Muhammad Arslan Nawaz
01-135212-050

Muhammad Younas
01-135212-073

A Smart Detection App for Visually Impaired Users with Voice Feedback

Bachelor of Science in Information Technology

Supervisor: Ms. Ameena Saeed

Department of Computer Science
Bahria University, Islamabad

May 2025

Bahria University
Islamabad Campus
(Department of Computer Science)

CERTIFICATE

We accept the work contained in the report titled "A Smart Detection App for Visually Impaired Users with Voice Feedback" as a confirmation to the required standard for the partial fulfillment the degree of BS(IT).

Supervisor	Internal Examiner	External Examiner
Name: Ameena Saeed	Name: Mr. Saeed-ur Rehman	Name: Salman
Date:	Date: 23/5/25	Date: 23/05/2025

Project Coordinator	Head of Department
Name: Dr. Kabeer Ahmed Bhatti	Name: Dr. Khurram Ehsan
Date:	Date:

Abstract

Visually impaired people often face difficulty in navigation and daily tasks, especially when it comes to identifying surrounding objects or recognizing currency denominations. A smart detection app is designed for visually impaired people to navigate their surroundings safer and independently. This solution integrates object recognition and currency detection. The core objectives of this mobile application are to detect and recognize Pakistani currency denominations as well as to detect indoor objects and provide immediate voice feedback.

The system architecture incorporates a camera module, lightweight deep learning models and a voice feedback system ensuring usability and accessibility for visually impaired users. Advanced deep learning techniques are used to ensure the safety and assistance of visually impaired people. For detection purposes, YOLOv8 model is used that generates a text output by labeling that particular object. For assistance of visually impaired users, text output is converted to voice feedback using Text-to-Speech library. The app ensures accurate and efficient identification of various denominations of indoor objects and currency. The design methodology follows an object-oriented, modular approach with UML modeling, ensuring maintainability and scalability. This application is developed with a focus on user friendliness and runs effectively on Android devices. Extensive testing demonstrates the reliability and potential of a system that significantly improves users' daily lives by providing greater autonomy and reducing their dependence on others.

A comprehensive literature review highlights related work in object detection and currency recognition, underscoring gaps in integration and localization.

This app fills these gaps by offering an all-in-one solution requiring no external hardware. The combination of Pakistani currency detection and object detection makes this mobile application extremely helpful for the visually impaired users.

Acknowledgments

To begin with, we would like to thank Almighty Allah for giving us courage in order to overcome all those challenges, which have arisen during completion of this project. The most grateful thanks go to our supervisor Ms. Ameena Saeed for her regular assistance and support in the course of the project. We are also grateful and thank our department of Computer Science at Bahria University Islamabad, for giving the required facilities and environment for doing this work. Lastly, our other faculty members, family, and friends are acknowledged for their incessant support.

MUHAMMAD ARSLAN NAWAZ
MUHAMMAD YOUNAS

Islamabad, Pakistan
May 2025

Contents

List of Figures

List of Tables

Acronyms and Abbreviations

OOD	Object-oriented Design
YOLO	You Only Look Once
TTS	Text-to-Speech
CNN	Convolutional Neural Network
DL	Deep Learning

Chapter 1

Introduction

1.1 Project Background

Visually impaired people often face many challenges in their daily lives, especially when moving around or performing tasks that sighted people do without thinking. Two of the most common and important difficulties they face are identifying objects and recognizing currency notes. These tasks may seem simple, but for someone who cannot see, they can lead to serious issues, like misidentifying money or using the wrong item.
In today's age of rapid technological growth, it's important to develop solutions that make life easier and more independent for people with disabilities. Our project focuses on creating a user-friendly mobile application specifically designed for visually impaired users in Pakistan. The app has two main goals: detecting indoor objects and recognizing Pakistani currency notes of various denominations. We use modern deep learning techniques to enable the app to recognize different objects and currency notes. After detection, the app generates a text label, which is then converted into speech using Flutter's Text-to-Speech (TTS) feature. This voice feedback helps users understand what is in front of them without needing visual assistance. What makes this project unique is its focus on localization and accessibility. The currency recognition feature is tailored to Pakistani notes, ensuring accuracy for local users. Similarly, the object detection function is designed to assist users in indoor environments like homes, offices, and classrooms, where visually impaired individuals often spend their time. By combining deep learning, mobile technology, and voice feedback, our app aims to empower visually impaired users. It helps them become more independent, improves their daily life, and gives them confidence to perform everyday tasks.

1.2 Problem Description

Although several mobile apps aim to support the visually impaired, most of them fail to meet the real needs of local users. A major issue is the lack of localization many apps don't recognize Pakistani currency, making them unhelpful for users in Pakistan who rely on accurate currency detection. Some apps only focus on detecting currency and ignore object recognition, while others promise both but don't perform well in either. Users often report that such apps take too long or do not work at all, making them unreliable and frustrating to use. Because of these limitations, many blind users lose trust in these tools. They may feel these apps are fake or unsafe, and instead of helping, the apps make them more dependent on others. This also impacts their privacy, as they need to ask someone else even for simple tasks like recognizing a currency note or identifying an object. The lack of a reliable, locally relevant, and comprehensive app creates a big gap in the lives of visually impaired people. Without such tools, their mobility, confidence, and quality of life suffer. Our project aims to fill this gap by providing a trustworthy and useful app designed with Pakistani users in mind.

1.3 Project Objectives

The main goal of this project is to develop a mobile application that helps visually impaired individuals perform daily tasks more independently. The app will support object recognition and identify Pakistani currency notes, giving users the ability to manage tasks on their own. One key objective is to train a deep learning model specifically for recognizing Pakistani currency. Since every country has unique currency designs, a local model is necessary for accurate detection. This will help users handle money confidently and avoid being misled. Another important goal is to build an object detection model that identifies commonly used indoor items. This feature will help users find household objects, locate personal items, or recognize things in places like offices and stores reducing their need to ask for help. In addition to visual recognition, the app will include voice feedback. Using text-to-speech technology, the app will give clear audio responses so users can easily understand the output. Overall, the project is about giving visually impaired people a tool that supports their independence. Whether it's recognizing a currency note while shopping or identifying items at home, the app aims to improve safety, freedom, and quality of life.

1.4 Project Scope

This project focuses on creating a mobile app that uses deep learning and voice technology to support visually impaired users. The app includes two main deep learning models: one for recognizing Pakistani currency notes and another for detecting common indoor objects. These models will work together to provide a practical and accessible experience. When a visually impaired person uses their mobile camera, the app will process the image through each model. These models will identify the object or currency and generate a text result. This result is then turned into voice feedback using text-to-speech technology. The spoken response will help users act immediately whether they're picking up an item at home or confirming a currency note's value in a market. The app is designed to make everyday tasks easier, more secure, and more independent for visually impaired individuals.

Chapter 2

Literature Review

2.1 Related Work

Much research work has been carried out to come up with assistive technology that is meant to increase the independence of visually impaired people. Most of the systems were having single functionality, they were either doing currency detection or object detection, none had them both. Besides, none of the apps was identifying Pakistan currency denominations. Most of the applications did not have the integration of currency and object models as a whole, which opened the way for improvement.

2.1.1 Object Detection Systems

A research paper, entitled "Object Detection System for visually impaired", uses new technology for smartphones to build an object detection system that gives the visually impaired people realtime auditory support in the form of voice-overs. The research employs smartphone-based object detection for giving real-time auditory feedback. Although this method is effective, it does not involve currency detection and measurement of the distance [1].

2.1.2 Currency Recognition Applications

The paper "Currency Recognition for the Visually Impaired People" discusses the process of developing the mobile application that uses the machine learning approach to recognize the Indian currency notes for the assistance of visually impaired people in the financial matters. Concentrating on identifying Indian currency and helping with transactions of money. It does not deal with object detection and distance measuring [2].

2.1.3 Integrated Assistive Devices

Research on "MagicEye: "An Intelligent Wearable Towards Independent Living of Visually Impaired" is an introduction of a wearable device which incorporates object detection, facial recognition, and currency identification, and that can help visually impaired users in daily activities. Brings the concept of comprehensive support in the form of a wearable device such as facial recognition. But it involves extra hardware that may not be accessible by all the users [3].

2.1.4 Real-Time Object Recognition

This research "Assisting Blind People Using Object Detection with Vocal Feedback" offers a real-time object detection model R based on the YOLO model that presents the audible feedback to guide blind people in their surrounding. Applies YOLO for object detection and gives auditory feedback. Although effective, it does not include currency detection/ detection of specific local currencies [4].

2.1.5 Banknote Recognition Systems

In the paper called "Banknote Recognition for Visually Impaired People (Case of Ethiopian note)" a mobile application is described, which identifies the Ethiopian banknotes correctly and gives feedback in the form of voice in local language for visually impaired users. Are about Ethiopian banknotes with a voice feedback in the local language. Limited to currency recognition [5].

2.1.6 Advancing AI-based Assistive Systems

The research "Advancing AI-based Assistive Systems for Visually Impaired People" states a modified YOLOv7 as well as YOLOv8 algorithm for multi-class object detection and currency classification that boosts the real-time detection capabilities for assistive technologies. Presents sophisticated algorithms of multi-class detection. However, it may require great measures of computation resources and hence less suitable for portable use [6].

2.1.7 Developing an Assistive Technology for Visually Impaired Persons

This study introduces an innovative assistive technology that has the capacity to empower the visually impaired consumers by availing to them the opportunity of independently identifying the Ethiopian banknotes solving problem of recognition and differentiation of the denominations of the currency. Focuses on Ethiopian currency identification. The scope is limited in comparison with your overall approach [7].

2.1.8 Critical Analysis Table

Table 2.1: Critical Analysis of Related Research Work

Author	**Features**	**Critical Analysis Perspective**
Dr. Hiren PhD	Object detection using smartphones; real-time auditory feedback	Limited to object detection, no currency detection and localization.
Abilash and Hari Vingesh	Currency Recognition using ML, focus on Indian currency, aids financial transactions.	Focuses only on currency recognition, no object detection and real-time multi-functional integration.
Sethuraman, Gaurav R, Tadkapally, Anitha Subramanyan	Magic Eye wearable device with object detection, facial recognition and currency identification.	Requires additional hardware (wearable device) making it less accessible and cost-effective for users relying on smartphones.
Hiba Najam	Real-time object detection with YOLO, vocal feedback for visually impaired users.	Effective object detection and feedback system but lacks currency recognition or localization.
Nuredin Ali AbdelKadir	Currency recognition for Ethiopian currency with voice feedback for visually impaired users.	Narrow scope; limited to Ethiopian currency only and lacks object detection.
Nasir Uddin Faysal, Thamid	Advanced multi-class detection using modified YOLOv7 and YOLOv8 algorithms.	High computational requirements, may not be optimized for lightweight and mobile implementations.
Samuel Alamirew	Assistive technology for Ethiopian currency recognition, supports independent financial transactions.	Limted to Ethiopian currency, lacks object detection.

