VISUALLY IMPAIRED INDOOR ASSISTANT

by

Kareem Emad El Din
Nouran Khaled
Shehab Mohsen
Sherif Akram

A dissertation submitted in partial fulfillment of the
requirements for the degree of
Bachelor of computer science

in

Department of Computer Science

in the

Faculty of Computer Science
of the
Misr International University, EGYPT

Thesis advisor:
Dr. Ammar Mohamed
Eng. Haytham Metawie

(July 2020)

Abstract

Visually impaired people struggle to live without assistance or face any aspect of life alone especially with people that cannot afford extra assistance equipment. Usually impaired people receive assistance by either human or wearable devices. The first one bears the burden on the human, while the second adds financial burdens nevertheless the hassle of identifying an object is not decreased. Smartphones are almost accessible to everyone and equipped with accessibility features including sensors that can be utilized to help both visually impaired and sighted people. Thus, this thesis proposes an approach using CNN, speech recognition and smartphone camera calibration aiming at facilitating the process of indoor guidance for visually impaired people. A smartphone's camera acts as the user's eyes. A pre-trained CNN model is used for object detection and the distance to objects is calculated to guide the user toward the right directions and to warn them of obstacles. The speech recognition part is used as a communication channel between visually impaired people and the smartphone. Also, the proposed approach supports object personalising that helps to distinguish user's item from other items found in the room. To evaluate the personalized objected detection, a customized dataset is created for two objects. The experimental results indicate that the accuracy is 92 and 87 percent for both objects respectively. Also, we experiment the detect distance of two objects against their real distances. The results achieve 0.05 and 0.08 error ratio.

Acknowledgments

We are heartily thankful to Dr. Ashraf, Dr. Ammar and Eng. Haytham also we would like to express our gratitude and would like to thank Engineer Salama Mohamed, because of all the great help and assistance he had provided in guiding us toward tackling the most common issues for the visually impaired people.

Glossary

Term	Definition
Adruino	is an open-source electronics platform based on easy-to-use hardware and software
CNN Android	library for parallel execution of CNN on Android devices
GPS	receivers take information which is transmitted from the satellites and uses triangulation to calculate a users exact location
RaspberryPi	can analyse an image, looking for items of interest and even recognizing faces and text
Tensorflow	open source deep learning framework for on device and off device inference.

Abbreviations

Term	Definition
CNN	Convolutional Neural Network.
COCO	is a large-scale object detection, segmentation, and captioning dataset.
DNN	Deep Neural Networks.
MVC	Model-View-Controller design pattern used to cope with changing code.
RGB	(red, green, blue)
SDD	Software Design Document
STT	Speech to text.
SURF	Speeded up robust features.
SSD	Single-Shot Detector.
TTS	Text to speech.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

1.1 Introduction

1.1.1 Background

Nowadays according to the World Health Organization[1], visually impaired people are in outrageous growth due to the leading causes of vision impairment, uncorrected refractive errors, and Cataracts. Globally, it is estimated that approximately 1.3 billion people live with some form of vision impairment as shown below in Fig 1.1. According to the latest survey provided by the World Health Organization, there are more than 2.2 million people with visual impairment in Egypt, 900,000 of which are totally blind. Many solutions have been devised, however, they're either too high in cost which makes them unavailable and affordable to most of the people, or inefficient products that can't be used in natural environments.

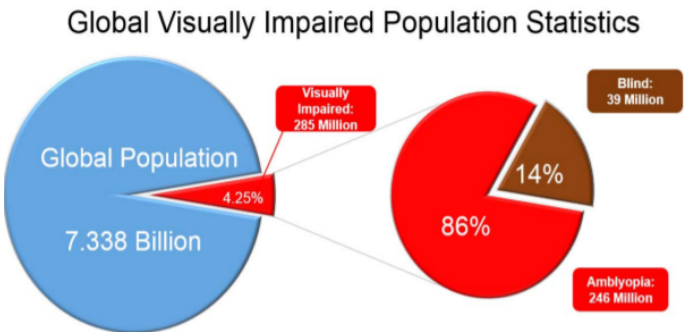

Figure 1.1: The ratio of visually impaired people to the world's total population

The biggest challenge for a visually impaired person, especially for someone with complete vision loss according to National Academies Press (US), is to navigate around new places safely and as mentioned by Abbas et al[2] getting things of independence, which is one of the ultimate goals that a person with impairment might strive to have. Many solutions have been proposed to achieve this goal, but unfortunately most of them are very expensive which makes them out of reach for most of the people, especially in countries with high poverty rates. External assistance might not be available all the time, because people naturally do not like to feel impaired or to be completely dependent on others. Smartphones were considered in this approach due to their high availability nowadays and due to their features and sensors that have high accessibility that could be utilized for the aid of the visually impaired, the camera is utilized to act as the user's eyes. Therefore, this paper will help them a lot in being partially dependent on themselves. It is not safe for visually impaired people to navigate on their own because there might be some obstacles or consequences along their way, as there might be stairs or sharp objects in their surroundings, which might put their life in danger, this approach also tends to the need of object personalization as visually impaired people usually need access to specific items that might not be present in our object detection dataset.

1.1.2 Problem Analysis

The biggest challenge for a visually impaired person, especially for someone with complete vision loss, is to navigate around new places safely as well as getting things of independence, which is one of the ultimate goals that a person with impairment might strive to have. Many solutions have been proposed to achieve this goal, but unfortunately most of them are very expensive which makes them unaffordable for most of the people, especially in countries with high poverty rates. External assistance might not be available all the time, because people naturally do not like to feel impaired or to be completely dependent on others[3]. Therefore, our application will help them a lot in being partially dependent on themselves. It is not safe for visually impaired people to navigate on their own because there might be some obstacles or consequences along their way, as there might be stairs or sharp objects in their surroundings, which might put their life in danger. According to European Blind Union blind or partially sighted people shall have a part in society as well as work independently, the People´s Advocate Institution is the only public institu-

tion in Albania that managed in constructing a building that has contributed to create a warm, non-discriminating and barrier-free building for all employees with or without sight. But,what if we created a mobile application that provide that for all people of the world. According to a survey done in 23 May 2019 by Juliana Damasio Oliveira, Rafael H. Bordini on 27 respondents who are visually impaired asking about the receptivity of having a virtual assistance at home and these were their answers in Fig 1.2.

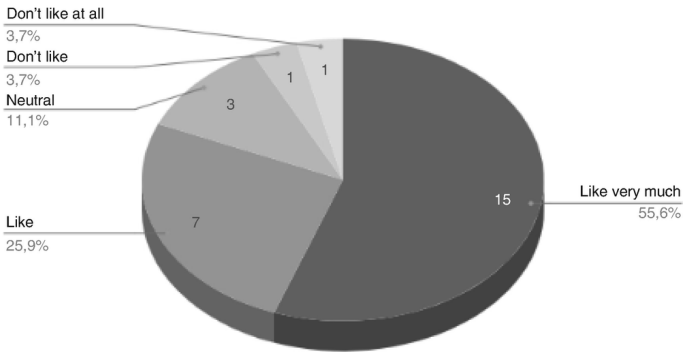

Figure 1.2: survey results on having external assistance

1.1.3 Problem Definition

Navigation of users with any type of impairment could now be done freely and safely even in new places which is the most tackled issue by anyone with vision impairment as well as find their own objects which is something provided in our system using find my object module.

1.2 Project Description

In-Door Assistant Mobile Application Using CNN and TensorFlow.

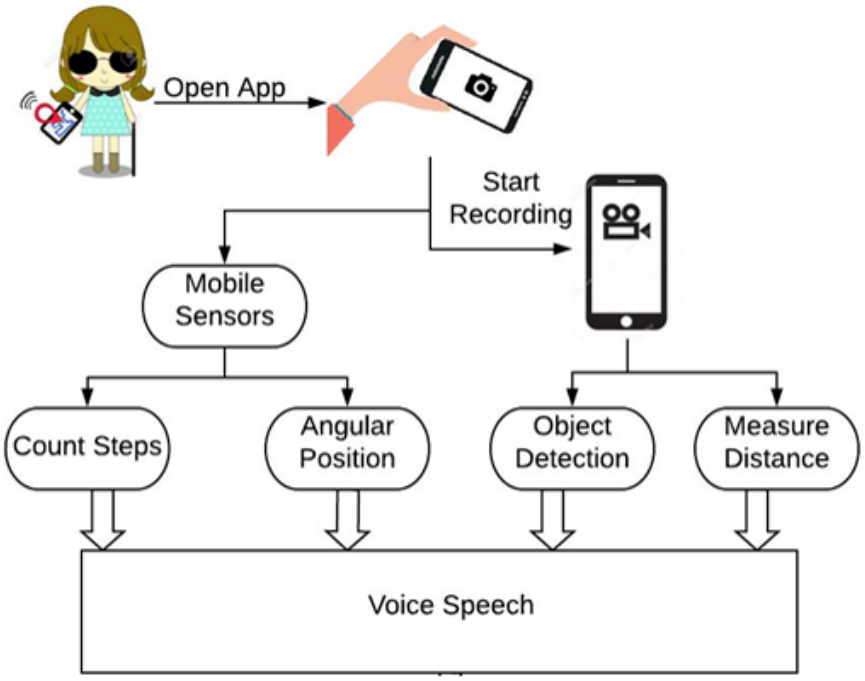

Figure 1.3: Business Context Diagram

1.2.1 Objective

The system Guide-Me aims to enable users with any type of impairment to navigate freely and safely even in new places which is the most tackled issue by anyone with vision impairment as well as find their own objects which is something provided in our system using find my object module simply elaborated in Fig1.3.

1.2.2 Scope

The system discussed in this document targets end users like people with partial or total impairment that would use Guide Me. Users would get audible directions to their destination as well as warnings if there is too near object that they should avoid in addition to that users shall have their own dataset to save their objects which our system will use to find a targeted object if it is asked for. It will also be beneficial and helpful for researchers and developers that may work on the visually impaired assistance application.

1.2.3 Project Overview

In order to provide an accurate assistant for a visually impaired person in a well-lit indoor environment by utilizing a smartphone, we developed a mobile application that uses machine learning and image processing, that will be used for the purpose of identifying the user, collecting data and detecting objects in the user's surroundings and categorizing them into generic household items or obstacles. A software will be developed using python, openCV and DNN to work with the data collected from the smartphone's camera, this software will identify the user using facial recognition, also allow the user to search by speech for his desired item and the software searches for it in the streaming frames and measures the distance to them to provide directions for the user to reach the object, it can also be used for warning the user of incoming obstacles and hazards, the user will also add his own personalized items using the camera with the help of a human assistant or voice instructions. The system overview is shown in Figs. 1.4, 3.1 respectively.

