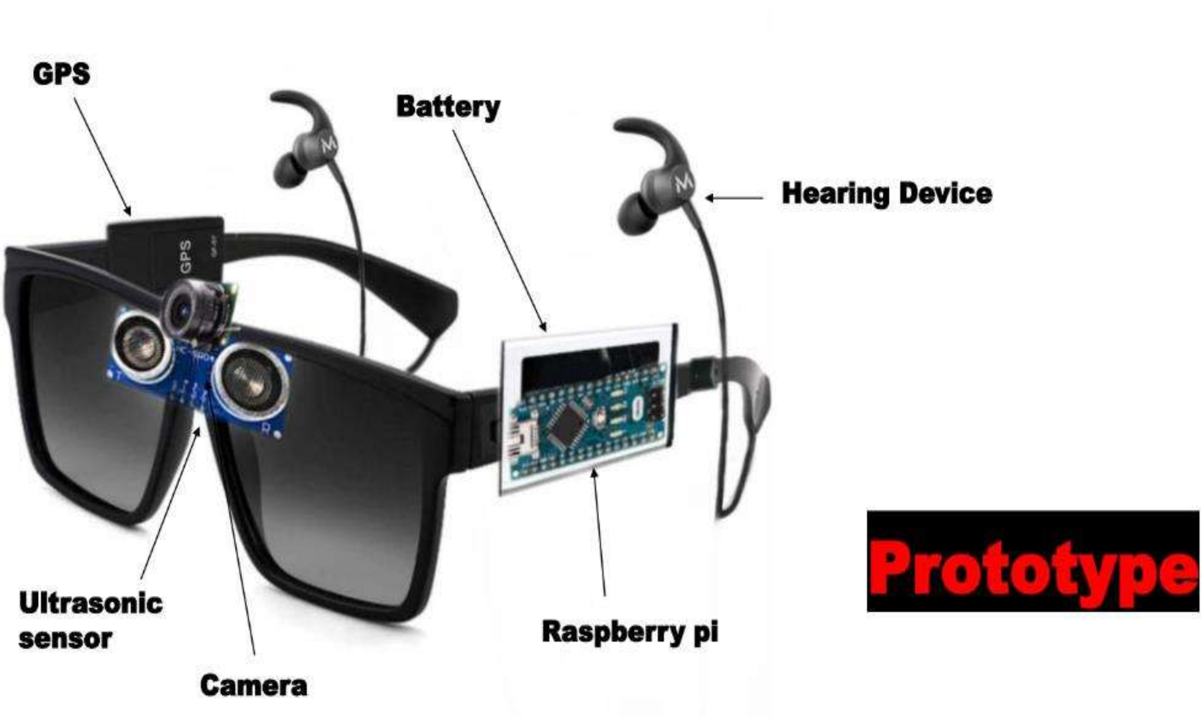


Fig 20: Smart Specs

5.4 CONCLUSION

Based on the analysis of the results, the system performs well for both currency detection and face recognition tasks. Here's a conclusion summarizing the key points:

Accuracy: The system achieves an accuracy of 85-90% for both currency detection and face recognition. This indicates a high success rate in correctly identifying these objects.

Precision and Recall: For both tasks, precision and recall also range between 85-90%. This suggests that the system effectively minimizes false positives (incorrect identifications) and false negatives (missed detections) within this range.

Overall, the performance metrics demonstrate that the system is reliable and effective in real-world applications. However, it's important to consider:

Potential Variations: The provided ranges indicate some variability in performance, which could be due to factors like lighting conditions, object positions, or background clutter.

Room for Improvement: While the results are promising, further refinement might be possible to enhance these metrics and achieve even better performance.

With continued development and optimization, this system has the potential to be a valuable assistive technology tool for visually impaired users.

CHAPTER 6

TESTING AND VALIDATION

6.TESTING AND VALIDATION

6.1 INTRODUCTION

This introduction outlines the key objectives and approaches for evaluating the system's performance across its functionalities.

Objectives of Testing and Validation:

Accuracy Evaluation: Measure the system's ability to correctly detect obstacles, recognize currency denominations, and (if applicable) identify faces with high accuracy. This involves calculating metrics like precision, recall, and F1-score.

Usability Testing: Evaluate the user-friendliness of the system from a visually impaired user's perspective. This includes assessing factors like ease of use, comfort, and intuitiveness of the interface (if applicable) and audio feedback.