2.1.9 Comparison Table

Table 2.2: Feature comparison with other projects

Feature	Object Detection	Currency Recognition	Localization	Voice Feedback	Additional Hardware Required
Smartphone Object detection	✓	✗	✗	✓	✗
Currency Recognition	✗	✓	✗	✓	✗
MagicEye Wearable	✓	✓	✗	✓	✓
Real-time object Recognition	✓	✗	✗	✓	✗
Ethiopian Bank Note System	✗	✓	✓	✓	✗
Advanced AI Systems	✓	✓	✗	✓	✗
Ethiopian Assistive Technology	✗	✓	✓	✓	✗
Detection App for Visually Impaired Users	✓	✓	✓	✓	✗

2.2 Identified gaps

There already exist systems that lack a system that combines both the currency and object detection in addition to a voice feedback module. These systems do not support Pakistani currency denominations and that makes them less effective to its Pakistani users. There are also problems with accuracy of their models in varying conditions of the environment including different lighting scenarios. Some of the systems require extra hardware to be used for object detection or currency recognition. Furthermore, some of the systems were doing currency and object detection where either they do not have voice feedback module or failed to recognize Pakistani currency.

Chapter 3

Requirement Specifications

3.1 Existing System

There are many of the current systems that were created to assist visually impaired people and they are either too generalized or lack the key features to be used efficiently day-to-day. These applications typically sustain a single task such as basic object detection or text reading and can't provide a holistic and all-rounded solution. In a number of these cases, the users are forced to buy extra hardware in order to support the full functionality of these systems. This makes them costly and less accessible to the users who only access the Internet through mobile devices.

Another limitation of most of the existing ones is lack of support for local needs. Most of the apps that are found in app stores are meant for international use and do not accept Pakistani currency notes. Consequently, visually impaired users in Pakistan experience difficulties when trying to access these applications at times of doing financial transactions. This leaves a significant hole in the usability and decreases the utility of such tools in local environments.

In addition, the features of the already existing applications are not always integrated properly. For instance, there can be an app that has currency recognition without the feature of object detection or voice feedback. Some offer object detection but do not convert them into audio, which is important for the persons who cannot see the screen. This scattered approach requires the users to rely on various apps to meet their needs which cause confusion and inconvenience.

3.2 Proposed System

The proposed system is to give comprehensive and useful solution to visual impaired people bringing together various necessary features into a solitary mobile application. Rather than many of the methods that are standalone, meaning, they provide either object detection or currency recognition independently, this system combines the two, using lightweight pretrained deep learning models. These models are particularly designed to detect indoor objects in the everyday settings as well as ability to recognize Pakistani currency notes with a high level of accuracy.

For the purpose of improving the user experience and usability, the system incorporates a module for voice feedback. This part transforms the detection results into meaningful audio messages, with which visually impaired users can benefit from clear and on-the-

spot guiding messages without depending on visual output. This aspect helps them to make better decisions in carrying out everyday activities like identifying things at home or determining the value of money when making purchases.
The performance of the proposed system is one of the key benefits of the system. The deep learning models are configured to perform with minimal delay meaning fast and prompt feedback. This design ensures that the system is reliable as well as effective when compared to other solutions that are plagued by either slow processing or inaccurate results most often.
Also, the system does not need any external apparatuses to operate or special hardware. All the processing is done through the user's mobile phone and this makes the solution cost-effective and available to many. Such a design decision eliminates the cost pressure for visually impaired people, and raises the convenience, as they can have access to the application without any assistance in different settings.
These gaps are covered by the proposed system with an all-in-one solution built on the top of the pretrained lightweight DL models for indoor object detection and currency recognition. This system incorporate a voice feedback module to assist the visually handicapped users. The accuracy of DL models is high and latency of system is low as compared to others. The suggested system do not have the need for any external hardware hence it is cost effective to the visually impaired users and available to the visually impaired users.

3.3 Requirement Specifications

3.3.1 Functional Requirements

Object Detection: The system shall detect and identify objects using mobile device camera.
The detected objects shall be labeled with predefined class names.
The system shall notify the user of each detected object through text-to-speech feedback.

Currency Recognition: The system shall detect Pakistani currency notes placed in front of the camera.
The system shall classify the denomination of the note using a trained classification model.
The identified currency value shall be announced to the user using audio feedback.
The recognition process shall be initiated via a dedicated "Currency mode" in the app.

Audio Feedback: The system shall provide real-time voice feedback for both object detection and currency recognition.
The system shall use a TTS engine.

User Interface: The app shall provide a simple, voice guided interface suitable for visually impaired users.
The interface shall have two main modes: Object Detection and Currency Recognition.

Mode Switching: The app shall allow users to switch between Object Detection mode and Currency Recognition mode.

3.3.2 Non-Functional Requirements

Performance: The system shall process and detect objects or currency within 10 seconds of capturing an image.
The voice Feedback shall be generated with a maximum delay of 5 seconds after detection.

Usability: The system shall provide a user-friendly interface accessibile to visually impaired users.
All functionalities shall be operable through touch input.

Reliability: The system shall have a detection accuracy of at least 90 percent under standard lighting conditions.

Portability: The app shall be compatible with Android mobile devices (version 10.0 and above).
The app should be optimized to run on devices with at least 3GB RAM.

Maintainability: The code shall follow modular design principles to allow easy updates and debugging.
Model files and configuration files should be separable to support retraining or improvement.

Security: The application shall not store user data or personal information.
Permissios for camera and audio access must be explicitly granted by the user.

3.4 Use Cases

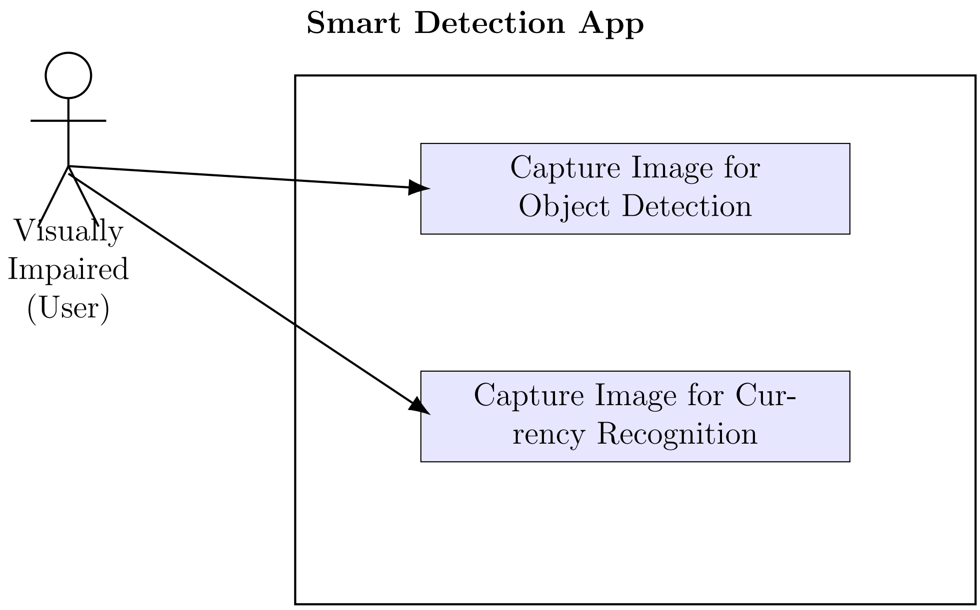

Figure 3.1: Use Case Flow for Visually Impaired User

Table 3.1: Object Detection Use Case

Use Case Name	Object Detection
Actor	Visually Impaired User
Precondition	App is launched and camera is enabled
Postcondition	User receives audio feedback about the detected object
Main Flow	1. User launches the app. 2. Selects the object detection mode. 3. Points the camera toward an object and captures the image. 4. System processes the image and detects objects. 5. Audio feedback is provided to the user via TTS.
Alternative Flow	If camera permission is denied, the app prompts the user to enable it.
Includes	Audio Feedback

3.4.1 Object Detection

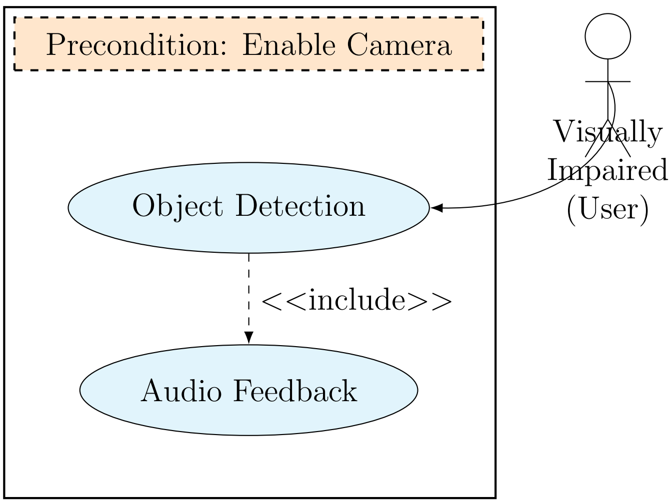

Figure 3.2: Use Case 1

Table 3.2: Currency Recognition Use Case

Use Case Name	Currency Recognition
Actor	Visually Impaired User
Precondition	App is launched and camera is enabled
Postcondition	User receives audio feedback about the currency denomination
Main Flow	1. User launches the app. 2. Selects the currency recognition mode. 3. Points the camera toward a note and captures the image. 4. System analyzes and recognizes the currency denomination. 5. Audio feedback is provided to the user via TTS.
Alternative Flow	If the note is unclear, the app requests the user to retake the image.
Includes	Audio Feedback

3.4.2 Currency Recognition

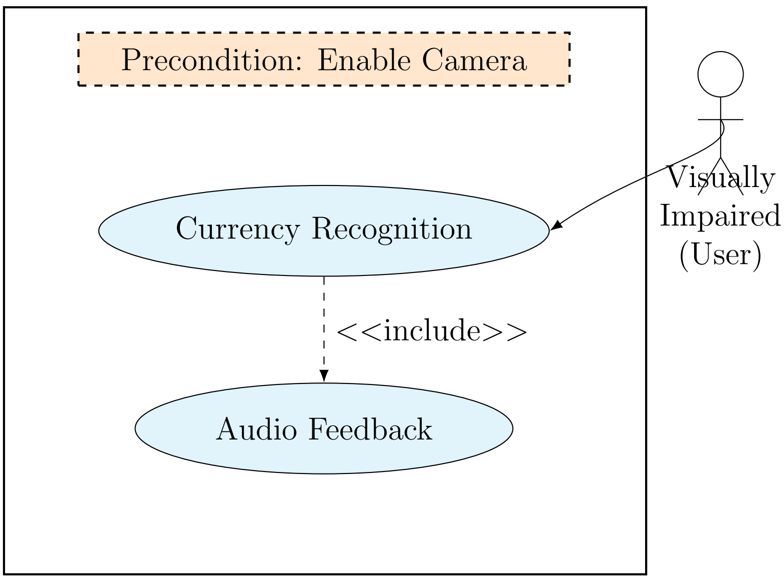

Figure 3.3: Use Case 2

Chapter 4

Design

4.1 System Architecture

The architectural diagram of the Smart Detection App (Figure 4.1) shows the main system components and their interaction to support the visually impaired users. The user interacts with the system through either mode by feeding it a visual input via the smartphone camera. This real-time image acquisition is the foundation of the dual recognition ability of this app: object detection and currency identification.

When the picture is acquired, it gets processed in parallel through two dedicated modules. The Object Detection Module whose foundation is laid in the YOLOv8n-OBB architecture, performs detecting of the common indoor objects such as furniture, appliances or objects that can be important for the user's environment. The Currency Recognition Module at the same time is oriented at the recognition and classification of different denominations of Pakistani currency with the help of a YOLOv8n.