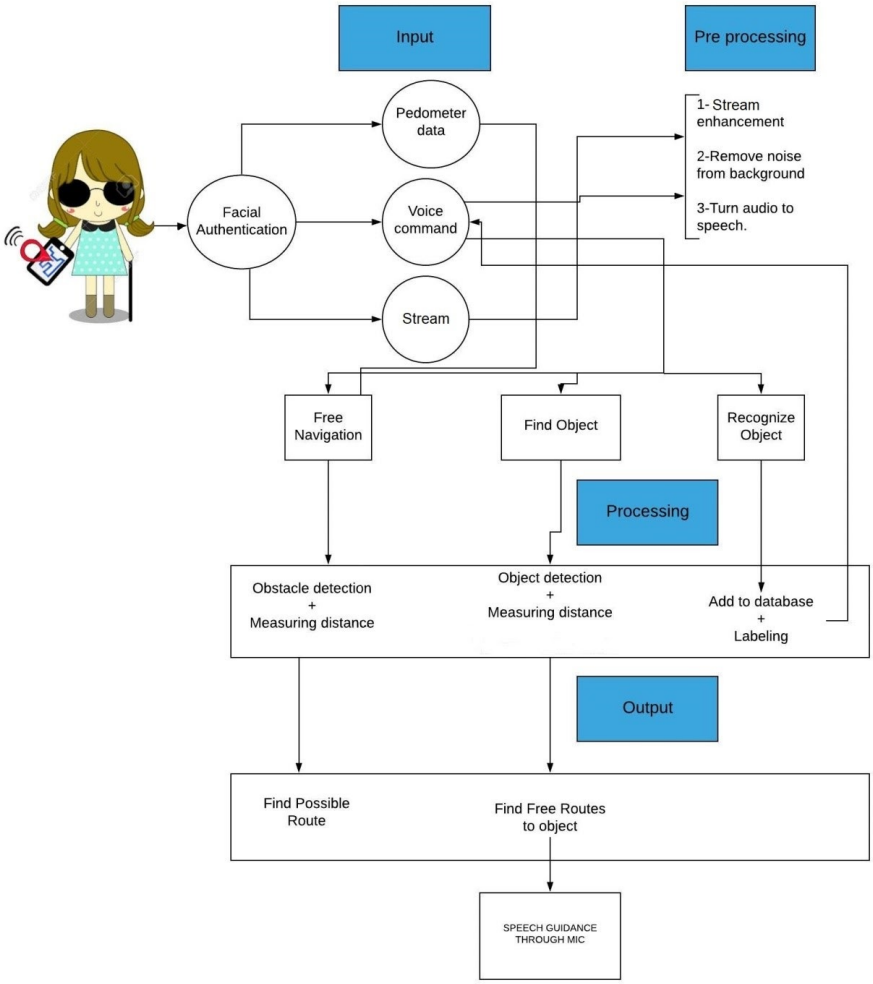

Figure 1.4: System Overview

1.3 Project Management and Deliverable

1.3.1 Tasks and Time Plan

Table 1.1: Tasks and Time Plan

Task	Start date	End date
Idea discussion	25-7-2019	30-8-2019
Idea research	1-9-2019	25-9-2019
Proposal presentation	26-9-2019	10-10-2019
Implementing Prototype	26-9-2019	10-10-2019
Design application	11-10-2019	1-11-2019
Dataset collecting	11-10-2019	1-11-2019
Dataset classification	1-11-2019	11-11-2019
SRS Writing	12-11-2019	12-12-2019
SRS Presentation	12-11-2019	12-12-2019
Implementing Application	13-12-2019	20-1-2020
Implementing Application	13-12-2019	20-1-2020
SDD Writing	25-1-2020	21-2-2020
SDD Presentation	25-1-2020	25-2-2020
Final Prototype implementation	26-2-2020	20-3-2020
Writing Paper	20-3-2020	1-4-2020
Deliver Paper	2-4-2020	2-4-2020
Writing Thesis	2-4-2020	20-5-2020
Final Presentation	24-6-2020	24-6-2020

1.3.2 Budget and Resource costs

Luxand Facial Recognition and the cost of it is 19 Dollar per month.

Chapter 2

Literature Work

2.1 Related works

Several research efforts have been proposed to help visually impaired people. For example, Milios et al. [4] proposed a mobile application that detects objects for the visually impaired. They used a CNN on the images that the user captures, they then used the predefined model of CNNdroid library to classify the objects from the captured image, the accuracy of Typewriter keyboard was (51.89%) and space bar was (47.91%).

The second function that was proposed in this application is bank note recognition, to achieve this goal they chose CraftAR SDK for android. Images of the desired objects are then taken to be recognized and utilized with the CraftAR database and SDK. They finally tested their paper on 10 people using a survey, and the results were good in general and the application was well accepted.

Another work was proposed by Liang-Bi et al.[5]. In this work a wearable glasses is proposed for detecting front objects and a walking stick to guide them through their route, mobile devices application, and on-line information platform to get help in case of collision to the user by sharing their GPS to close friends or family members through mobile device application. Lastly the authors mentioned that in the future, the authors will integrate deep learning techniques[6] for recognizing images and to develop intelligent walking guiding related functions.

This paper is another proposed methodology by Dhruv et al.[7] they used point feature matching to identify common public surroundings. They divided their paper into two objects: The first one is by creating their template of popular street signs(Pharmacy, special

people..etc), Second step is template matching using SURF which detects signs from captured images and relevant points from the template. Eventually, they reached accuracy of 91.67% their problem was variation in color and illumination in captured images or in the template. Their future work includes increasing their accuracy as well as the number of common signs and implementing the same concept in the indoor navigation.

Laviniu et al. [8] developed a paper using smartphone sensors and additional two sensory modules, First one is based on RaspberryPi platform with arm processor and the other one is based on a very popular Arduino platform. In outdoors it uses GPS modules to detect coordinates and move from one point to another then speaks to the user using TTS technology to speak to the user, from the other hand the mobile can recognize user commands using google's voice recognition technology. Inside the building GPS is not available therefor it uses light sensors present in the phone to do indoor navigation, it also uses accelerometer[9] in the phone to detect the fall of the person and help the call someone to help. Tests made show the efficiency of the paper, which can be improved with the development of android based portable devices. Here comes our project main goal which is to achieve navigation and assistance for the visually impaired people only on their mobile phone.

Meanwhile Treephop et al.[10] proposed a healthcare paper on mobile regarding warning paper and obstacle detection for helping visually impaired people. The authors detect objects by a set of processes ordered by obstacle detection of video gathered with CNN execution of darknet library. It is implemented in Python, after that all the scripts will be connected to the Android environment, and start the notification module where videos of a maximum of 32 MB size and not greater than 60 seconds long are analyzed on the server. Distance is measured between the person and the centroid of each detected object by using Pythagorean theorem, and if distance is within threshold the paper will send the alert message to the Firebase paper which sends it directly to the visually impaired person smartphone and wait for response by clicking the "save" button, then the paper will continue the analyzation process else it will send the alert message to the caretaker.

Varsha et al. [11] proposed an android mobile application to detect objects. First the user should capture image of the object then enhancements are formed on the captured image like (sharpening..etc) then the authors used CNN which is found in tensorflow api[12] to detect objects found in the image,the application can work without internet connection as they created their own database which contains 80 classes of objects lastly they used speech synthesis to generate a speech of the detected objects.

Manabu et al.[13] made a mobile application to detect the most common hazards that could face a visually impaired pedestrian. They trained their dataset to detect stairs,bicycle,crosswalk,sidewalk if the application detected any of the mentioned items it should alert the user by vibrating. In order to achieve this, they used tensorflow api and developed their algorithm using CNN, for the experiment in this research , they used 637 images including obstacles such as Stairs, Bicycle, and 393 images of the scenes such as Rail Tracks, Sidewalk, Crosswalk were prepared for training. Lastly the experiment showed an accuracy of 90%.

Finally, Summan et al. [14] proposed an indoor navigation system by applying three main functionalities. First, they used localization as a prerequisite, so that once the user enters a building their current position is fetched and stored in the database. Then, they introduced a user profile where they can store the user's name, gender, height, and most importantly the user's steps length, along with the data from the mobile sensors, such as gyroscope, compass, etc. Finally, path planning is done using the previously mentioned calculated data, along with the building information model, to help guide the user inside the building. The authors tested the system through applying quantitative analysis.

2.2 Comparison with proposed approach

Table 2.1: Comparison with proposed approach

Points of comparison	Method	Accuracy achieved	Objectives	Classifier	sensors
Intelligent Eye	Mobile phone	good in general and the application is well accepted	detecting color,light,objects and banknotes	CNN	embedded light sensor
Smartphone apps of obstacle detection for visually impaired and its evaluation	Mobile phone	90%	detect the most common hazards(stairs,crosswalk,sidewalk)	CNN	ultra sonic sensors
Automated identification of public signs	images taken from the internet	91.67%	detecting common public signs	SURF	sensory system,3d CMOS linear sensor and 6d of inertial sensor
our proposed system	Mobile phone	92%	Guide user to required object,provide safe navigation	CNN	Accelerometer sensor

.

Chapter 3

System Requirements Specification

3.1 Introduction

3.1.1 Purpose of this chapter

The main purpose of this chapter is to outline the requirements for Visually Impaired In-Door Assistant: detect and analyze objects in the surroundings to help user navigate in new rooms freely as well as find user's own objects and give directions to his intended object. This is done with the aid of sensors such as accelerometer and pedometer. This document will provide a detailed overview of our software product's parameters and goals and explain purpose and the features of Visually Impaired In-Door Assistant[15] and describes its interfaces,hardware, software requirements and ex-plains what the system will do. This chapter discusses how our stakeholder, team, and audience see the product and its functionality.

3.1.2 Scope of this chapter

This chapter is the requirements work product that formally specifies Visually Impaired In-Door Assistant. This targets end users like people with partial or total impairment that would use Guide Me. Users would get audible directions to their destination as well as warnings if there is too near object that they should avoid in addition to that users shall have their own dataset to save their objects which our system will use to find a targeted object if it's asked for. It will also be beneficial and helpful for researchers and developers that may work on the visually impaired assistance application.

3.1.3 Overview

The proposed system uses mobile camera to act as the eyes of the blind person, it sends the captured stream to the main model which is object detection with a pretrained dataset of house items, the user then chooses between the system's two main functionalities using speech input by translating it through natural language processing either to safely navigate the room or to look for an item he seeks. If the user chooses safe navigation the objects in the frame are detected and distance to reach them is calculated and the user is notified by speech output if the object is too close to the other and is blocking their path and where he could move to avoid that obstacle. If the users chooses finding objects, he then is prompted to say the objects name and moves his phone to capture a stream with as many frames as possible and if the object is detected in the frame the mobile vibrates meaning that the object is in that direction, and the closer he gets to the object the more intense the vibration becomes, if the object is not found after a certain time period of searching frames the user is notified by speech that the object is not found as shown in figure 1.4.

3.1.4 Business Context

Our work is motivated by both application domain and previous work, our system aims to provide an independent life for anyone with visual impairment as well as provide safe navigation which we aimed to cover in our three modules for example the most tackled problem for visually impaired people as mentioned by Eng. Salama (real life user of our system) is new places navigation. Our system solves this problem by processing the user's captured stream of the room and finding objects in the surrounding area the system then safely navigate the user based on his chosen module whether it is free roaming or it is towards a targeted object as shown in figure 1.3 .

3.2 General Description

3.2.1 Product Functions

1. Authenticate user using facial recognition
2. Speech recognition using natural language processing
3. Scanning frames captured by user for any hazards
4. Search for required objects when asked to.

5. Measure distance between user and surrounding objects to avoid collision.
6. Alert user if there's near objects that he's about to hit.