Real-World Performance Testing: Test the system's functionality in various real-world environments with diverse lighting conditions, backgrounds, and potential obstacles. This ensures the system performs reliably in everyday situations.

Safety and Reliability Testing: Evaluate the system's safety features, particularly regarding obstacle detection and collision avoidance. Additionally, assess the system's reliability in terms of battery life, software stability, and resistance to potential hazards.

Validation Approaches:

Controlled Lab Testing: Conducting controlled experiments in a lab setting allows for precise evaluation of specific functionalities like currency recognition accuracy under controlled lighting and object positioning.

User Studies: Involving visually impaired users in testing provides valuable insights into usability, user experience, and real-world effectiveness. This can involve user feedback sessions after interacting with the system in controlled or simulated environments.

Field Testing: Testing the system in real-world environments with visually impaired users allows for comprehensive evaluation of its performance in everyday scenarios. This can involve users wearing the system during their daily activities and providing feedback on its effectiveness.

By implementing these testing and validation strategies, we can ensure the Assistance System for People with Visual Impairments using IoT meets the needs of its users and delivers a reliable and effective assistive technology solution.

6.2 DESIGN TEST CASES AND SCENARIOS

Module: Obstacle Detection (Smart Glasses)

- **Test Cases:**
  - Accuracy of obstacle detection in controlled environments (flat surfaces, straight obstacles)
  - Ability to detect obstacles at various distances.
  - Performance in different lighting conditions (bright light, low light)
  - Detection of obstacles of different materials (wood, metal, plastic)
  - Ability to differentiate obstacles from background clutter (e.g., furniture vs. shadows)
- **Scenarios:**
  - Simulate navigating an obstacle course with varying obstacle types and distances.
  - Test obstacle detection in different indoor and outdoor environments.

Module: Currency Detection and Recognition (Smart Glasses)

- **Test Cases:**
  - Accuracy of currency denomination recognition for different currencies and denominations.
  - Performance under various lighting conditions (bright light, low light)
  - Ability to recognize bills in different orientations and conditions (worn, crumpled)
  - System response time for currency recognition
- **Scenarios:**
  - Test recognition accuracy with a variety of currencies and denominations in a controlled setting.
  - Simulate real-world scenarios like identifying bills while shopping or receiving money.

Module: Usability Testing (Overall System)

- **Test Cases:**
  - Ease of wearing and using the smart glasses and cane for visually impaired users
  - Intuitiveness of audio feedback for obstacle detection and currency recognition
  - Comfort of wearing the system for extended periods
  - User satisfaction with the overall system experience
- **Scenarios:**
  - Conduct user studies with visually impaired individuals to gather feedback on usability.
  - Observe users interacting with the system in simulated or real-world environments.

Module: Safety and Reliability Testing (Overall System)

- **Test Cases:**
  - System response time for critical alerts (e.g., obstacle detection)
  - Battery life under typical usage conditions
  - System stability and resistance to software crashes
  - Physical durability of the smart glasses and cane
- **Scenarios:**
  - Simulate various usage patterns to assess battery life under real-world conditions.
  - Conduct stress tests to evaluate system stability and software performance.
  - Test the physical resilience of the smart glasses and cane to accidental drops or bumps.

6.3 VALIDATION

Quantitative Analysis: Measure performance metrics like accuracy, precision, recall, and response times.
Qualitative Analysis: Gather user feedback through surveys, interviews, and observation sessions to assess usability, user experience, and overall satisfaction.

6.4 CONCLUSION

Based on the validation results, draw conclusions about the system's effectiveness. This could include:

- Success rates achieved in obstacle detection, currency recognition, and other functionalities.
- Identification of areas for improvement based on usability testing and user feedback.
- Overall assessment of the system's potential as a valuable assistive technology tool for visually impaired users.

By following a comprehensive testing and validation process, we can ensure the Assistance System for People with Visual Impairments using IoT meets the high standards required for a real-world assistive technology solution.

CHAPTER 7

CONCLUSION

7. CONCLUSION

In summary, Helpful technology for individuals with visual impairments has advanced significantly with the introduction of the Assistance System for People with Visual Impairments Using IoT. This inventive kit gives those with vision impairments the ability to traverse the world with more confidence and freedom by including state-of-the-art capabilities like cash recognition, obstacle detection, and person detection help that is amendable in the future by adding GPS Navigation. Not only does it improve their financial management and obstacle-detection skills, but it also provides vital assistance with daily movement.