Both modules send the respective outputs to a central Processing and Classification unit. In this, the information is combined and analyzed to establish the suitable verbal feedback for the user. This processed data is then handed over to the Text-to-Speech (TTS) Module that converts the data into clear natural audio.

At last, the audio output is passed directly on to the user which does not require any visual input but provides real-time help.

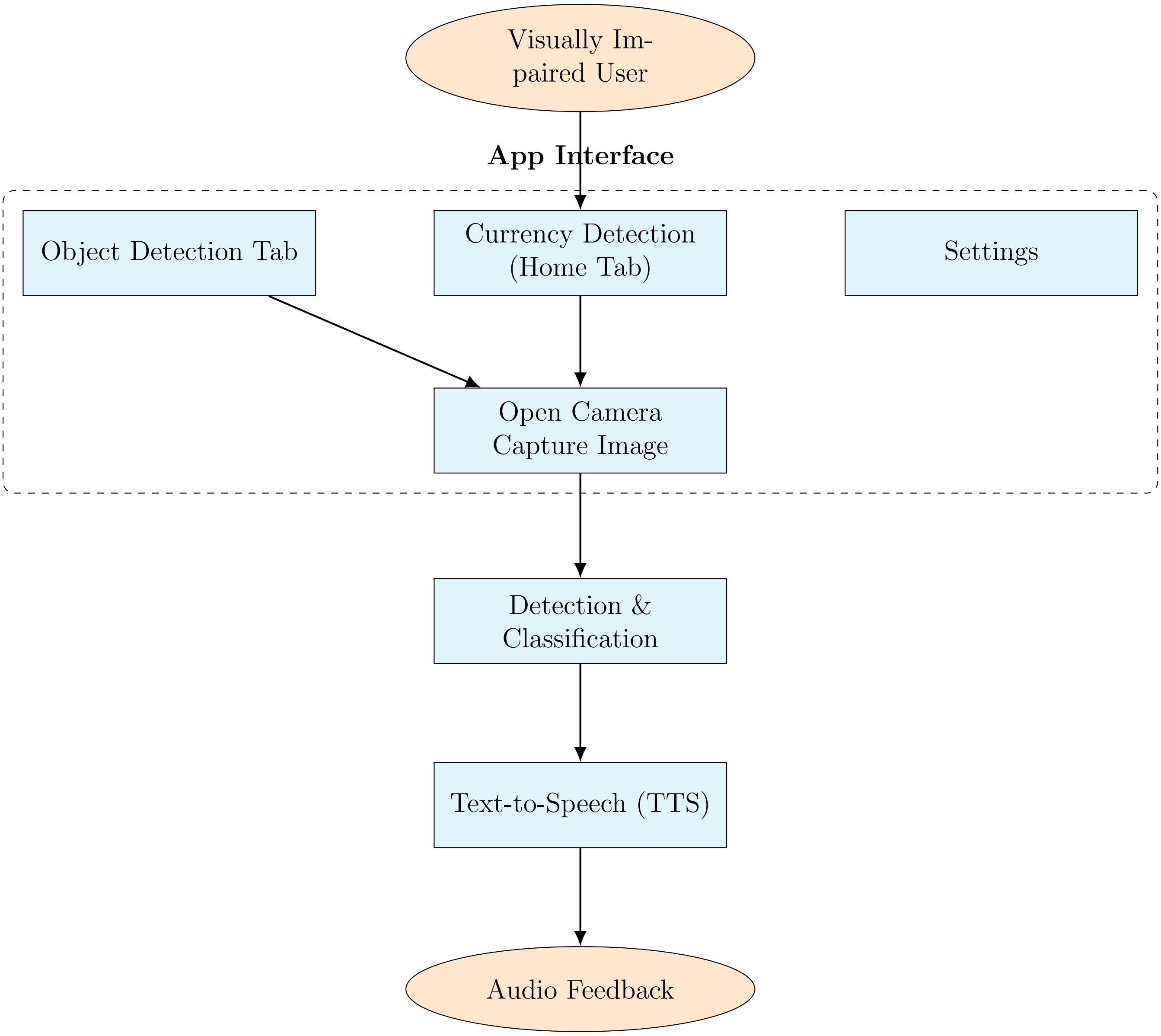

Figure 4.1: System Architecture Diagram of Smart Detection App

4.1.1 Contextual View

The contextual view of system interaction (Figure 4.2) facilitates a general idea toward which the Smart Detection App falls into the target usage environment. It does not only look at its internal mechanics but the external entities that affect and interact with the same system. At the focus, there is the Smart Detection App that serves as the interconnect between visually impaired user and the world around. The user interacts with the app through the camera of the smartphone. This visual recall charts an active picture of the user's immediate surrounding from interior objects to currency notes. Two key external factors feed into the system:

Physical Environment

Currency Notes

General objects are contributed by the physical environment while the currency notes are considered as a different input class because of its specific importance and detection mechanism. These visual elements are received and computed by the app which then gives voice feedback in turn. The verbal clues are meant to help the user's actions and explain objects around them or currencies; thus, enhancing their independency in undertaking day-to-day activities. This contextual model provides an indication of real-world implications of the system. It provides bidirectionally flow of interaction whereby, the user sends input and receives auditory feedback to form a closed loop system. It also explains that the system is not self-contained, it is highly assimilated into the practical environment of the user.

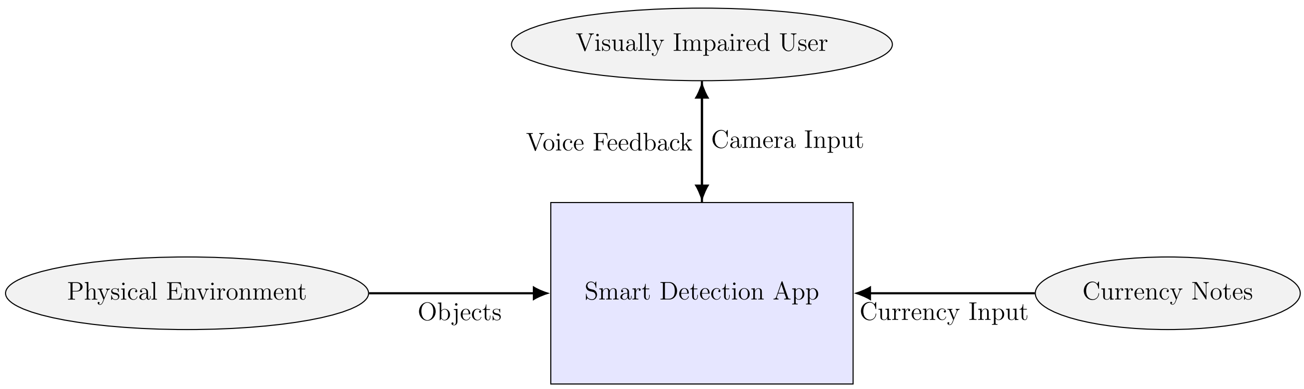

Figure 4.2: Contextual View of System Interaction

4.2 Design Constraints

4.2.1 Hardware Limitations

Over-dependence on the capabilities of the mobile device on which the smart detection application is running renders the efficiency of the smart detection application. Lower power devices may fail to run the models efficiently hence lag or delayed feedback. Besides that, the quality of the mobile camera influences the clarity and resultant of the input frames. Low-quality images, including those that are blurry or poorly lit, may jeopardize the detection of objects and currency, thus making it harder for people with poor eyesight to get proper help. This limitation makes it imperative that the application needs to be optimised for varied range of devices without sacrificing on the performance side.

4.2.2 Model Size

So that the application runs smoothly on regular Android devices, the models of object detection and currency recognition must be lightweight. The excessive memory consumption, long load times, and poor responsiveness may occur due to large sizes of the model especially when working with devices with low RAM. It is essential to maintain the models compact so that in the process of real-time detection, the system resources will not get used up. This usually involves a trade-off between accuracy and size of the model during training, as well as deployment.

4.2.3 Battery Usage

The app needs constant demand of resources such as camera and processor when scanning for objects, or money in real time thereby draining a lot of battery power. Long term use may resulting in very rapid battery deterioration, especially if the device has an older or smaller battery. That is a primary concern for the visually impaired users who rely on their devices the whole day. This impact of that could be mitigated by optimization of processing cycles , use of low-power modes.

4.2.4 User Feedback Accuracy

As it is an application that is supposed to act as a helpful assistant, it should always provide precise feedback. False positive or untrue detections can mislead the user or cause errors in the real world scenarios. For instance, confusing the currency denominational units might prove fatal in terms of financial errors, while wrong object recognition could harm the user's safety. Consequently, the system has to undergo extensive testing and refinement so as to reduce inaccuracies and increase trustworthiness.

4.2.5 Lighting Conditions

The lighting environment whereby the image has been taken has a very high effect on the object detection as well as currency detection performances. Shadows or bad lighting can influence the perceived view on the objects so that the models could not recognise the objects accurately. This constraint is highlighting the need for strong techniques for pre-processing images or advice for users to take photos under well-lit conditions.

4.2.6 Network Independence

The application should have an offline capability that should allow it to work without being connected to the internet. This eliminates the dependency of the users on connectivity particularly in distant or low network places.

4.2.7 Device Compatibility

There is an extremely wide range of Android devices when it comes to hardware specifications, screen sizes, positioning of cameras, and versions of the operating system. Compatibility in a wide variety of divergent devices is a critical constraint. The app should be adaptive and tried out in several configurations to guarantee a coherent and user-friendly experience for all of the users.

4.2.8 Security and Privacy

Although the app does not store the images or personal data from the users, it can process live camera input, which might contain the sensitive information. The prevention of transmission and storage of no data without the consent of a user is vital in ensuring that user trust and privacy standards are maintained.

4.3 Design Methodology

The design methodology of this project outlines a structured approach followed to develop a smart detection mobile application for visually impaired users. The process is divided into several key stages, each playing an important role in building and delivering the final product. Below is a detailed explanation of each stage:

4.3.1 Data Collection

This is the first step where images of Pakistani currency notes and common indoor objects are gathered. The quality and variety of the collected data are crucial, as they form the foundation for training accurate and reliable machine learning models.

4.3.2 Pre-processing

Once data is collected, it undergoes a cleaning and preparation process. This includes resizing images, enhancing quality, removing noise, and formatting them into a consistent structure suitable for model training. Good pre-processing helps improve the accuracy of the models.

4.3.3 Dataset Annotation

In this stage, the collected images are labeled with relevant information, such as the object name or currency denomination. A special annotation tool is used to draw bounding boxes around the items. This labeled dataset is essential for training the models to recognize objects and notes.

4.3.4 Model Training

After annotation, the dataset is used to train two deep learning models. One model is trained to recognize Pakistani currency while the other is trained to detect indoor objects. The models learn to identify patterns and features through repeated exposure to the labeled data.

4.3.5 Model Testing

This stage evaluates the performance of the trained models using a separate set of test data. It checks how accurately the models can detect and classify new images. Necessary adjustments or improvements are made based on the testing results.

4.3.6 App Development

In this phase, a mobile application is developed using Flutter. The app serves as the interface where users interact with the system. It provides features like camera access, mode selection, and communication with the models for detection.

4.3.7 Integration of Models and App

The trained models are integrated into the mobile application. This step ensures that the app can use the models to detect objects and currency notes in real time and display or speak the results to the user through voice feedback.