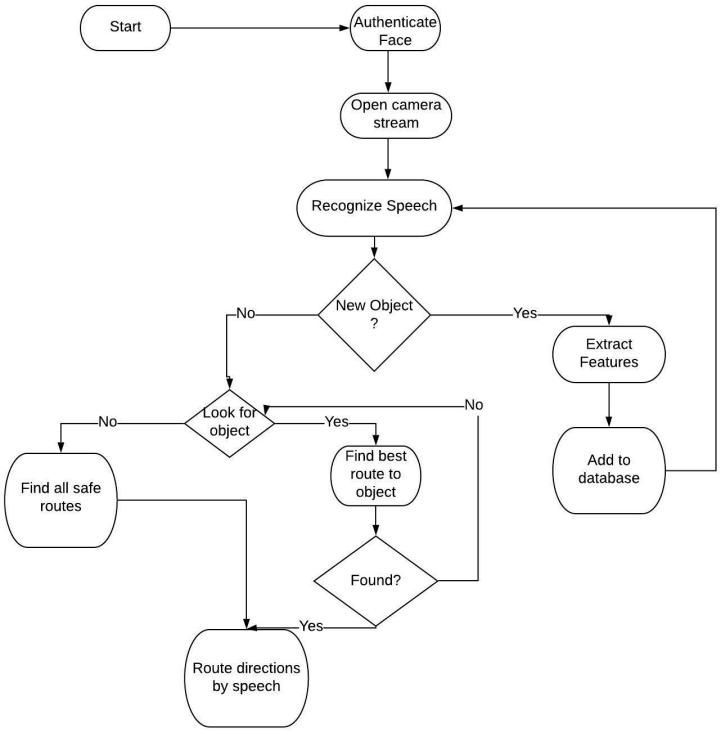

Figure 3.1: Flowchart of our system

3.2.2 Similar System Information

Intelligent Eye: A Mobile Application for Assisting Blind People through four functions.[16]

1. Light detection: Which is done through the embedded light sensor in the phone to read light intensities
2. Color Detection: They obtain the image through the back camera and detect the color using OpenCV library, the RGB color of area touched by user color name is then spoken to the user using text to speech engine available in the smart phone.
3. Object recognition: Allows recognizing objects from images captured by the camera of a mobile device, they developed the activity using CNNdroid library the system then displays top ve results.
4. Banknotes recognition: Enables blind users to identify banknotes through CraftAR SDK for android.

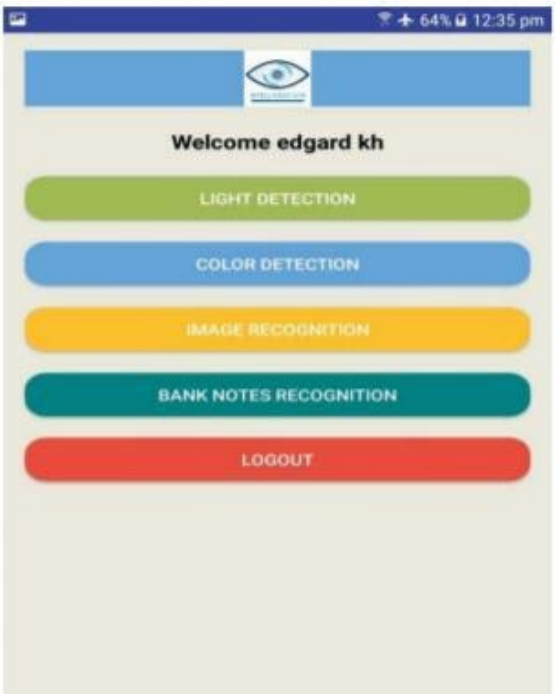

Figure 3.2: Similar System

This section is a proposed methodology for automatic identification of public surroundings. They divided their system into two objects First one is by creating their template of popular street signs(Pharmacy, special people..etc), Second step is template matching using SURF which detects signs from captured images and relevant points from the template. Eventually , they reached accuracy of 91.67 percent their problem was variation in color and illumination in captured image or in the template. Their future work includes increasing their accuracy as well as number of common signs and implementing same concept in the indoor navigation. All of the previously mentioned functions are implemented to facilitate lives of visually impaired people and achieved using only mobile phone which is the same concept we are going to use in our system.

3.2.3 User Characteristics

Our system users are all people who have any type of visual impairment according to California optometric association which are:

1. Loss of Central Vision: The loss of central vision creates a blur or blindspot, but side vision remains intact. People with this type of impairment don't need external help with our system.
2. Loss of Peripheral (Side) Vision: Central vision remains making it possible to see directly ahead. as a result they won't need external help as well.
3. Blurred Vision: Causes both near and far to appear to be out of focus. This type of impairment might need external assistance to de ne personalized items of the user.
4. Generalized Haze: Causes the sensation of a film or glare that may extend over the entire viewing eld. In this case it varies from one person to another.
5. Extreme Light Sensitivity: Washed out image and/or glare disability. this type of impairment need external assistance to capture personalized items of the user.
6. Night Blindness: Inability to see outside at night or in dimly lighted inte-rior areas. In this case they'd need external help if they're setting up the application in insufficient light.

All previously mentioned impairments must have basic knowledge in using Android mobile devices as well as basic knowledge in understanding English navigation commands.

3.2.4 User Problem Statement

Detection and improvement of classification accuracy of surrounding objects as well as measuring distance between user and surrounding objects. So user needs to get a real-time information about the area he's navigating in.

3.2.5 User Objectives

By using Guide-Me, users with any type of impairment could now navigate freely and safely even in new places which is the most tackled issue by anyone with vision impairment as well as find their own objects which is something provided in our system using find my object module.

3.2.6 General Constraints

One of the main constraints of the system is the connection inside any building that the mobile could face. In addition, the position of the mobile must be not tilted in any direction to provide an accurate feedback of the detected objects as well as estimated distance.

3.3 Functional Requirements

3.3.1 Authentication

FR1	
Function	Register
Description	User can register a new account
Input	Video stream of the user's face, Audio Stream for the username
Source	Smartphone's frontal camera and mic
Output	Home page with camera view
Action	Check audio for clear speech of username, Check video input for 10 frames containing the user's face.
Pre-condition	None
Post-condition	The user's data is sent and stored into the database for further processing
Dependencies	None

FR2	
Function	Login
Description	The user can login into his account from any device
Input	Video stream of the user's face
Source	Smartphone frontal camera
Output	Home page with camera view
Action	Check video input for at least one face, use facial recognition model to identify the user.
Pre-condition	The user has an account in the database
Post-condition	The Frame captured is processed and encoded on the server and classified accordingly using facial recognition
Dependencies	F1

FR3	
Function	Facial encoding
Description	The system converts facial features to data.
Input	Image of a face
Source	Smartphone frontal camera
Output	Encodings
Action	Check the image for a face, Facial features are extracted and converted to data which can be used for comparisons.
Pre-condition	None
Post-condition	Encodings are saved on the server along with the the user id so they can be used for recognition later.
Dependencies	None

FR4	
Function	Facial Recognition
Description	This model allows uses Aritifical Intelligence techniques in order to identify the user existing in the frame
Input	Image containing user's face
Source	Smartphone frontal camera
Output	User info if the user exists in the database
Action	Check video input for at least one face, encode the facial features and compare them to existing ones in the database
Pre-condition	The user has an account with facial images available in the database
Post-condition	If the user is valid, he is allowed to access the application and his own customized objects.
Dependencies	F1,F2.f3

3.3.2 All Object Detetcion Model

FR5	
Function	Video capture
Description	User captures video for the room
Input	Video Stream
Source	Smartphone camera
Output	Home page
Action	Video validation model .
Pre-condition	User is logged in ,camera isn't blocked
Post-condition	The user's data is sent and stored into the database for further processing
Dependencies	F1,F2

FR6	
Function	Object Detection
Description	Detect and track objects in the video
Input	Video Stream
Source	database
Output	Labels detected and coordinates
Action	Check frame by frame for objects and recognize them and track their x,y coordinates.
Pre-condition	Video stream contains objects.
Post-condition	Coordinates are used to keep track of where an object resides and sent to the navigation model
Dependencies	F1,F2,F5

3.3.3 Customized Model

FR7	
Function	capture images
Description	users capture 10 images for the object he want to add.
Input	10 images of the object
Source	smartphone camera
Output	homepage
Action	none
Pre-condition	The user is logged in,camera isnt block
Post-condition	the users data is sent and stored into the database for further processing.
Dependencies	F1,F2

FR8	
Function	transform images
Description	transfrom them to a lower scale so the training process is faster.
Input	images
Source	Database
Output	images with lower scale
Action	Transform image with specific width and height
Pre-condition	images sent from database
Post-condition	images with lower scales used to extract features from it .
Dependencies	F1,F2,F7

FR9	
Function	Create bounding box
Description	Detect objects in the photo and draw a boundary box.
Input	images
Source	Server
Output	Dataset with coordinates of the object
Action	Check images contain at least one object and get coordinates from it.
Pre-condition	Images is sent
Post-condition	Dataset are saved so they can be used to generate tf record.
Dependencies	F1,F2,F7,f8

FR10	
Function	Generate dataset records
Description	dataset records(Tensorflow) that can be served as input data for training of the object detector.
Input	dataset
Source	server
Output	dataset record(Tensorflow) files
Action	Check if rows and columns are valid.
Pre-condition	Images is sent
Post-condition	Generate a train.record and a test.record file which can be used to train our object detector.
Dependencies	F1,F2,F7,F8,F9

FR11	
Function	Train model
Description	Object detection model with high accuracy .
Input	dataset records files(Tensorflow records format)
Source	Server
Output	Model with high accuracy
Action	Check if tfrecord file is exist.
Pre-condition	TFrecords file is sent
Post-condition	Model are saved with the user id so they can be used for customized object detection later.
Dependencies	F1,F2,F7,F8,F9,f10

3.3.4 Distance

FR14	
Function	Navigation
Description	The system finds the route for the desired item.
Input	Localization matrix
Source	Room localization module
Output	Path
Action	We use Algorithms to parse the localization matrix for the shortest route, check if the route is free
Pre-condition	Localization matrix is available.
Post-condition	The system sends the route to the voice directions module.
Dependencies	F1,F2,F5,F12

3.3.5 Video Assistance

FR15	
Function	Video validation
Description	The system checks that the entire surroundings are captured in the video
Input	Video Stream, accelerometer data
Source	Smartphone camera, accelerometer sensor
Output	Alert if the video is invalid, alert on video completion
Action	Check that the phone has rotated around itself
Pre-condition	The user has choosen a menu item, and is asked to capture his surroundings
Post-condition	The video is sent to the server for further processing and feature extraction
Dependencies	F1,F2,F5

3.3.6 Audio and Voice Assistance

FR16	
Function	Audio Menu
Description	The system presents menu options using speech.
Input	None
Source	Smartphone speaker
Output	Audio
Action	Convert text to audible speech
Pre-condition	The user is logged in
Post-condition	The system awaits audio command from the user.
Dependencies	F1,F2

FR17	
Function	Audio choice
Description	The system converts audio speech to text
Input	Audio
Source	Smartphone Mic
Output	Encodings
Action	Check audio for speech, look for desired item in menu.
Pre-condition	The system has presented the voice menu.
Post-condition	The user is redirected to his choosen option's page
Dependencies	F1,F2,F14

FR18	
Function	Voice Directions
Description	The system converts the path into audible directions.
Input	Path
Source	Navigation module.
Output	Audio
Action	The path is prcoessed and turn into directions where the user can move to reach his item.
Pre-condition	Path is generated
Post-condition	None
Dependencies	F1,F2,F3,F4,F5,F6,F7,F8,F9,F10,F11