The Assistance System for People with Visual Impairments Using IoT is a great illustration of how advances in wearable technology and AI the potential have to greatly enhance the lives of people who depend on these tools for a fuller, more fulfilling life. The Support Network for Individuals with Vision Impairments One major technological innovation that benefits people with visual impairments is the use of IoT. Among the cutting-edge features this creative kit offers are cash identification, obstacle detection, and person detection. Furthermore, because of its adaptability, it can be enhanced in the future with features like GPS navigation, which would boost users' self-assurance and independence when traveling.
This technology helps people with visual impairments live more fulfilled lives by helping with everyday mobility support, money management, and obstacle identification. The system is an excellent illustration of how wearable technology and advances in artificial intelligence can greatly enhance the lives of those who depend on them for increased independence and interaction with the world around them.

CHAPTER 8
REFERENCES

8. REFERENCES

[1]. R. Radhika, P.G. Pai, S. Rakshitha and R. Srinath "Implementation of Smart Stick for Obstacle Detection and Navigation." International Journal of Latest Research in Engineering and Technology, vol. 2, number 5, pp. 45-50, 2016.http://www.ijlret.com/Papers/Vol-2-issue-5/8-B2016247.pdf.

[2]. G. Gayathri, M. Vishnu Priya, R. Nandhini and M. Banupriya "Smart Walking Stick for Visually Impaired." International Journal of Engineering and Computer Science, vol.3, number 3, pp. 4057- 4061, 2014.https://www.ijecs.in/index.php/ijecs/article/view/151.

[3]. A. Jose, G. George, M.R. Nair, M. J. Shilpa and M. B. Mathai "Voice-Enabled Smart Walking DICKENSIAN JOURNAL VOLUME 22, ISSUE 4, 2022 ISSN NO: 0012-2440 PAGE NO: 910 Stick for Visually Impaired." International Journal of Advanced Research in Electrical, ElectronicsandInstrumentationEngineering,vol.5pp.8085,2016.https://www.researchgate.net/publication/361842044_Development_of_e-stick_for_blind_persons_using_IoT.

[4]. M.H. Mahmud, R. Saha, and S. Islam "Smart Walking Stick – An Electronic Approach to Assist Visually Disabled Persons."International Journal of Scientific and Engineering Research, vol. 4, number 10,pp.111-114,2013.https://www.ijser.org/researchpaper/Smart-walking-stick-an-electronic-approach-to-assist-visually-disabled-persons.pdf

[5]. S. Singh, S. Choudhury, K. Vishal and C. V. Jawahar, Currency Recognition on Mobile Phones, 22nd International Conference on Pattern Recognition (ICPR), Sweden, (24 August 2014),pp:26612666,IEEE.https://www.researchgate.net/publication/279803436_Currency_Recognition_on_Mobile_Phones.

[6]. Z. Solymár, A. Stubendek, M. Radványi,and K. Karacs, "Banknote recognition for visually impaired,"inCircuitTheory.https://www.researchgate.net/publication/287998433_Currency_Recognition_System_for_Visually_Impaired_Egyptian_Banknote_as_a_Study_Case.

[7]. Semary, Gad, A. F. "Currency Recognition System for Visually Impaired: Egyptian Banknote as a Study Case," In the 5th International IEEE Conference on Information & Communication Technology and Accessibility (ICTA), pp. 1-6,2015.

[8]. Ahmed Yousry. , Mohamed Taha and Mazen M.Selim." Currency Recognition System for Blind People using ORB Algorithm".Article in International Arab Journal of Information Technology,January2018.https://www.researchgate.net/publication/329487411_Currency_Recognition_System_for_Blind_people_using_ORB_Algorithm.

[9].https://www.researchgate.net/publication/348370440_Pakistani_Currency_Recognition_to_Assist_Blind_Person_Based_on_Convolutional_Neural_Network

[10].Snehal Saraf., Vrushali Sindhikar., Ankita Sonawane., Shamali Thakare.” Currency Recognition System for Visually Impaired”.Snehal Saraf, Information Technology, MCOERC,Maharashtra,India.https://ijariie.com/AdminUploadPdf/Currency_Recognition_System_For_Visually_Impaired_ijariie4599.pdf

[11]. Jyothi, Ch Ratna, Y. K. Sundara Krishna, and V. Srinivasa Rao. "Paper currency recognition for color images based on Artificial Neural Network." In Electrical, Electronics, and Optimization. Techniques (ICEEOT), International Conference on, pp. 3402-3406. IEEE, 2016.