4.3.8 System Testing

The complete system including the mobile application and integrated models is tested thoroughly. This ensures that all components work together smoothly and the application performs well under real conditions.

4.3.9 System Deployment

Finally, the application is deployed for actual use on Android devices. This step involves making the app available to users, ensuring it runs properly, and is ready for practical use in real environments.

This structured design methodology ensures a reliable and efficient development process that results in a functional and user-friendly application tailored to the needs of visually impaired users.

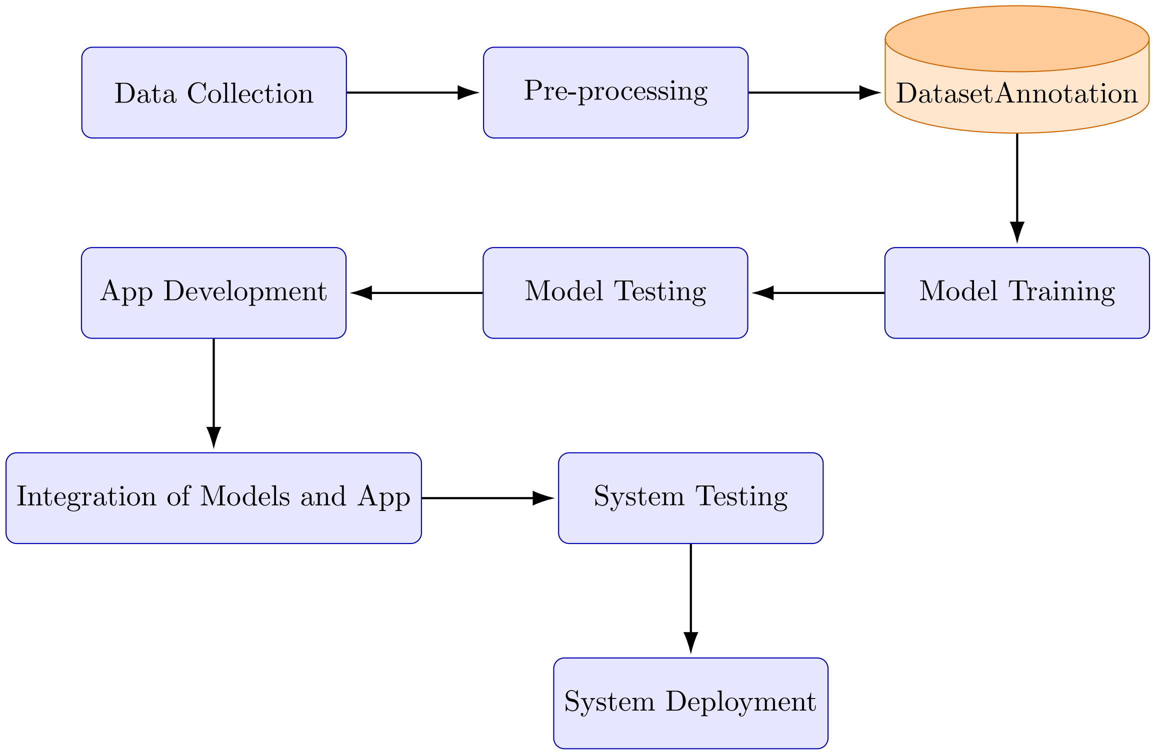

Figure 4.3: Design Methodology Diagram

4.4 High Level Design

The high-level design breaks down the architecture into functional views as follows:

4.4.1 Conceptual View

Represents the logic of the system with such main modules as Camera Input, YOLOv8 Object Detector, Currency Classifier, and TTS.

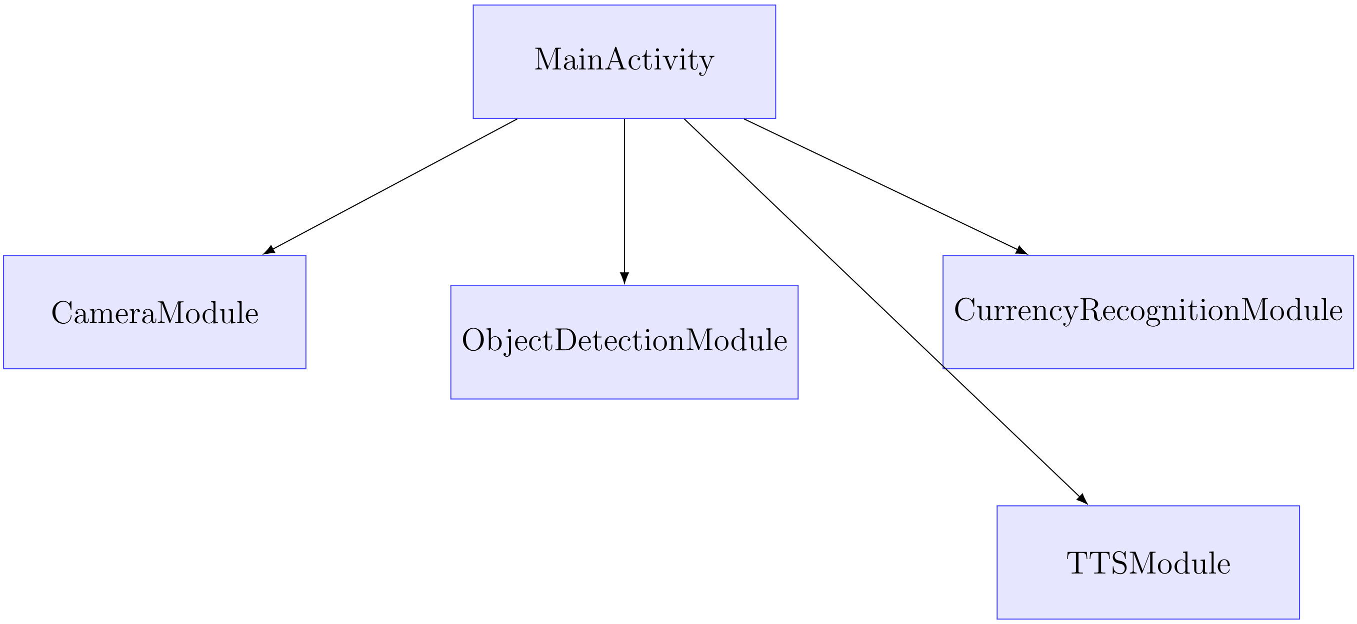

Figure 4.4: Component Diagram

4.4.2 Process View

Illustrates the runtime data flow between modules using a Data Flow Diagram and Activity Diagram.

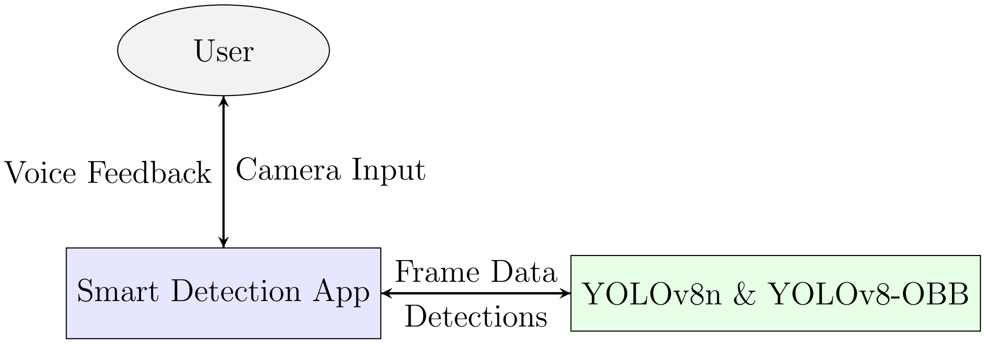

Figure 4.5: DFD (Level 0)

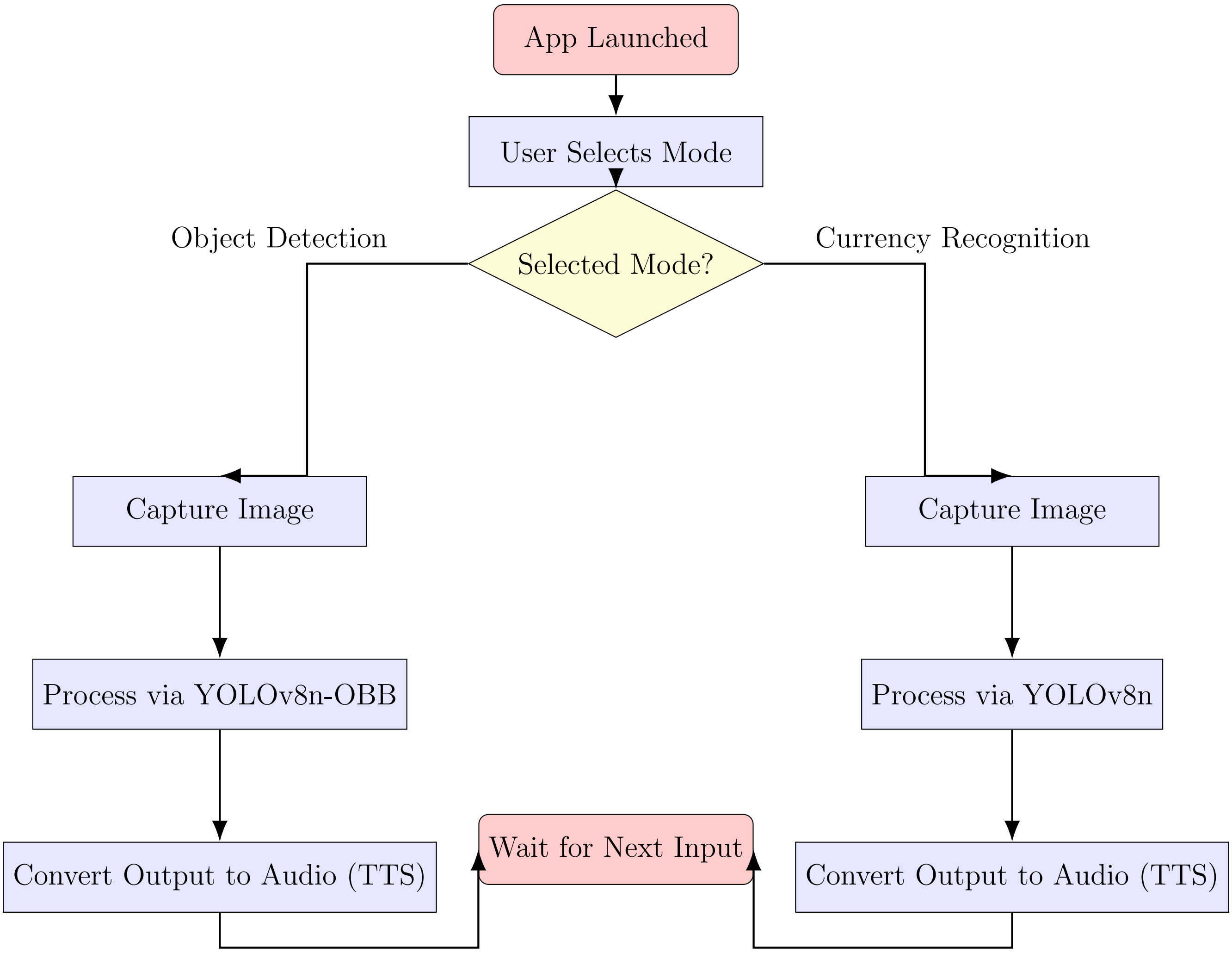

Figure 4.6: Activity Diagram

4.4.3 Physical View

Depicts system deployment; all components are hosted on a smartphone device.