3.4 Interface Requirements

3.4.1 User Interfaces

3.4.1.1 GUI

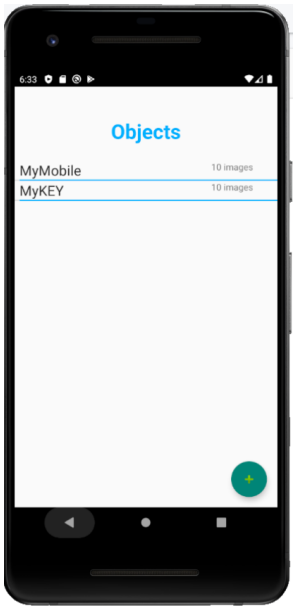

Figure 3.3: user add a new object

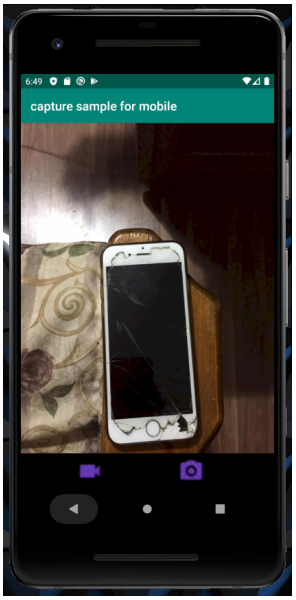

Figure 3.4: caption a video for the object

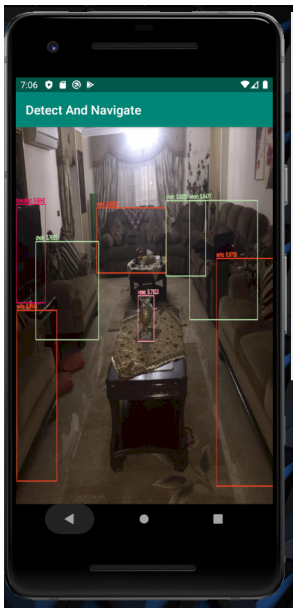

Figure 3.5: detect and navigate

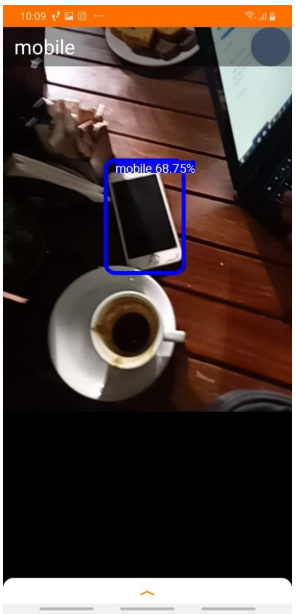

Figure 3.6: detect a certain object

3.4.1.2 API

1. Tensorflow Lite: used for machine learning applications such as neural networks, will be used in our object detection models.
2. Luxand Facial Recognition: used for processing and manipulating faces for user authentication.
3. Google Assistant: used for capturing the user's speech and converting it to text, and converting text to speech.

3.4.2 Communications Interfaces

The communication interface is one of the most important requirements of our software as it will need a connection to the internet or a local host connection.

3.5 Performance Requirements

Requires real time server response with ability to process large numbers of frames, For model training, the system must be able to handle large training datasets to ensure model accuracy[17].

3.6 Design Constraints

The room where the user is located must be well lit. The size of the room that can be captured is directly proportional to the quality of the smartphone's camera.

3.6.1 Standards Compliance

Minimum android SDK 19.

3.6.2 Hardware Limitations

1. Mobile must have access to a camera and is compatible with camera API version 2.
2. Surrounding room must be well lit.
3. Quality of results is directly proportional to camera quality and memory.
4. Phone must be connected to the internet.

3.7 Other non-functional attributes

3.7.1 Performance and Speed

The Guide-Me system must be interactive and the delays involved must be reduced. So in every action response of the Guide-Me system. Detection and classification must have no delays.

3.7.2 Reliability

The Guide-Me is reliable. It must be make sure that the system is reliable in its operations. This would be mainly focusing on the detection and classification. As sensors readings should be accurate and error free. When room is scanned all objects detected are being classified, it is very important to identify the type of the object as well as personalized objects correctly with no errors and the user should be able to trust the Guide-Me system fail rate is almost 0 percent.

3.7.3 Maintainability

Guide-Me system could be improved by different developers so its maintainability should be easy by documenting the code and the design as we implemented them using design patterns as MVC design pattern to manage the code to model, view and controller and at the the end link with the cloud, single tone for database connection and observer for notifications .

3.7.4 Usability

The user should use our application and interact with it using speech, and also be able to reach their object easily by guiding him through good implemented design that guide him the easiest way to process. Proportion of functionalities or tasks implemented does not need time to be learned. Also, this system is easy to be memorized due to the small number of tasks the user will do.

3.8 Preliminary Object-Oriented Domain Analysis

3.8.1 Class Diagram

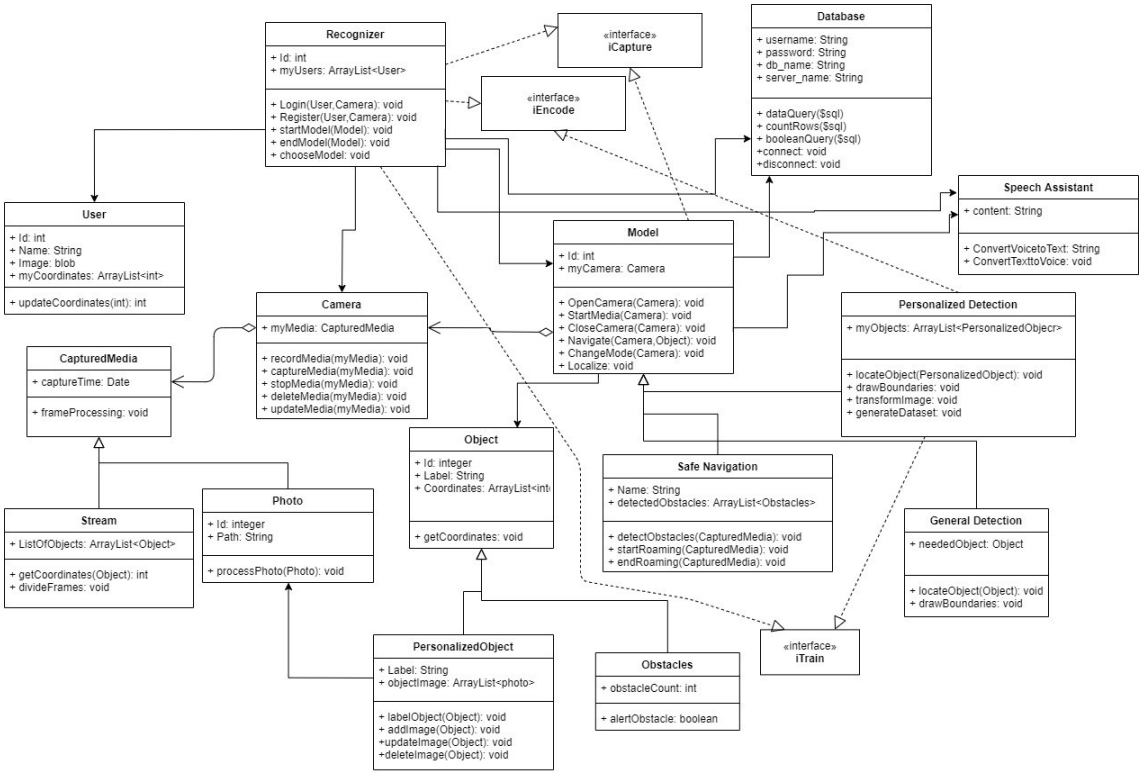

Figure 3.7: Class diagram

3.8.2 Database Diagram

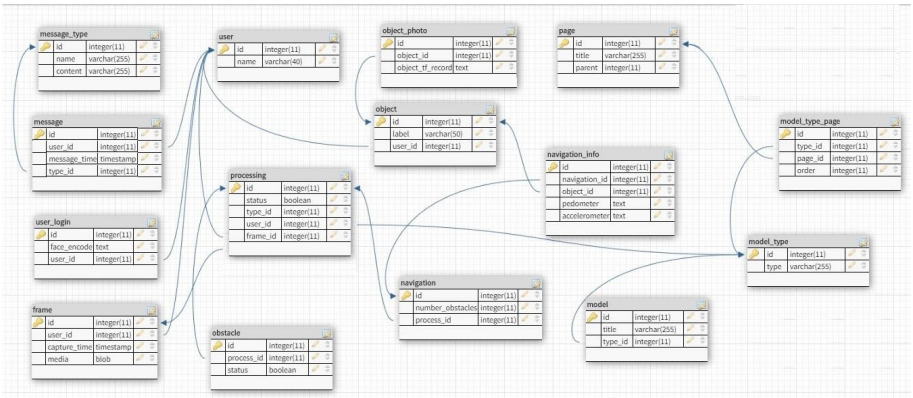

Figure 3.8: Database

3.8.3 Context Diagram

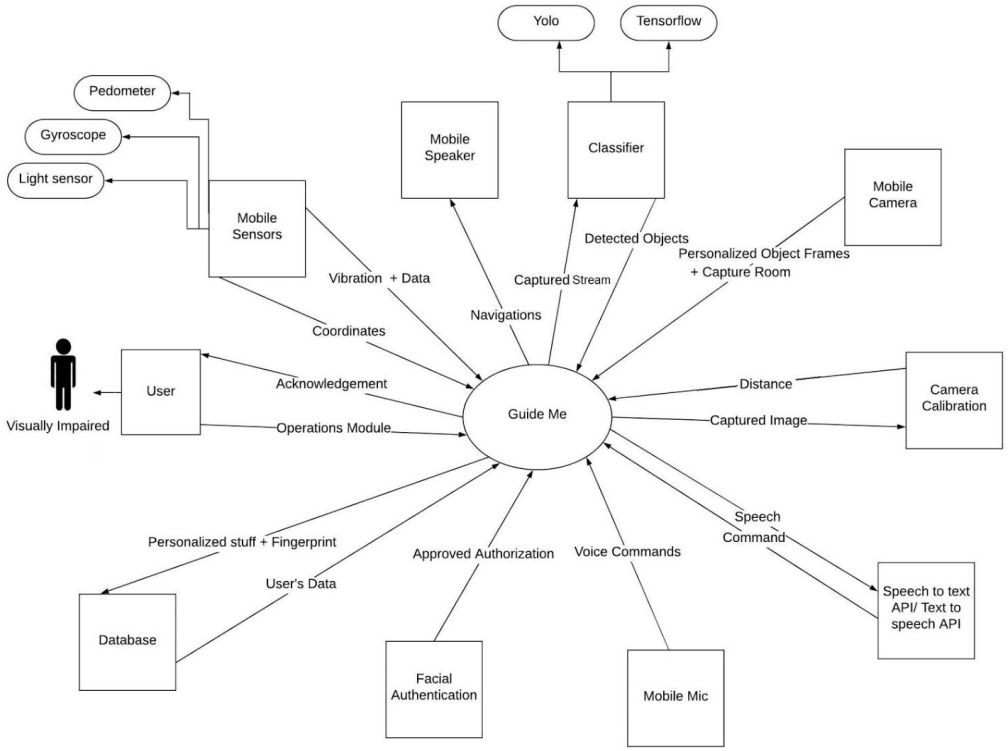

Figure 3.9: Context Diagram

3.8.4 Block Diagram

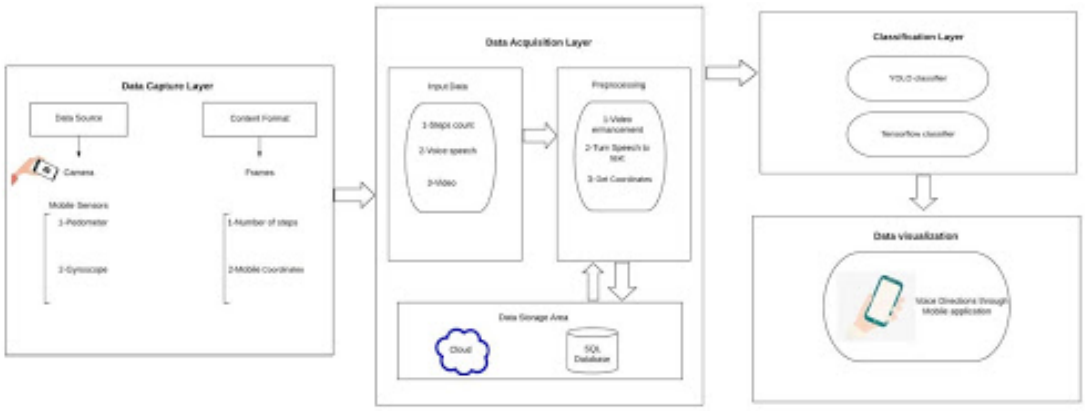

Figure 3.10: Block Diagram

3.9 Operational Scenarios

3.9.1 Scenarios

3.9.1.1 Scenario 1: Capture room process

The visually impaired person starts the application and whether his module is free roaming or find a specific object he captures the whole room and the application starts to send data to the system and analyze data to detect the room objects as well as calculate distance between user and these objects.