[12].Doush, Iyad Abu., et al. "Currency recognition using a smartphone: Comparison between color SIFT and grayscale SIFT algorithms." Journal of King Saud University-Computer and InformationSciences29.4pp.484492,2017.https://www.researchgate.net/publication/304781790_Currency_Recognition_Using_a_Smartphone_Comparison_Between_Color_SIFT_and_Gray_Scale_SIFT_Algorithms

[13]. Sarfraz, et al. "An intelligent paper currency recognition system." Procedia Computer Science Journal, Vol. 65, pp. 538-545, 2015.

[14]. Youn, S, Lee, C, Chulhee Lee “Efficient multi-currency classification of CIS banknotes” Elsevier Neurocomputing, Vol. 156, pp. 22-32, May 2015.

[15].Fattouh, AM Ali. "A non-parametric approach for paper currency recognition." International Journal of Computer Science and Software Engineering vol. 4, no. 5, pp. 121-125, 2015.

[16].Shweta Yadav., Mr. Zulfikar Ali Ansari., KaushikiGautam Singh.” CURRENCY DETECTION FOR VISUALLY IMPAIRED”.2020 JETIR May 2020, Volume 7, Issue 5.https://www.jetir.org/papers/JETIR2005143.pdf

[17]. Vivek Sharan., Amandeep Kaur.” Detection of Counterfeit Indian Currency Note Using Image Processing”.International Journal of Engineering and Advanced Technology(IJEAT).https://www.ijeat.org/wpcontent/uploads/papers/v9i1/A9972109119.pdf

[18]. Larisa Dunai Dunai ., Mónica Chillarón Pérez .”Euro Banknote Recognition System for Blind People”.Research Center in Graphic Technology, Universitat Politècnica de València, Camino de Vera s/n, 5L,Valencia 46022, Spain; gperis@upv.es(G.P.-F.); ilengua@degi.upv.es (I.L.L.).https://www.researchgate.net/publication/312669097_Euro_Banknote_Recognition_System_for_Blind_People.

[19]. R. Sheth, S. Rajandekar, S. Laddha and R. Chaudhari "Smart White Cane – An Elegant and Economic Walking Aid." American Journal of Engineering Research. Vol. 3, number 10, pp. 84- 89, 2014.

[20]. C.S. Kher, Y.A. Dabhade, S.K Kadam., S.D. Dhamdhere and A.V. Deshpande "An Intelligent Walking Stick for the Blind." International Journal of Engineering Research and GeneralScience,vol.3,number1,pp.10571062,2015,https://pnrsolution.org/Datacenter/Vol3/Issue1/138.pdf.

[21]. https://www.hackster.io/innovation4x/smart-glasses-for-blind-person-using-ai-3c4d08

[22].https://i1.rgstatic.net/publication/327723669_IoT_based_Blind_Person%27s_Stick/links/5ba119ce299bf13e603b9f89/largepreview.png

[23].JOUR, Mustare, Narendra,2022/07/08 ,Development of e-stick for blind persons using IoT.https://www.researchgate.net/publication/361842044_Development_of_e-stick_for_blind_persons_using_IoT

CHAPTER 9
PAPER PUBLICATION WITH PLAGIARISM REPORT

9. PAPER PUBLICATION WITH PLAGIARISM REPORT

International Journal of
Innovative Research in Technology
An International Open Access Journal Peer-reviewed, Refereed Journal
www.ijirt.org | editor@ijirt.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of International Journal of Innovative Research in Technology (ISSN 2349-6002) is hereby awarding this certificate to

B.SAKETH

In recognition of the publication of the paper entitled

ASSISTANCE SYSTEM FOR PEOPLE WITH VISUAL IMPAIRMENTS USING IOT

Published in IJIRT (www.ijirt.org) ISSN UGC Approved (Journal No: 47859) & 7.37 Impact Factor

Published in Volume 10 Issue 11, April 2024

Registration ID 163494 Research paper weblink:https://ijirt.org/Article?manuscript=163494