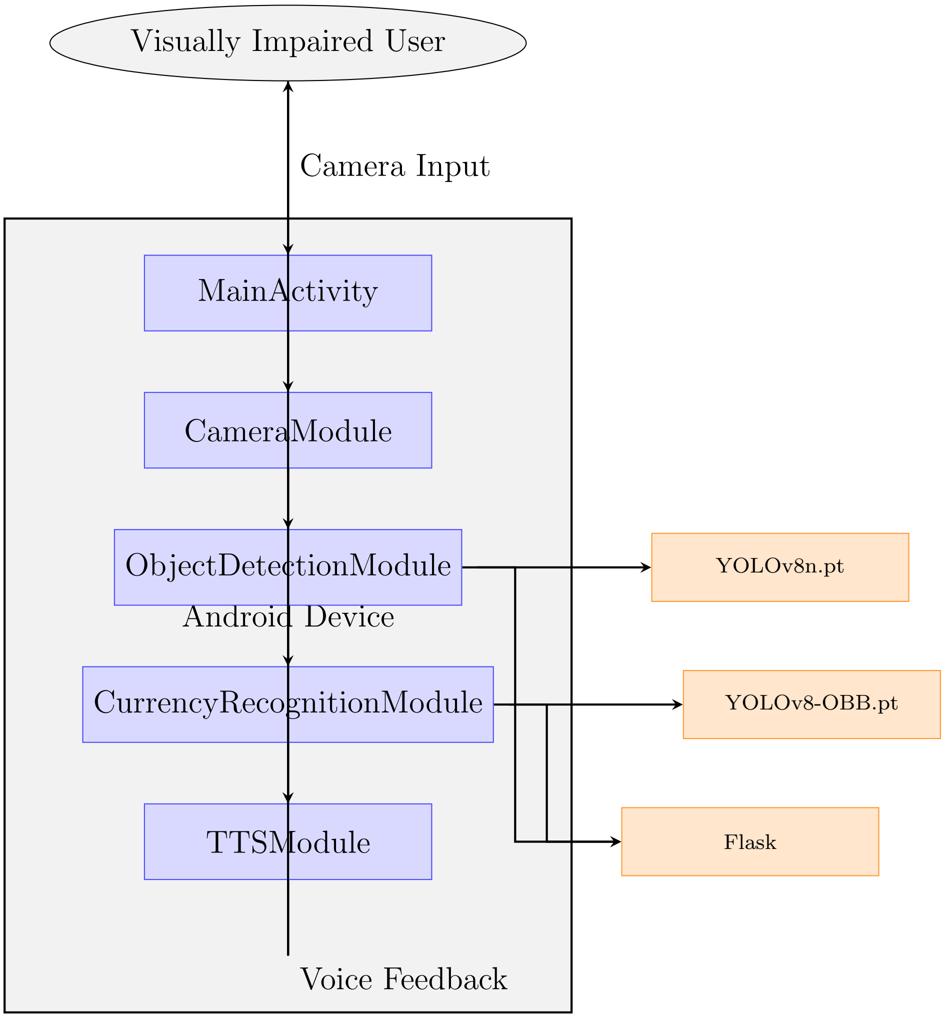

Figure 4.7: Deployment Diagram

4.5 Low Level Design

4.5.1 Class Diagram

Each major module is described through classes and their interactions. The system is organized into:

CameraModule: Captures images.

ObjectDetectionModule: Detects objects using YOLOv8n-OBB model.

CurrencyRecognitionModule: Detects currency denomination using YOLOv8n.

TTSModule: Converts text output to speech.

MainActivity: UI controller, managing user interaction and module coordination.

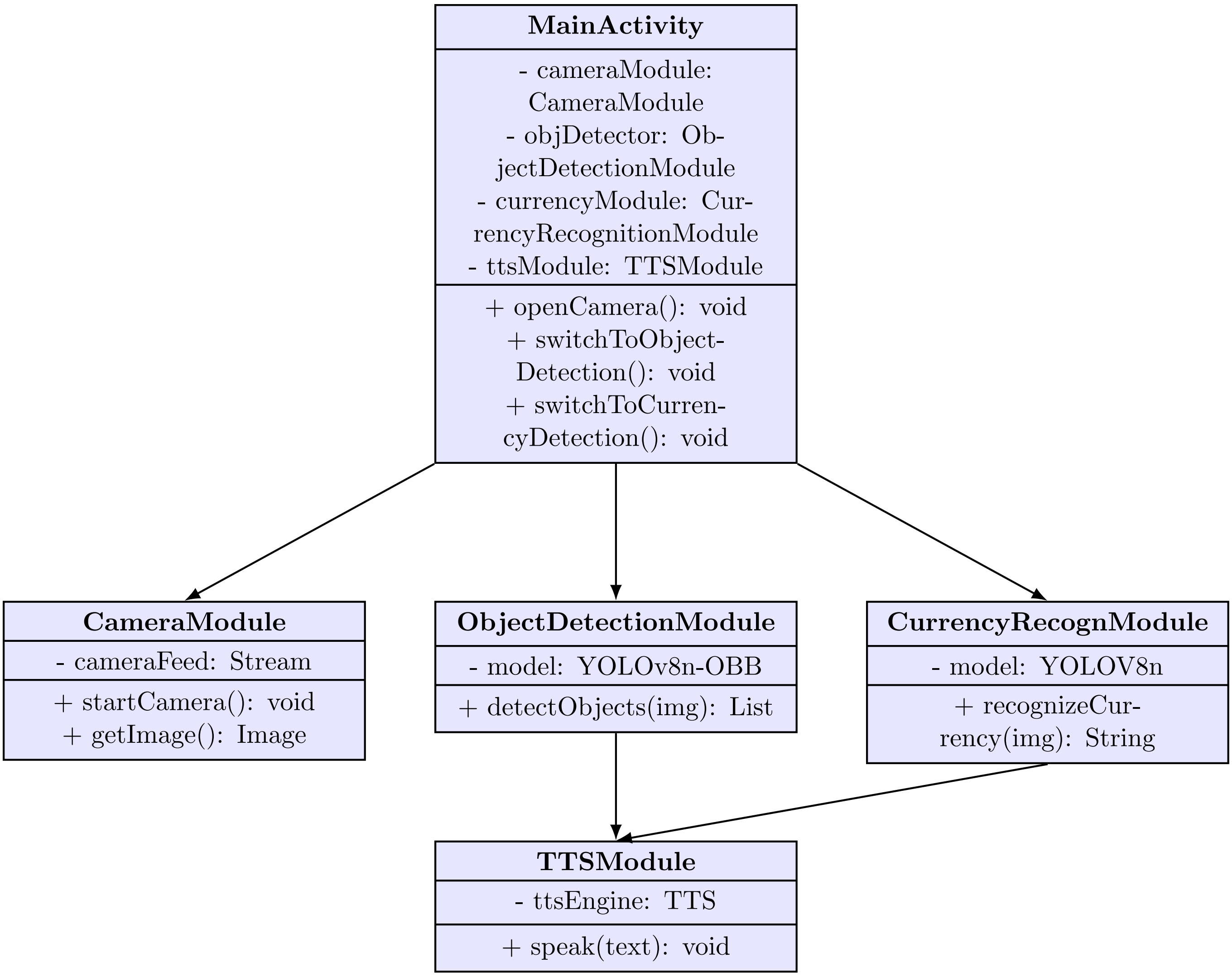

Figure 4.8: Class Diagram

4.5.2 Sequence Diagram

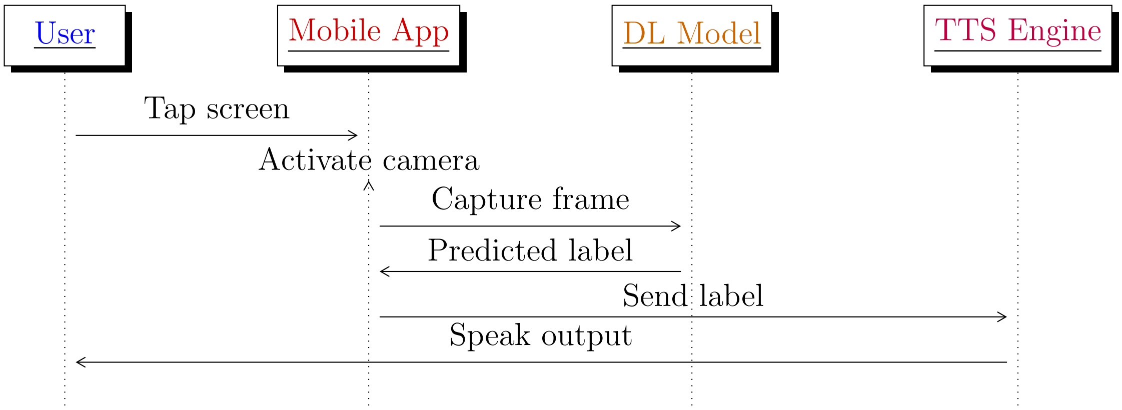

Figure 4.9: Sequence Diagram

The sequence diagram illustrates the interaction between the user, the mobile application, the deep learning model, and the text-to-speech engine. It represents the flow of actions from the moment a user initiates the process to the point when the system provides audio feedback.

User: The process begins when the user taps on the mobile screen to start the detection process. This input is received by the application.

Mobile App: Once the tap is detected, the app activates the camera and captures a frame. The captured image is then passed on to the deep learning model for analysis.

DL Model: The deep learning model processes the input image and returns a predicted label. This label could be the name of a detected object or the denomination of a currency note.

TTS Engine: After receiving the label, the mobile application sends it to the text-to-speech engine. The TTS engine converts the label into a spoken message and delivers the voice feedback back to the user.

This interaction ensures that the visually impaired user receives real time information about their surroundings through voice output. It emphasizes a smooth and quick flow from user action to system response, enhancing the accessibility and usability of the application.

4.6 Database Design

Instead of conventional database system, this project is more on gathering and processing custom datasets in order to train machine learning models to detect objects and recognise currency. While the data is the spine of the detection abilities of the app, it will be chosen very carefully to fit the needs of blind users in Pakistan.

4.6.1 Dataset Collection Process

The dataset used in this project was handcrafted so as to make it depict the real-world scenarios. The currency recognition model involved gathering and taking images of different Pakistani currency notes under different conditions. Such conditions consisted of different lighting conditions, angles, backgrounds and positions of the notes in an attempt to assist the model generalize better. Likewise, for the object detection, a list of frequently found objects inside homes and households was compiled, and various pictures of these selected objects were taken with the help of a conventional camera on a standard smartphone.
By the means of using gathered images inside local surroundings and from true devices the dataset is very similar to an input that will be used for real app utilization. This approach enhances the relevance and the reliability of the trained models during practice.

4.6.2 Annotation with Roboflow

Once the necessary pictures were obtained, the dataset was uploaded to Roboflow, an online service that is highly popular for dataset management, image preprocessing, and annotation. All the images were, then, manually annotated using Roboflow's in-built annotation tools. For currency recognition dataset the annotations contained denomination labels like "PKR-10", "PKR-20" etc. For the object detection dataset, labels represented such things as "chair", "clock", "laptop" or other everyday house interior objects.
The consistency of annotations and flexibility to resize and augment, as well as prepare the images according to the model's input requirement was made possible by roboflow. It also enabled exporting the annotated datasets in YOLO format that can be used in the modern deep learning frameworks.