3.9.1.2 Scenario 2: Customized dataset handling

The visually impaired person is the one who manipulates the customized dataset which includes

1. Add object to personalized dataset.
2. Delete object from personalized dataset.
3. Edit object from personalized dataset.

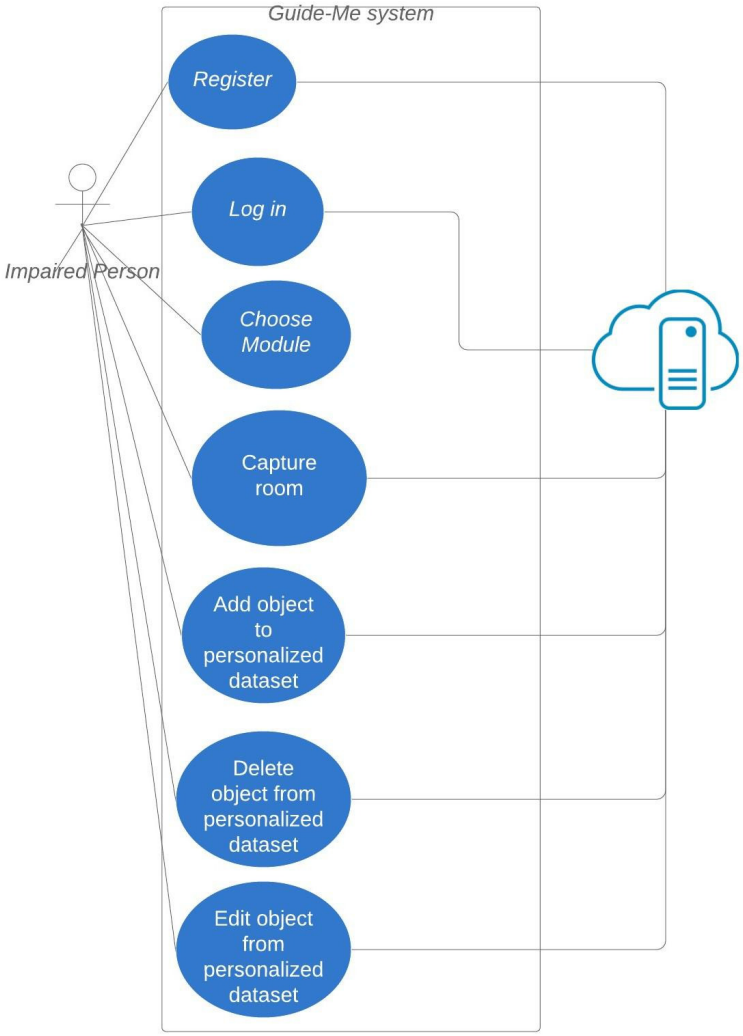

Figure 3.11: Use Case Diagram

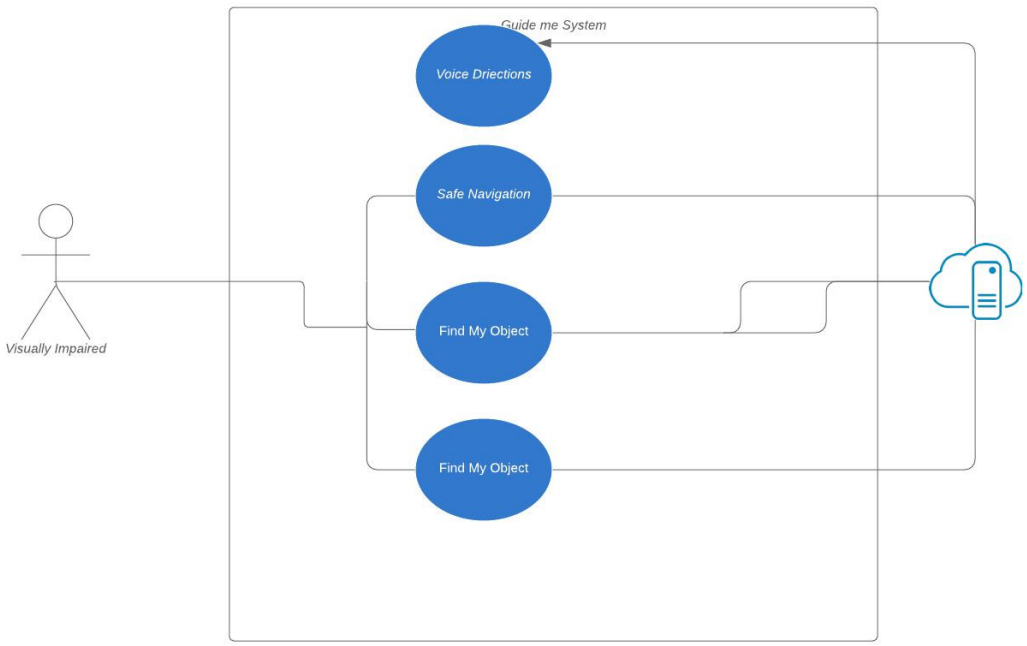

Figure 3.12: Use Case Diagram 1

Chapter 4

Software Design Document

4.1 Introduction

4.1.1 Purpose

This software design document purpose is to fully describe the architecture of our Visually Impaired In-Door Assistant: system. Our system depends on people with visual impairments using their smartphones as their eyes to navigate their surroundings. This document will explain in details, the components of the system represented in the block diagram, the ow of the project with sequence diagram, the data handling in the ER diagram also the implementation of the project and its development will be shown in the class diagram. This software design document is intended for the stakeholders and developers of our system. This document is also presented as a part of a graduation project at Misr International University.

4.1.2 Scope

The system discussed in this document targets end users like people with partial or total impairment that would use Guide Me. Users would get audible directions to their destination as well as warnings if there is too near object that they should avoid in addition to that users shall have their own dataset to save their objects which our system will use to find a targeted object if it's asked for. It will also be beneficial and helpful for researchers and developers that may work on the visually impaired assistance application.

4.1.3 Overview

The proposed system uses mobile camera to act as the eyes of the blind person, it sends, the system identifies the user using facial identification and retrieves his personalized items from the database if he has any, then a camera stream is opened, the captured stream to the main model which is object detection using a TenorFlow model with a pre-trained dataset of house items, the user then chooses between the system's two main functionalities using speech by translating it through STT either to safely navigate the room or to look for an item he seeks. If the user chooses safe navigation the objects in the frame are detected and distance to reach them is calculated and the user is notified by speech output using a TTS tool[18] if the object is too close to the other and is blocking their path and where he could move to avoid that obstacle. If the user chooses finding objects, he then is prompted to say the objects name and moves his phone to capture a stream with as many frames as possible and if the object is detected in the frame the mobile vibrates meaning that the object is in that direction, and the closer he gets to the object the more intense the vibration becomes, if the object is not found after a certain time period of searching frames the user is notified by speech that the object is not found. The user can also add his personal items using a video stream of the item and his audio input as a label for it, either through voice assistance or with the help of a human assistant.

4.2 System Overview

In order to provide an accurate assistant for a visually impaired person in a well-lit indoor environment by utilizing a smartphone, we developed a mobile application that uses machine learning and image processing, that will be used for the purpose of identifying the user, collecting data and detecting objects in the user's surroundings and categorizing them into generic household items or obstacles. A software will be developed using python, openCV and Deep Neural Networks to work with the data collected from the smartphone's camera, this software will identify the user using facial recognition, also allow the user to search by speech for his desired item and the software searches for it in the streaming frames and measures the distance to them to provide directions for the user to reach the object, it can also be used for warning the user of incoming obstacles and hazards, the user will

also add his own personalized items using the camera with the help of a human assistant or voice instructions as shown in Fig 1.4.

4.3 System Architecture

4.3.1 Architectural Design

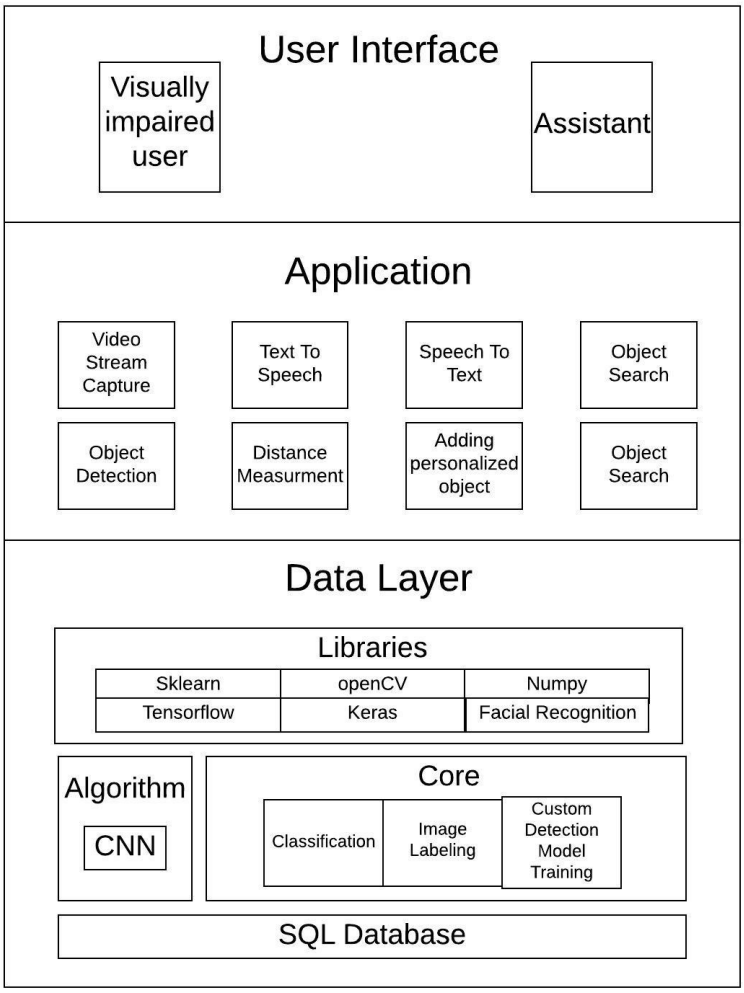

Figure 4.1: Architectural Design

4.3.2 Decomposition Description

4.3.2.1 Class Diagram

As shown in Fig 3.7

4.3.2.2 Recognizer

Class name	Recognizer
Super class	None
Sub class	None
Purpose	This class controls users and models; it chooses the specified model for detection and connects it with its user
Collaborations	This class is aggregated by Model class and aggregates CapturedMedia class and it assists Recognizer class
Attributes	id, array list of user
Operations	login (user, camera), register(user, camera), start model(model), end model (model), choose model

4.3.2.3 User

Class name	User
Super class	None
Sub class	None
Purpose	This class is responsible for handling user information and contains user's face credentials
Collaborations	This class is aggregated by Recognizer class
Attributes	id, name, image, array list of my coordinates
Operations	update coordinates (int)

4.3.2.4 Camera

Class name	Camera
Super class	None
Sub class	None
Purpose	This class allows user to use mobile camera and capture media for processing
Collaborations	This class is aggregated by Model class and aggregates CapturedMedia class and it assists Recognizer class
Attributes	Media Data
Operations	record media (my media) , Capture media (my media), Stop media (my media), Delete media (my media), Update media (my media)

4.3.2.5 Model

Class name	Model
Super class	None
Sub class	SafeNavigation, GeneralDetection, PersonalizedDetection
Purpose	The purpose of this class is to allow user to detect object and navigate to object
Collaborations	This class assists Recognizer class and it aggregates Camera and Object classes
Attributes	id, camera
Operations	open camera (camera), Start media (camera), Close media (camera), Navigate (camera, object), Change mode (camera), Localize ()

4.3.2.6 Database

Class name	Database
Super class	None
Sub class	None
Purpose	This class purpose is to send queries and execute it in the database and retrieve information
Collaborations	This class assists the Recognizer class
Attributes	username, password. Db $_name, server_name$
Operations	Data query (), Count rows (), Boolean query(), Connect(), Disconnect().