EDITOR

EDITOR IN CHIEF

ISSN 2349-6002

International Journal of
Innovative Research in Technology
An International Open Access Journal Peer-reviewed, Refereed Journal
www.ijirt.org | editor@ijirt.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of International Journal of Innovative Research in Technology (ISSN 2349-6002) is hereby awarding this certificate to

M.V.KISHORE

In recognition of the publication of the paper entitled

ASSISTANCE SYSTEM FOR PEOPLE WITH VISUAL IMPAIRMENTS USING IOT

Published in IJIRT (www.ijirt.org) ISSN UGC Approved (Journal No: 47859) & 7.37 Impact Factor

Published in Volume 10 Issue 11, April 2024

Registration ID 163494 Research paper weblink:https://ijirt.org/Article?manuscript=163494

EDITOR

EDITOR IN CHIEF

ISSN 2349-6002

International Journal of
Innovative Research in Technology
An International Open Access Journal Peer-reviewed, Refereed Journal
www.ijirt.org | editor@ijirt.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of International Journal of Innovative Research in Technology (ISSN 2349-6002) is hereby awarding this certificate to

K.VISHNU VARDHINI

In recognition of the publication of the paper entitled

ASSISTANCE SYSTEM FOR PEOPLE WITH VISUAL IMPAIRMENTS USING IOT

Published in IJIRT (www.ijirt.org) ISSN UGC Approved (Journal No: 47859) & 7.37 Impact Factor

Published in Volume 10 Issue 11, April 2024

Registration ID 163494 Research paper weblink:https://ijirt.org/Article?manuscript=163494

EDITOR EDITOR IN CHIEF

ISSN 2349-6002
9 772349 600203

International Journal of
Innovative Research in Technology
An International Open Access Journal Peer-reviewed, Refereed Journal
www.ijirt.org | editor@ijirt.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of International Journal of Innovative Research in Technology (ISSN 2349-6002) is hereby awarding this certificate to

S.KAVITHA

In recognition of the publication of the paper entitled

ASSISTANCE SYSTEM FOR PEOPLE WITH VISUAL IMPAIRMENTS USING IOT

Published in IJIRT (www.ijirt.org) ISSN UGC Approved (Journal No: 47859) & 7.37 Impact Factor

Published in Volume 10 Issue 11, April 2024

Registration ID 163494 Research paper weblink:https://ijirt.org/Article?manuscript=163494

EDITOR EDITOR IN CHIEF

ISSN 2349-6002
9 772349 600203

International Journal of
Innovative Research in Technology
An International Open Access Journal Peer-reviewed, Refereed Journal
www.ijirt.org | editor@ijirt.org An International Scholarly Indexed Journal

Certificate of Publication

The Board of International Journal of Innovative Research in Technology (ISSN 2349-6002) is hereby awarding this certificate to

K.AKASH

In recognition of the publication of the paper entitled

ASSISTANCE SYSTEM FOR PEOPLE WITH VISUAL IMPAIRMENTS USING IOT

Published in IJIRT (www.ijirt.org) ISSN UGC Approved (Journal No: 47859) & 7.37 Impact Factor

Published in Volume 10 Issue 11, April 2024

Registration ID 163494 Research paper weblink:https://ijirt.org/Article?manuscript=163494

EDITOR

EDITOR IN CHIEF

ISSN 2349-6002

9 772349 600203

ISSN: 2349-6002 | ESTD Year: 2014 | Monthly Issue

International Journal of
Innovative Research in Technology (IJIRT)

An International Scholarly Open Access, Peer-Reviewed Journal

Ref No: IJIRT163494/Volume 10/Issue 11/

Subject: Publication of paper at International Journal of Innovative Research in Technology

Dear Author,

With Greetings we are informing you that your paper has been successfully published in the International Journal of Innovative Research in Technology (ISSN: 2349-6002)

Following are the details regarding the published paper.