4.6.3 Importing Dataset for Model Training

When the annotation process was over, the revised datasets were exported from Roboflow and then transferred to Google Drive. This made it easy and convenient option and persistent storage that could be accessed from any place.
The datasets were then uploaded to Google Colab, a cloud based Jupyter notebook environment that has the ability to perform GPU acceleration and has easy integration with Google Drive. This setup enabled the rapid training of YOLOv8n-OBB model for object detection and YOLOv8n for currency classification with the help of annotated datasets.

4.6.4 Benefits of Custom Dataset

By using a custom-built dataset, there were several benefits including:

Achieved local currency and objects relevance, representing those objects which users are interacting with in real life.

Faculted the ability to control data quality, annotation precision, and labeling conventions completely.

Helped it make it possible to train lightweight, efficient models that can be deployed in mobile devices.

4.7 GUI Design

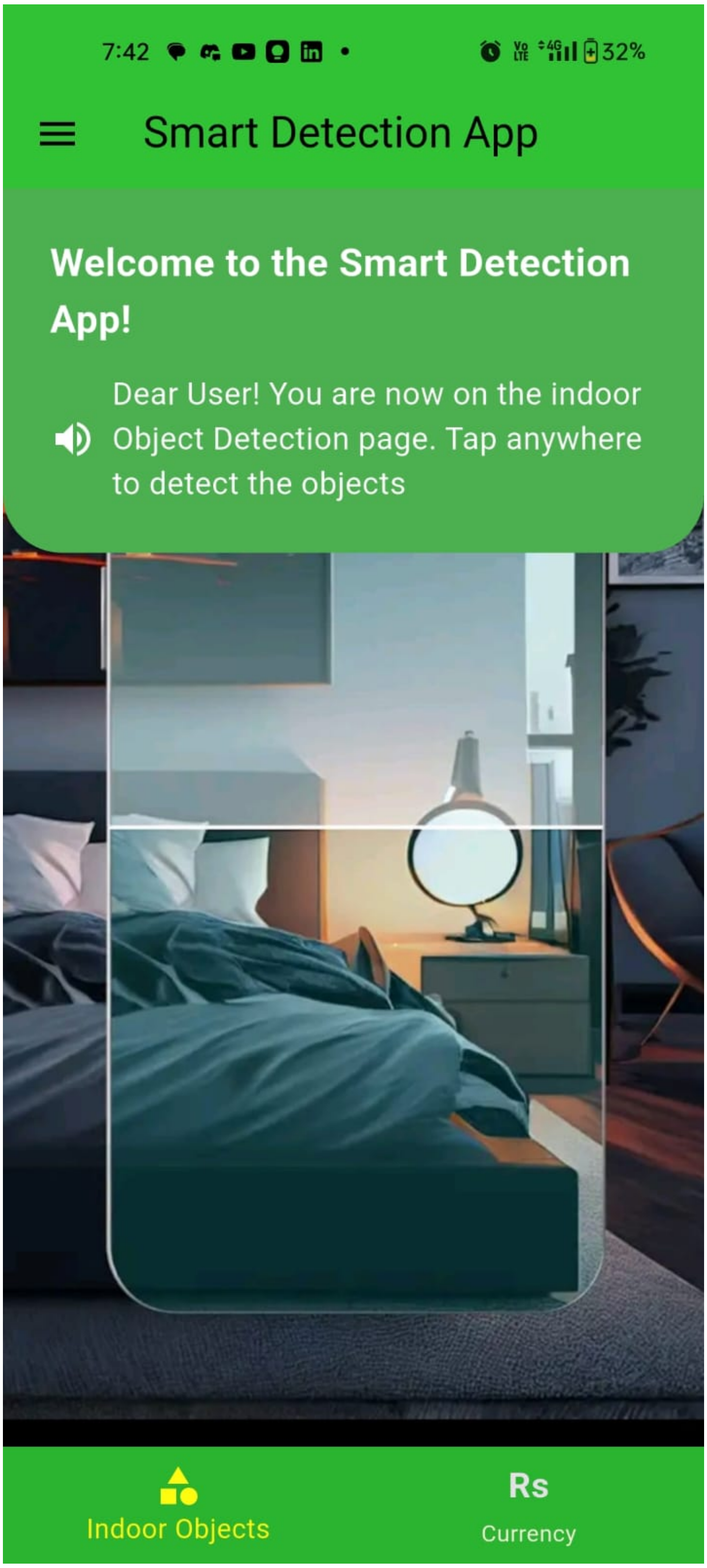

Figure 4.10: User Interface

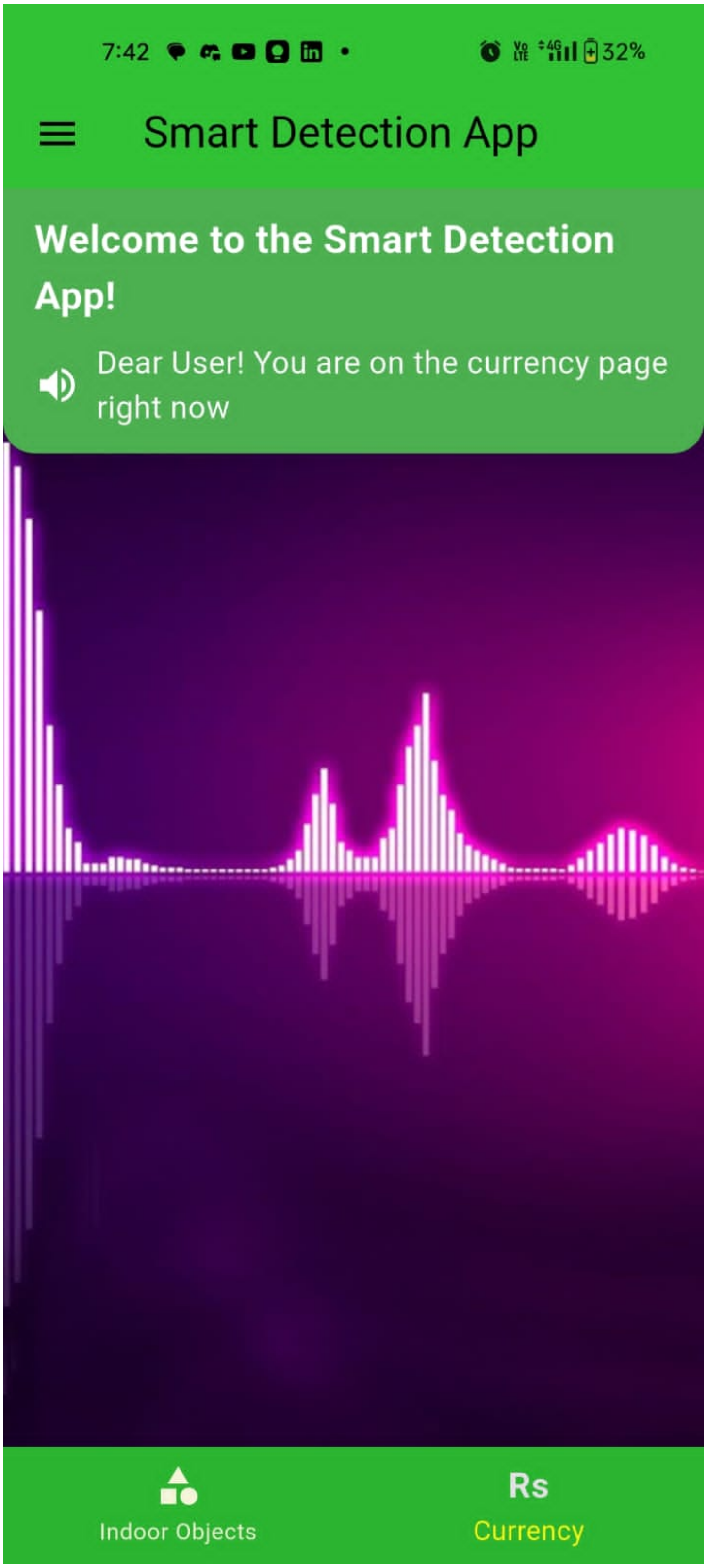

Figure 4.11: User Interface

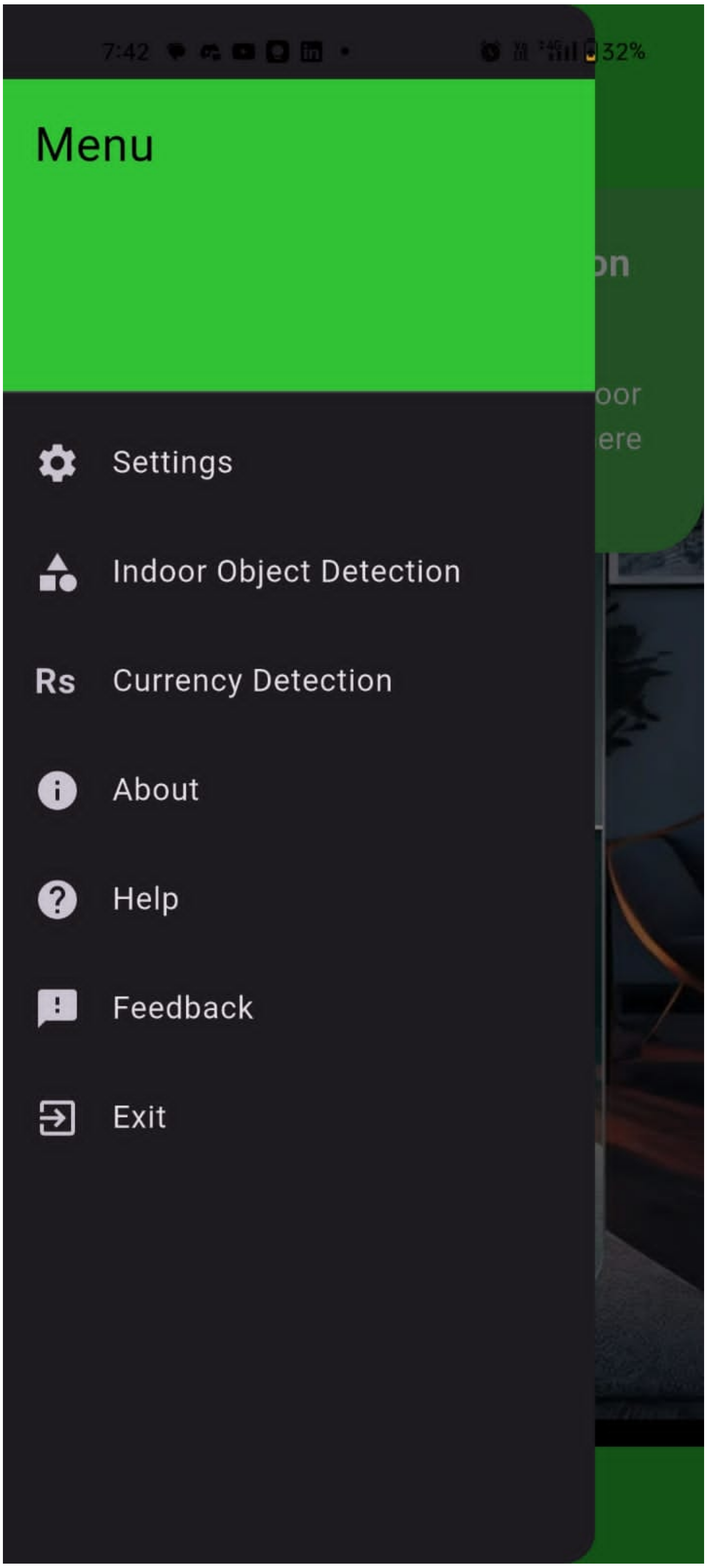

Figure 4.12: User Interface

4.8 External Interfaces

The operation ability of the Smart Detection App involves smooth connection with a number of external interfaces. These components offer indispensable services that are not implemented from the scratch but incorporated into the system for efficiency, performance, and reliability. Listed below is an elaborate discussion on each of the interface employed:

4.8.1 Android Camera API

It is used to retrieve the device's in-built camera for real-time visual data capturing. Continuously, the app retrieves frames from the camera, which in turn is worked on by the object detection and currency recognition modules. The Android Camera API includes several functionalities like auto-focusing, resolution control that are very important elements in attempts to record quality input in various lightings.