4.3.2.7 Speech Assistant

Class name	Speech Assistant
Super class	None
Sub class	None
Purpose	This class allows the application to convert user's voice commands into text and convert text to voice instructions
Collaborations	This class assists the Recognizer class
Attributes	$text_content$
Operations	ConvertVoiceToText(), ConvertTextToVoice()

4.3.2.8 Personalized Detection

Class name	Personalized Model
Super class	Model
Sub class	None
Purpose	This class allows the user to detect personalized objects, it trains the detection model and connects the user to his customized model
Collaborations	This class inherets the Model class and is assisted by Stream class
Attributes	array list of personalized objects
Operations	locate object personalized object(), Draw boundaries(), TransformImage(), GenerateDataSet ()

4.3.2.9 Object

Class name	Object
Super class	None
Sub class	Obstacles,Personalized Object
Purpose	This class purpose is to allow model to differentiate between different objects after detection by giving labels and getting object's coordinates
Collaborations	This class is aggregated by Model class
Attributes	id, label and array l ist of coordinates
Operations	getCoordinates()

4.3.2.10 Obstacles

Class name	Obstacles
Super class	Object
Sub class	None
Purpose	This class allows the model to differntiate between objects and obstacles after measuring distance to user
Collaborations	This class inherets object class
Attributes	obstacleCount
Operations	AlertObstacles()

4.3.2.11 Personalized Object

Class name	Personalized Object
Super class	Object
Sub class	None
Purpose	This class purpose is to allow user to add photos of new object and give it a label so it can be processed by custom detection model
Collaborations	This class inherits object class
Attributes	label, object image
Operations	labelObject (object), AddImage(object), UpdateImage(object), DeleteImage(object)

4.3.2.12 General Detection

Class name	General Detection
Super class	Model
Sub class	None
Purpose	This class is responsible for general object detection; it detects objects in stream, draws a boundary box around the object and measures distance to object.
Collaborations	This class inherits model class
Attributes	NeededObject
Operations	LocateObjects(),DrawBoundaries()

4.3.2.13 Photo

Class name	Photo
Super class	CapturedMedia
Sub class	None
Purpose	This class allows the user to capture face image for login and registration and allows user to capture object image for custom detection
Collaborations	This class is aggregated by PersonalizedObject class and User class
Attributes	id, path
Operations	process photo()

4.3.2.14 Stream

Class name	Stream
Super class	CapturedMedia
Sub class	None
Purpose	The purpose of this class is to capture a stream and send it to the model for object detection processing
Collaborations	This class assists the GeneralDetection class
Attributes	array list of object
Operations	GetCoordinates(Object) ,DivideFrames()

4.3.2.15 Captured Media

Class name	Captured Media
Super class	None
Sub class	Stream, Photo
Purpose	The purpose of this class is to allow user to capture media in different types to allow detection
Collaborations	This class is aggregated by Camera class
Attributes	CaptureTime
Operations	FrameProccessing()