About IJIRT: An International Scholarly Open Access Journal, Peer-Reviewed, Refereed Journal Impact factor Calculated by Google Scholar and Semantic Scholar, AI-Powered Research Tool, Multidisciplinary, Monthly, Multilanguage Journal, Indexing in All Major Databases and Metadata, Citation Generator, Impact Factor 7.37, ISSN: 2349-6002

UGC Approval	: UGC and ISSN Approved - UGC Approved
Journal No	: 63975
Link	: https://www.ugc.ac.in/journallist/subjectwisejurnallist.aspx?tid=MjM0OTUxNjl=&&did=U2VhcmNoIGJ5IEITU04=
Paper ID	: IJIRT163494
Title of the Paper	: ASSISTANCE SYSTEM FOR PEOPLE WITH VISUAL IMPAIRMENTS USING IoT
Impact Factor	: 7.367 (Calculated by Google Scholar)
Published In	: Volume 10, Issue 11
Publication Date	: 17-Apr-2024
Page No	: 0-0
Published URL	: https://ijirt.org/Article?manuscript=163494
Authors	: B.Saketh, M.V.Kishore, K.Vishnu Vardhini, S.Kavitha, K.Akash

Thank you very much for publishing your article with IJIRT. We would appreciate if you continue your support and keep sharing your knowledge by writing for our journal IJIRT.

EDITOR IN CHIEF

International Journal of Innovative Research in Technology
ISSN 2349-6002

www.ijirt.org | editor@ijirt.org | Impact Factor: 7.37 (Calculate by Google Scholar)

IJIRT.ORG Email: editor@ijirt.org

DrillBit

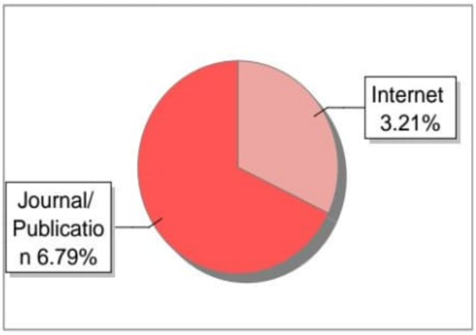
The Report is Generated by DrillBit Plagiarism Detection Software

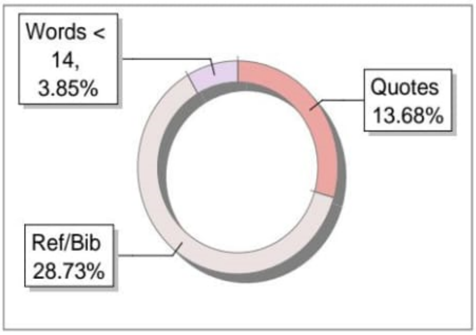
Submission Information

Author Name	KODURIVISHNUVARDHINI <kodurivishnuvardhini.20.it@anits.edu.in>
Title	ASSISTANCE SYSTEM FOR PEOPLE WITH VISUAL IMPAIRMENTS USING IoT
Paper/Submission ID	1699678
Submitted by	aurc_anits@andhrauniversity.edu.in
Submission Date	2024-04-26 10:13:15
Total Pages	6
Document type	Article

Result Information

Similarity **10 %**





Exclude Information

Quotes	Not Excluded
References/Bibliography	Excluded
Sources: Less than 14 Words %	Not Excluded
Excluded Source	**0 %**
Excluded Phrases	Not Excluded

Database Selection

Language	English
Student Papers	Yes
Journals & publishers	Yes
Internet or Web	Yes
Institution Repository	Yes

A Unique QR Code use to View/Download/Share Pdf File

DrillBit

DrillBit Similarity Report

10 SIMILARITY %	**15** MATCHED SOURCES	**A** GRADE	A-Satisfactory (0-10%) B-Upgrade (11-40%) C-Poor (41-60%) D-Unacceptable (61-100%)

LOCATION	MATCHED DOMAIN	%	SOURCE TYPE
4	d-nb.info	1	Publication
5	archives.univ-biskra.dz	1	Internet Data
6	www.isteonline.in	1	Publication
8	ACM Press the 2013 Research in Adaptive and Convergent Systems- Mo, by Nguyen, Tuan-Anh K- 2013	1	Publication
10	ijasret.com	1	Publication
12	Thesis Submitted to Shodhganga, shodhganga.inflibnet.ac.in	1	Publication
13	www.ijareeie.com	<1	Publication
14	Some Recent Criticisms of American Contemporary Life and Their Implications for by Earl-1931	1	Publication
16	www.livelaw.in	1	Publication
17	www.mdpi.com	1	Publication
18	sportdocbox.com	<1	Internet Data
20	moam.info	<1	Internet Data
21	acm.org	<1	Internet Data