4.8.2 Flutter (TTS) Library

This interface is used to transform the detected and identified visual information into communication feedback in form of voice to the user. Flutter TTS library is selected due to the clear pronunciation and easy adaption to Android platforms. It guarantees the users real-time verbal feedback from the system something that is particularly important in the case of visually impaired who rely totally on audio clues to interact with the system.

4.8.3 Flask API for Model Inference

The deep learning models that are used in this application are deployed to the mobile application through a Flask API where it acts as a light weight and fast channel between the mobile application and the server-side model inference engine. Instead of converting the models into other formats, the trained PyTorch models are deployed to a backend server that is used to serve the requests to the Flask API. When the user captures an image, the app communicates with the server using the API and sends its image to the server, whereby the image is processed by the relevant model, be it for the object detection or currency classification. The results of prediction are returned back to the app in real time. This approach makes inclusion of models easier, provides scalability, and guarantees accurate and speedy inference with flexibility relating to models management and update.

Chapter 5

System Implementation

Implementation is the stage where theoretical concepts and design ideas are transformed into a working product. In this project, it involves developing a mobile application that can recognize indoor objects and currency notes using deep learning models. The final system meets the initial design specifications and is optimized for real-time performance and accessibility.

5.1 System Architecture

The system is developed using Python for model training and Flutter (Dart) for the mobile application. The models are deployed using Flask API. The Flutter Text-to-Speech (TTS) service is used to convert text output into voice feedback. The application follows a modular and component-based architecture to maintain flexibility and scalability. It consists of multiple functional modules that work together to provide object and currency detection along with audio feedback.

5.1.1 System Components and Functionality

MainActivity: Acts as the main interface for users. It manages camera input, mode selection, and starts the detection process based on user input.

CameraModule: Connects to the device camera and captures real-time frames. It provides quality input for accurate model predictions.

ObjectDetectionModule: Uses the YOLOv8n OBB model, which is a fast and efficient real-time object detector. It identifies and marks multiple objects in the frame.

CurrencyRecognitionModule: Uses a YOLOv8n model trained on Pakistani currency to recognize different denominations from captured images.

TTSModule (Text-to-Speech): Converts the detection results into voice output using the flutter TTS.

5.1.2 Communication Between Components

All components are coordinated through the MainActivity controller. The CameraModule captures images and sends them to the selected detection module. Once processed, the output text is sent to the TTSModule to provide voice feedback. This seamless flow ensures a minimal delay between image capture and audio output, enhancing the user experience.

5.1.3 Tools and Technologies Used

Programming Language: Python for model training and Dart for mobile application development

Frameworks: PyTorch for deep learning and OpenCV for image processing

Model Architecture: YOLOv8 OBB for object detection and YOLOv8n for currency recognition

IDE: Visual Studio Code for mobile app development and Google Colab for model training

Platform: Android with support starting from Android 9 and above

5.1.4 Processing Logic

When the app is running, the user selects either object detection mode or currency recognition mode. Once selected, the user taps anywhere on the screen and it captures an image. Depending on the selected mode, the image is sent to the YOLOv8n model for currency recognition or to the YOLOv8 OBB model for object detection. The detected result is then sent to the TTS engine for audio feedback.

5.1.5 Application Access Security

Security measures have been added to the application to ensure safe and limited access:

Authentication: The app uses the native Android lock screen for protection. It does not collect or store any sensitive personal data.

Authorization: As the app is single-user, role-based access is not needed.

Safe Data Storage: Temporary image data is not saved after detection. This ensures privacy and proper memory management.

Auditing/Logging: No personal or biometric data is recorded or stored.

5.1.6 Database Security

Since the application does not use a persistent or external database, most common database security concerns do not apply. However, the system is prepared for future upgrades.

Authentication/Authorization: A secure backend or Firebase can be added later if cloud storage or online features are introduced.

Auditing: Currently, no logs or user data are permanently stored.

Data Handling: Image processing is done locally, and temporary data is cleared post-processing.

Chapter 6

System Testing and Evaluation

For this project, which is targeted towards improving the life of the visually impaired users by offering them an Android OS smart detection app with audio response, it was especially important to execute the tests because of the sensitive and high level of accuracy needs of the end users. It was necessary to ensure not only technical correctness but usability of accessibility of the application.
This chapter provides an overview of the process of testing carried out for this project. The applicant went through detailed evaluations in the term of functionality, performance, ease-of-use and security, whereby the application should meet expectations not only in absolute terms but also with regards to what is already on market in the domain.

6.0.1 Test Case Tables

Table 6.1: Object Detection Test Case

Test Case ID	TC_OD_001
Test Case Name	Object Detection with Valid Object
Precondition	App is launched and camera is enabled
Test Steps	1. Launch the app. 2. Select object detection mode. 3. Point camera to a valid object and capture image.
Expected Result	The app should detect the object and provide correct audio feedback
Actual Result	The app successfully detected the object and provided correct audio feedback
Status (Pass/Fail)	Pass

Table 6.2: Currency Recognition Test Case

Test Case ID	TC_CR_001
Test Case Name	Currency Recognition with Valid Note
Precondition	App is launched and camera is enabled
Test Steps	1. Launch the app. 2. Select currency recognition mode. 3. Point camera to a valid currency note and capture image.
Expected Result	The app should recognize the currency note correctly and provide accurate audio feedback
Actual Result	The app successfully recognized currency note and provided accurate audio feedback
Status (Pass/Fail)	Pass

6.1 Graphical User Interface (GUI) Testing

The GUI was tested to make sure the interface was user friendly, coherent, and accessible in particular taking into account the future users. Although the app is designed for visually impaired users, the layout and interactions were tested for ease and clarity to facilitate partial vision or aiding tools such as screen readers.
Key checks included:

Placements of the buttons and choices for mode selection were distinguishable.

Text to speech feedback was well-implemented with UI events.

Icons and color schemes were available to individuals with low vision.

6.2 Usability Testing

Simulation of realistic use was used for the usability testing. The objective was to make sure that the users are capable of using the app spontaneously with little learning curve. This involved carrying out checks on the mode selection (object or currency detection), image capturing and voice feedback parts.
Through a small focus group of test users and observers, feedback was collected from which changes were implemented in consideration of what they learned.

6.3 Software Performance Testing

The app's performance was measured in terms of:

Detection speed and accuracy of YOLOv8n-OBB for object detection.

Accuracy of currency classification using YOLOV8n.

Response time from image capture to voice feedback.

The models ran efficiently on middle-level Android devices where the average latency of detection is sustained to less than 1 second for object, and about 1.2 for currency recognition. These results testified that the app was in a position to provide real-time assistance to people without any significant lags.