4.3.2.16 Sequence Diagrams

1. Registration Sequence

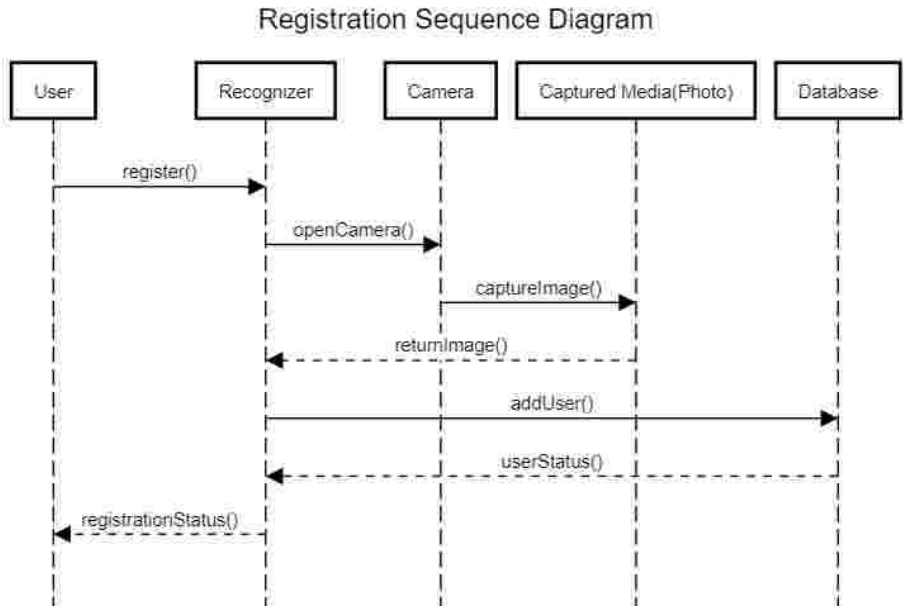

Figure 4.2: Registration Sequence Diagram

2. Adding Personalized Object Sequence

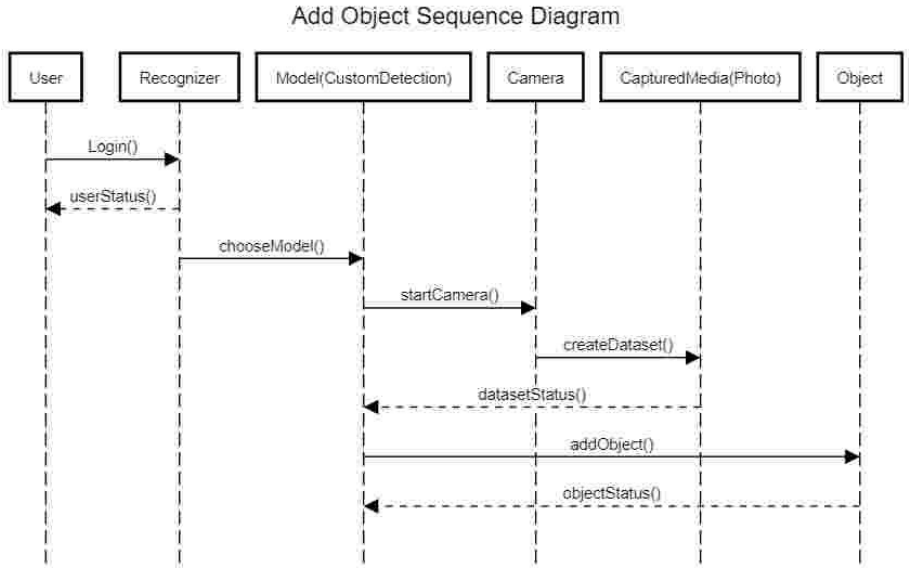

Figure 4.3: Adding Object Sequence Diagram

3. Safe Navigation Sequence

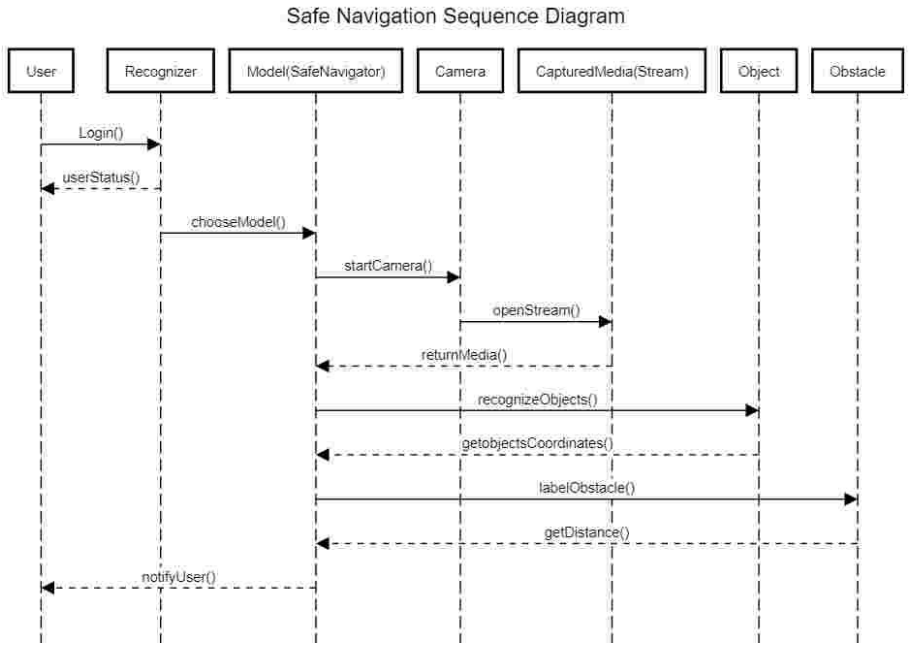

Figure 4.4: Safe Navigation Sequence Diagram

4. Train Customized Model Sequence

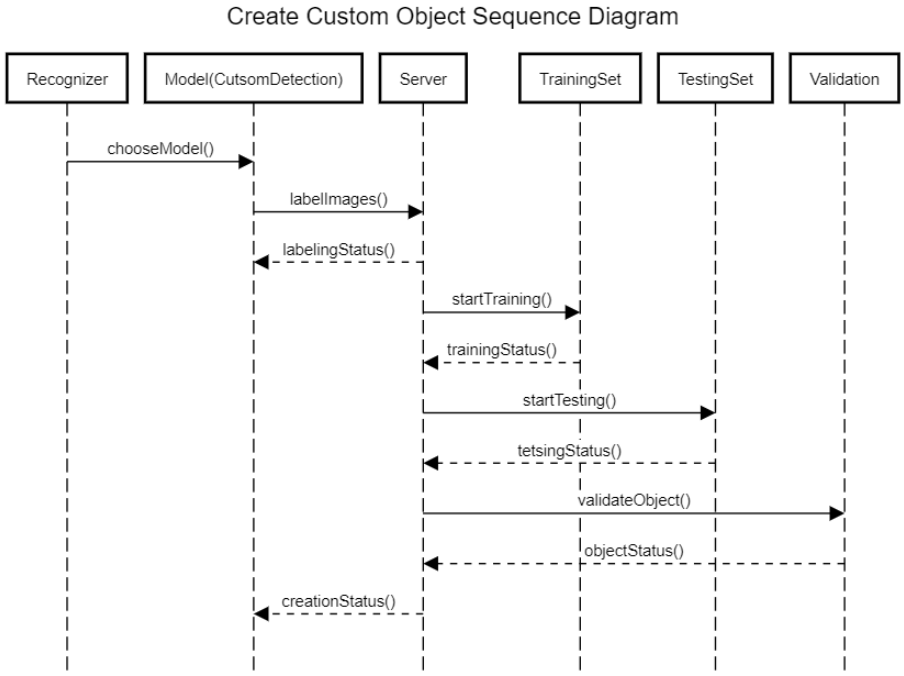

Figure 4.5: Training on Customized Object Sequence Diagram

4.3.3 Process Diagram

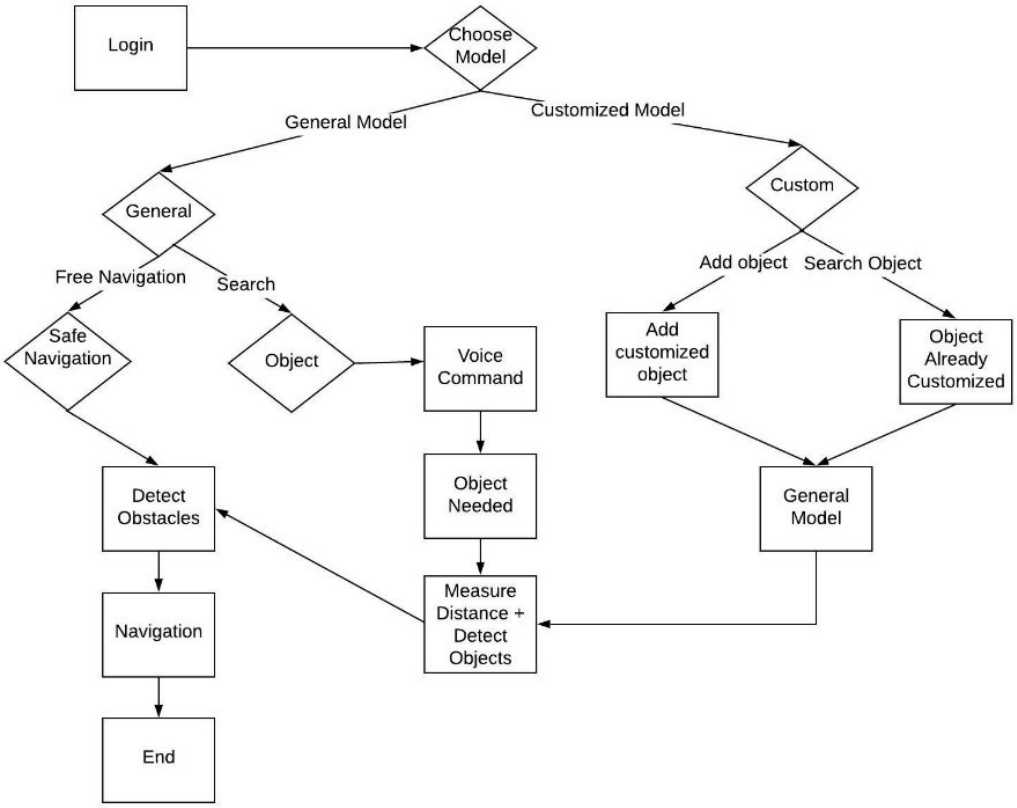

Figure 4.6: Process Diagram

4.3.4 Design Rationale

1. N-Layered Architecture:

   This architecture[19] has been chosen to develop our system as it separates the logic of the layer from the presentation of it which makes each layer independent from the others and as a result a change in a layer will not change in the rest of the layers which makes it easier to implement and to change.

2. Convolutional Neural Networks:

   CNN consists of several different layers[20] such as the input layer, at least one hidden layer, and an output layer. They are best used in object detection for recognizing patterns such as edges (vertical/horizontal), shapes, colors, and textures. Therefore, why we chose it in our project to apply along with TensorFlow for optimal results in object detection

4.4 Data Design

4.4.1 Data Description

As shown in fig 3.8

4.5 Component Design

4.5.1 Dataset

1. Luxand Dataset
   Luxand FaceSDK[21] returns the coordinates of all human faces that appear in the picture or notifies if no face is found. FaceSDK can track all the faces appearing in a video stream. It also allows finding out if a newface appears in the frame, or if one of the subjects leaves the frame. This in turn enables easy implementation of people counting.
2. COCO Dataset
   Common Objects in Context (Coco) dataset[22], which contains around 123,287 images, is a large-scale object dataset made by TensorFlow to detect common objects. This dataset is used with all of the suggested modules.
3. Customized Dataset
   The following dataset items vary according to each user, since that it is made of the objects that the user wants to recognize the user's assistant takes 30 seconds video of the desired object from different angles, the system then transforms them to (300*300) width and height coordinates, in order to start the feature extraction process. Finally, the dataset record is generated and linked with the user id, which was previously generated from the face authentication module.

4.5.2 Data Preprocessing

1. Audio To Speech Conversion
   We used Google Speech-to-Text which enables us to convert audio to text by applying powerful neural network models in an easy-to-use API. The API recognizes 120 languages and variants to support global user base.

2. Customized dataset creation

   The user takes a 30 seconds video for the object he/she wants to add the video is then sent to the server and divided into frames to be saved as a collection of images, then these images are resized to (300*300) width and height coordinates, then each frame is labelled

4.5.3 Processing and Classification

1. Face Authentication (Model 1):

   The user's face is scanned when he signs up for the first time, each face has a unique id which will be saved in database whenever the user adds new object in the customized dataset explained below in model3, these data will be retrieved with id generated from the authentication module.

2. General Object Detection (Model 2):

   The model is trained on the coco dataset using quantized mobilenet sdd the dataset used contains only indoor objects. When an image is subsequently pro-vided to the model, it'll yield a list of the objects it identifies, the area of a bounding box that contains each object, and a score that shows the certainty that discovery was rectified.

3. Customized Model (Model 3):

   After labelling the dataset is then split into 80 percent for training, and the other 20 percent are for testing. Then the TFRecords are created that can be served as input data for training of the object detector, so it is needed to extract features of the object from each image and convert it to a TFRecord. Finally, everything is in place and ready to train the model using quantized mobilenet sdd classifier. To make this model run on a mobile, it is important to start by creating a TensorFlow frozen graph that can be used with TensorFlow lite, then convert the frozen graph to the to the TensorFlow Lite flat buffer format, and finally save it in the database to be ready for use along with user id.

4. Distance Measuring Navigation (Model 4):

   In the process of object detection, the width of each detected object is collected, the distance between the user and the object is then calculated and detected

through utilizing the triangle similarity. The formula for measuring the distance is D = (W x F)/P where D is the distance to the object, F is the focal length of the camera, W is the real width of the object and P is the collected width of the object in pixels, The model calculates the distance between the user and the other detected objects, if object pixel width is more than 80 percent of screen size, the system will notify the user by producing a sound alert to avoid a crash event.

4.6 Human Interface Design

4.6.1 Overview of User Interface

The user interactions with the system will be authorized using facial recognition, and his choices will be captured through audio input and the system will present his results using audio output, thus resulting in an easy and convenient way for the visually impaired population to use this application, although the personalization module can be used by a human assistant using buttons if needed.

4.6.2 Screen Images

1. General Object Detection

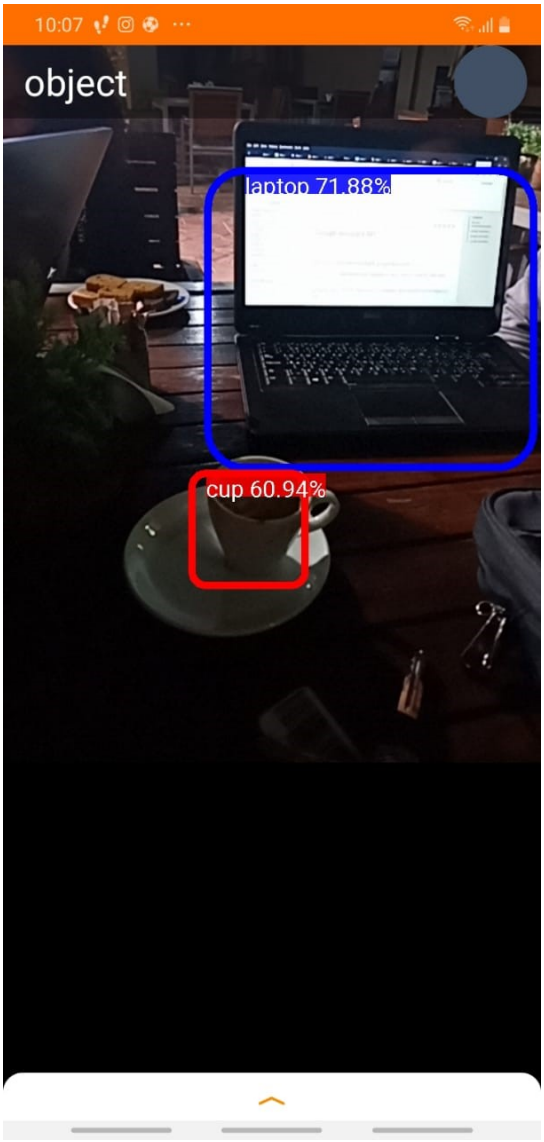

Figure 4.7: General Object Detection

2. Adding Objects

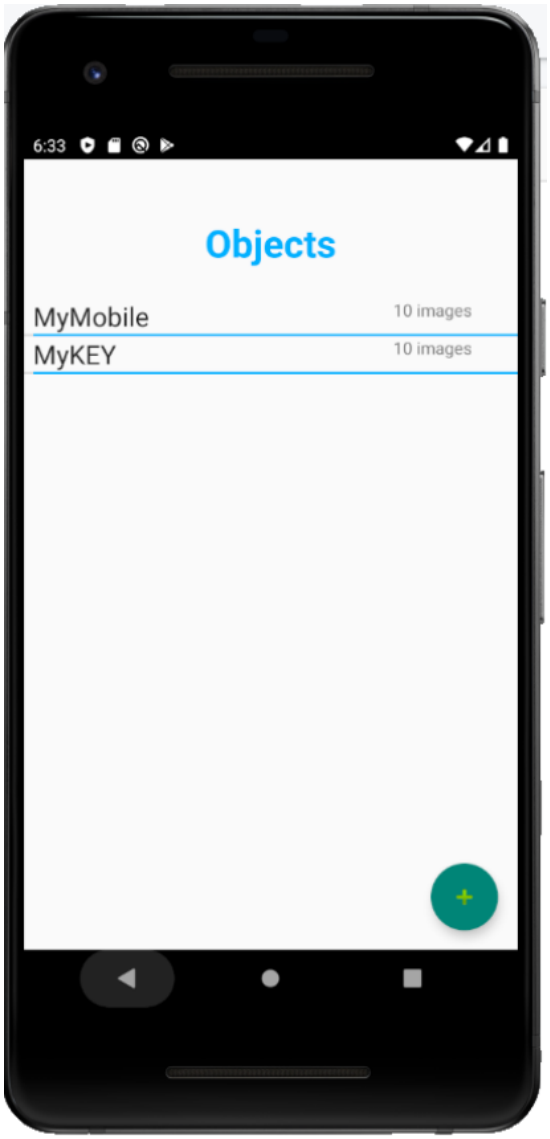

Figure 4.8: Adding Personal Object

The main interface in our application is a regular camera view, a stream from the front camera is used for facial identification for the user, The system presents the options available at the screen using audio when the user touches any part of the screen, The navigation module uses the back camera to capture a stream of the surroundings, and the Adding objects screen can either be interacted with using audio if the user is the one adding his customized item or through buttons if the user is getting human assistance.