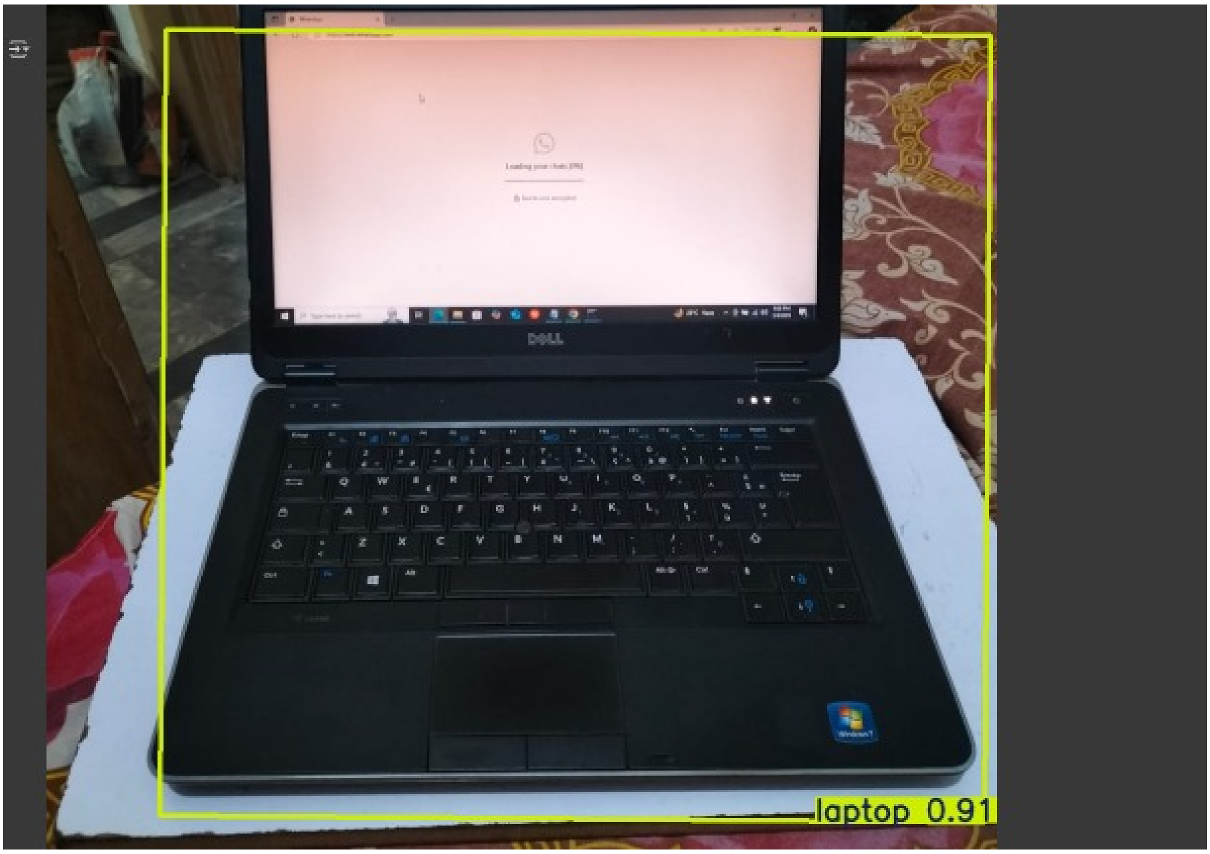

Figure 6.1: Laptop

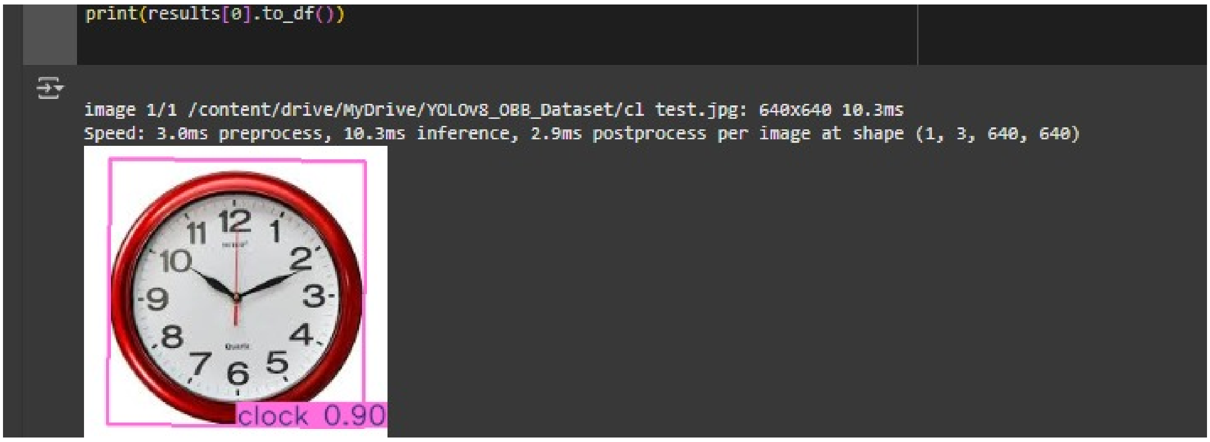

Figure 6.2: Clock

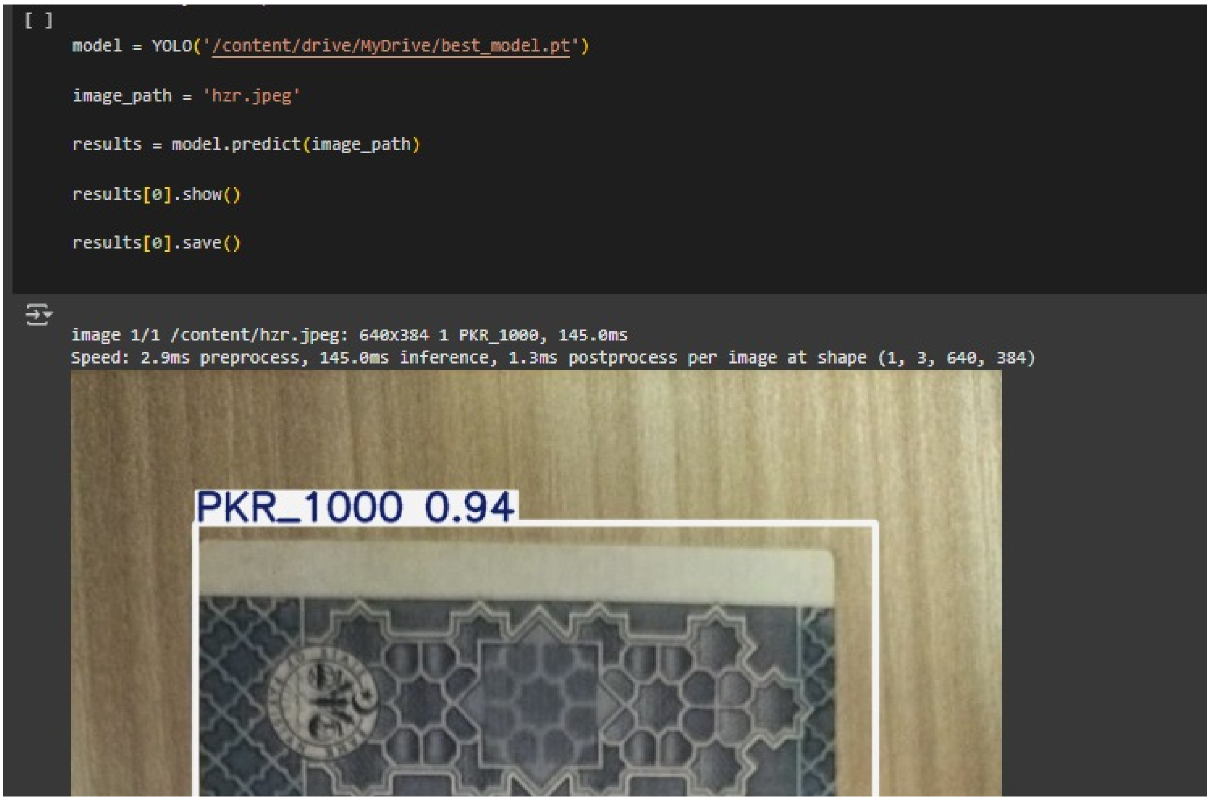

Figure 6.3: Currency

6.4 Compatibility Testing

The app was tested on multiple Android devices of varying specifications and screen sizes to verify compatibility. This ensured that the core functionalities image capture, processing, and audio feedback performed consistently across different environments.
Compatibility tests confirmed that the application worked seamlessly on Android 9 and above, on android phone.

6.5 Exception Handling

For testing robustness of the system, various scenarios of exceptions were used such as:

Camera unavailability or permission denial.

Detection of objects/currency failed.

The app was well able to handle these exceptions gracefully, either by retrying the operation or offering error messages in words.

6.6 Load Testing

The app though light weight and not meant for situations of high concurrency environments, basic load test ran checks on how the application behaves in the process of re-run operations.
The app underwent continuous process with repeated mode switch and detection duties. It demonstrated stability on extended use with fewer memory leaks and no crashes which meant that it has reliable performance under load.

6.7 Security Testing

Security testing were concerned with privacy and integrity of captured data and internal working. Since the application handles sensitive details such as currencies values and surrounding objects, it was very essential to ensure:

Not a single item of data was stored or transmitted without the user having been fully informed and agreed to it.

The app required permissions only when necessary (camera).

Android security best practices were observed in the development process.

6.8 Installation Testing

The process of installation was tested in various Android devices to ensure that the process was smooth and without error. The manual installation of the app package (.apk) was performed. Uninstallation was tested as well in order to ensure the removal of associated data.

6.9 Evaluation Summary

Although the application accomplished most of its key purposes, the evaluation process indicated a couple of the limitations:

There may be some variation in performance by devices of very low processing power.

Detection accuracy is susceptible in very cluttered or in dark environments.

However, the strengths of these apps are enormous such as; real-time feedback, user-friendliness and accurate detection, which far supersedes the limitations. The future research can be dedicated to improving the detection models.
Results showed accuracy between 91% to 94%.

Chapter 7

Conclusions

The completion of this Final Year Project is an important milestone in developing an assistive technology solution to empower the visually impaired persons. During the design, implementation, and testing of the mobile application in the object and currency recognition, a number of important insights and teachings come forth.
From the technical point of view, the implementation of real-time object detection using the YOLOv8 model has shown how advanced deep learning algorithms could be successfully implemented in mobile devices without harming performance. The capability of detecting everyday objects and currency notes with fair accuracy has confirmed the actual applicability of the present solution in helping visually impaired users navigate their environment and to independently perform everyday tasks. Moreover, integrating Flutter Text-to-Speech (TTS) to give the auditory feedback has made it even easier for the user to interact with the application without requiring the visual cues.
The value of using the modular structure to the design of software has been one of the most important lessons learned in the course of this project. Even though it has various components such as the Camera Module, Object Detection Module, Currency Recognition Module, and TTS Module, the system has retained clarity, scalability, as well as maintenance friendliness. All modules deal with clearly defined tasks, and can be upgraded or improved separately, and this provides long-term flexibility.
The process of implementing a product also emphasized the notion about the need to optimize performance and manage resource in mobile applications, particularly in real-time image processing. Some of them include keeping latency low during detection, controlling accuracy dependence on lighting variability and presenting appropriate calibration of the hardware and software components. Using experimentation and testing, we managed to find the balance between speed and precision, an important component of real world tasks.
From the point of view of accessibility and usability, the feedback-focused design has shown that even such interface choices as the time of the voice prompts can make a big difference to the users' experience for visually impaired people. The use of such elements was not only a design decision but also an expression of empathy towards the end-users, an element that will be used in subsequent iterations for the project.

7.1 Future Enhancements

Though this current application serves its purposes in the core, many room for future improvement exist. One possible improvement might be to incorporate natural language

processing (NLP), so that users can give voice command, as this would further make interaction in the system easier for users who might find touch navigation difficult.
Further, to make the application more skewed to the needs of the users from varied regions and backgrounds; multilingual voice output and the extension of object recognition abilities to a broader spectrum of categories of objects could be integrated.
Lastly, cooperation with the organizations standing behind the visually impaired communities would be helpful to perform large scale testing of users. Their feedback may prove to be priceless information about the usage in the real world that will help us model the app into a stronger and more user-centric product.

7.2 Concluding Remarks

Finally, this project did not only succeed in its technical goals but also showed the enormous impact that technology can have if it is constructed with a social purpose in mind. The trip from the idea to implementation has included a multi-faceted learning process of software design, integration of deep learning, as well as develop UI testing from accessibility perspective and user centric evaluation.
Despite the limitations, the basic development of the system demonstrates that for technology to be available it can be functional and empowering as well. With deeper research, user interaction, and technical improvement, this app can well emerge to be a truly powerful utility for the vision-impaired population, allowing it to be independent and part of a swiftly digitizing world.

Appendix A

User Manual

Installation

Download the APK file.

Grant camera permission.

Using the App

Open the app and select either object or currency mode.

Point the camera, capture image and listen to the feedback.

Troubleshooting

Ensure camera permissions are granted.

Use in well-lit environments.

Bibliography

[1] Dr. Hiren PhD , *Object Detection System for Visually Impaired Persons Using Smartphone*, ResearchGate, 2022. [Online]. Available: https://www.researchgate.net/publication/356066523_Object_Detection_System_for_Visually_Impaired_Persons_Using_Smartphone

[2] Abilash and Hari Vingesh, *Currency Recognition for the Visually Impaired People*, IEEE, 2023. [Online]. Available: https://ieeexplore.ieee.org/document/9753373

[3] Sethuraman, Gaurav R, Tadkapally, *MagicEye: An Intelligent Wearable Towards Independent Living of Visually Impaired*, arXiv, 2023. [Online]. Available: https://arxiv.org/abs/2303.13863

[4] Hiba Najam , *Assisting Blind People Using Object Detection with Vocal Feedback*, arXiv, 2024. [Online]. Available: https://arxiv.org/abs/2401.01362

[5] Nuredin Ali AbdelKadir, *Recognition for Visually Impaired People (Case of Ethiopian Note)*, arXiv, 2024. [Online]. Available: https://arxiv.org/abs/2209.03236

[6] Nasir Uddin Faysal, Thamid, *Advancing AI-based Assistive Systems for Visually Impaired People*, IEEE, 2024. [Online]. Available: https://ieeexplore.ieee.org/document/10303628

[7] Samuel Alamirew, *Developing an Assistive Technology for Visually Impaired Persons: Ethiopian Currency Identification*, ResearchGate, 2024. [Online]. Available: https://www.researchgate.net/publication/377461186_Developing_an_Assistive_Technology_for_Visually_Impaired_Persons_Ethiopian_Currency_Identification