4.7 Requirements Matrix

Req. ID	Req. Type	Req. Name	Req. Description	Module	Statues	Req. Reference
Fr1	Required	Register	User can register a new account	User	Completed	Sequence diagram
Fr2	Required	Login	The user can login into his account from any device	User	Completed	Sequence diagram
Fr3	Required	Facial features extraction	The system converts facial features to data.	System	Completed	Sequence diagram
Fr4	Required	Live stream	User open a stream for detection	User	Completed	Class Diagram
Fr5	Required	Object detection	Detect and track objects in the stream	System	Completed	Database Diagram
Fr6	Required	Capture images	User captures 10 images for the object he want to add.	User	Completed	Sequence diagram

Req. ID	Req. Type	Req. Name	Req. Description	Module	Statues	Req. Reference
Fr7	Required	Create bounding box	Detect objects in the photo and draw a boundary box.	System	Completed	Class Diagram
Fr8	Required	Generate dataset records	Dataset records (Tensorflow) that can be served as input data for training of the object detector.	System	Completed	Database Diagram
Fr9	Required	Train model	Train user's customized model with customized object.	System	Completed	Sequence diagram
Fr10	Required	User Positioning	The system tracks the user's location compared to the object	System	Completed	Class Diagram
Fr11	Required	Audio Menu	The system presents menu options using speech.	System	Completed	Class Diagram
Fr12	Required	Voice commands	The system converts audio speech to text	System	Completed	Class Diagram
Fr13	Required	Navigation	The system finds and converts the path into audible directions.	System	In Progress	Sequence Diagram

Figure 4.9: Requirements Matrix

Chapter 5

Evaluation

5.1 Introduction

The proposed system approach utilizes the smartphone's camera to recognize the user's face using the facial identification module[23] as an authentication step, also to capture real-time video stream, then perform object detection and distance calculation, the microphone is used to capture the user's audio command using the speech to text module, when the needed object is located, navigation directions are generated and the text to speech module is used to give the user audible directions. If the user is in free roam mode, only alerts are triggered in case the user will meet an obstacle shortly. The approach allows searching for personalized items, that the user can add with the help of a human assistant by capturing a video for this object, and the images are processed and labeled to train a new model in the custom object detection module. A flowchart of the approach is shown in the following Figure 3.1 .

5.2 Experiments

5.2.1 Classification

1. General Object Detection (Model 1)

   To build a CNN-based model[24, 25], it is necessary to prepare huge amount of image data for object to be classified to achieve high accuracy, so the proposed general object detection approach uses a pre-trained TensorFlow Lite object classification model for object detection[26], The model is trained on the coco dataset using quantized Mobilenet SSD[27] the dataset used contains only indoor objects such as mobiles, chairs and tables. The model has been trained to detect 80 object categories such as mobiles, people, and chairs. When passed an image, it will output a set of detection results which are discussed in the processing section below, this model crops images to size 300x300 pixels using image transformation with RGB values for each pixel, and returns the location of the detected object, the classification of that object from the available labels and the confidence score that the class was detected successfully and the number of objects in the image, in this approach this model uses video stream from smartphone camera as input.

2. Customized Model (Model 2)

   The purpose from this model is to allow users to add their objects to be detected to differentiate between two objects of the same category and to allow users to add objects of new categories. To build a new object dataset the user takes a 30 seconds video which is then divided into 600 frames captured using a smartphone's camera in a well-lit indoor environment. The dataset of each object captured frames is saved as a collection of images. Our training process make use of transfer learning which is the usage of an already trained model to train on your data. This makes the training process taking less time and usually producing better results. For this model we will use Single Shot Detector(SSD)[28] with MobileNet (model optimized for inference on mobile)[29] pre-trained on COCO dataset called SSD MobileNet v2 quantized COCO ,also faster R-CNN is better in the accuracy and more faster, but we cannot use it as cannot be converted to tflite format. To make this TensorFlow model running on a mobile, it is important to start by creating a TensorFlow

frozen graph that can be used with TensorFlow lite, then converting the frozen graph to the TensorFlow Lite flatbuffer format which allows the creation of the TensorFlow Lite model which is then assigned to the user after the validation process is a success. The personalized model is then assigned to the user and saved in the database along with the label map so that the user can call the model to detect the object trained whenever needed.

5.2.2 Navigation

This proposed module is built upon the two previous models after the user specifies the needed object and it is detected this module takes place.

1. Distance Measurement

   There are three approaches when calculating distance listed as follows:

   (a) Using Trigonometric Functions based on Elevation angle

   (b) Using Mobile Accelerometer data

   (c) Using Triangle Similarity

   The first two approaches depend on the mobile sensors and the angle of which the mobile is hold which results in overhead processing and change of distance value due to minimal movement of hand. The visually impaired person is not guaranteed to hold the phone steadily, so we decided to use the third approach based on Triangle Similarity[30] which depends on simple math and minimal camera calibration[31] to increase process speed. The triangle similarity proves that both real time triangle and the captured triangle between the user and the object are similar and since distance is directly proportional to the pixel width of the object adding the focal length of the camera[32] to the equation helps us deduce the formula for measuring the distance as stated by the equation

   $$D = (W \times F)/P$$

   where D is the distance to the object, F is the focal length of the camera, W is the real width of the object saved in a HashMap[33] containing average width of each object and P is the collected width of the object in pixels. The distance is measured in inch and converted into feet to be able to tell the user the number of steps needed to reach object. We calculate the focal length of the smartphone camera used and to get the pixel width of the object we used

the width of the recognition bounding box.

2. Obstacle Detection

   To label objects as obstacles we used simple math approach as well after objects are detected and distance is measured to each object the specified object is labeled as needed object while other objects that their distance is less than the threshold which is equal to the distance covered by one human step or if the object area covers more than 80% of the camera's screen area and the object is labeled as an obstacle.

3. Navigation

   The purpose of this module is to safely navigate the user through his/her way to finding their object and it also gives the user an option to safely navigate through the room without a destination. The model also identifies the specified object. If there are multiple objects of the specified type, the user will be navigated to the nearest one. The model will then identify the coordinates of the bounding box for that object and identify its position. And after distance is measured the model will pronounce the direction for the user to take. The model will keep track of the distance, and once it reaches the threshold the user will be notified that the navigation process is finished. If the distance increases between the user and the object, or the object is no longer detected, the user will be notified with a sound message. This model will also navigate the user away from obstacles by pronouncing a direction that contains no objects labeled as obstacles[34].

5.3 Results

5.3.1 General Object Detection

TensorFlow Object Detection API can be used with different pre-trained models. In this work, a SSD model with MobileNet (SSD MobileNet v1 COCO) was chosen[35]. The model had been trained using COCO dataset which consists of 2.5 million labeled instances in 328 000 images, containing 91 object types such as “person” and “bottle” as shown in figure 5.1. The SSD MobileNet v1 COCO-model is reported to have mean Average Precision (mAP) of 21 on COCO dataset.

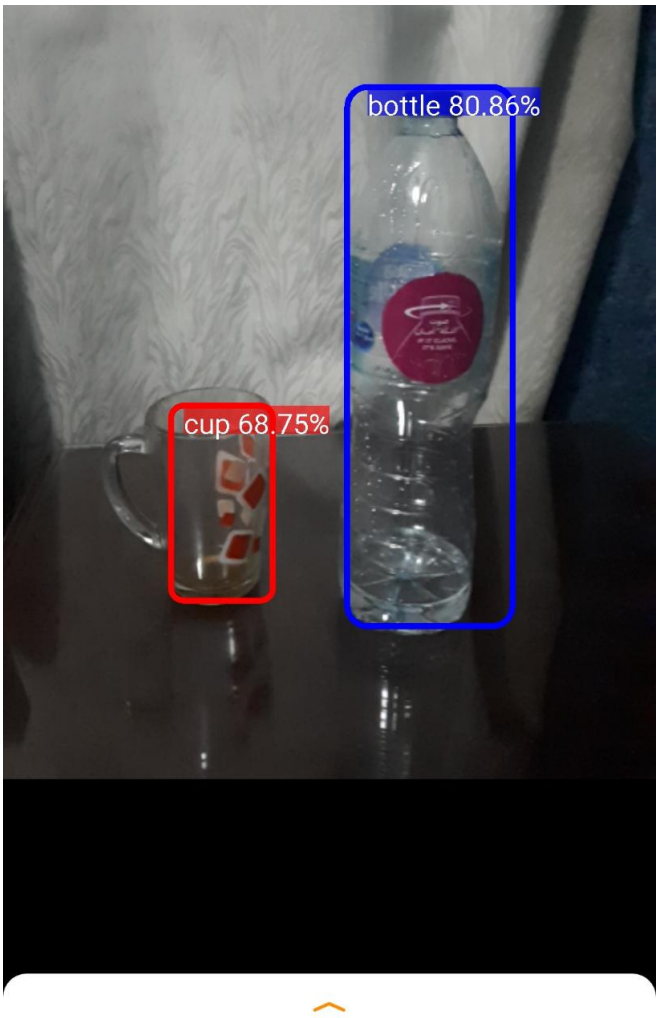

Figure 5.1: Object Detection

The pre-trained SSD model was able to recognize the bottle with accuracy of 80% but was less sure about the identity of the cup with accuracy of 68%

5.3.2 Customized Model

1. Prepare Dataset:

   The dataset of each object captured frames is saved as a collection of images, these images are big in size, because they have a very high resolution, it was then required to transform them to a lower scale[36], so that the training process could be faster and fit the input size for SSD quantizied model(300*300). The dataset is split into training set and testing set with ratios of 0.8 and 0.2 respectively. It is vital to create TFRecords that can be served as input data for training of the object detector, so it is needed to extract features of the object from each image by manually tagging the objects in the images using LabelImg as in figure 5.2 and convert it to a TFRecord. Finally, everything is in place and ready to train the model using quantized Mobilenet SSD classifier Only remaining problem: region proposal methods such as R-CNN are more accurate.

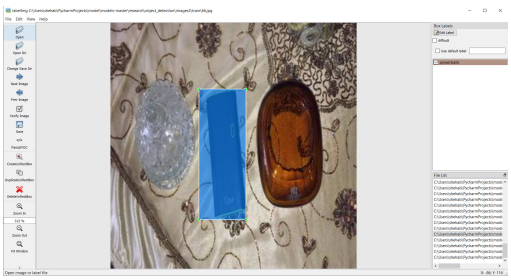

Figure 5.2: Trained data image being labeled using labelImg

2. Training Process:

   The pre-trained SSD model (SSD MobileNet v1 COCO) was fine-tuned for our dataset using manually labeled images. A provided configuration file (ssd mobilenet v1 coco.config) was used as a basis for the model configuration (after testing different configuration settings, the default values for configuration parameters were used). The provided checkpoint file for SSD MobileNet v1 COCO was used as a starting point for the finetuning process. The training

was stopped after 17400 time-steps when the mPA somewhat leveled out as in Fig. 5.3. The mAP value increased up to 8000 time-steps. However, the mAP values kept fluctuating even after that, which raised suspicion that even longer training might improve the detection results. Training the model in co-lab server took over 20 hours with CPU or, alternatively, about 2 hours with GPU for each model. The total loss value was reduced rapidly for models due to starting from the pre-trained checkpoint file as in Fig. 5.4.

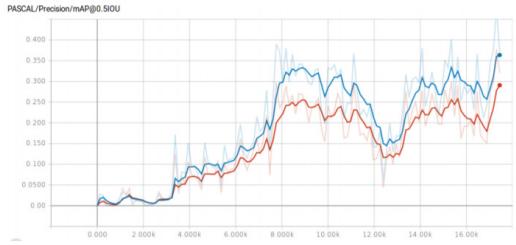

Figure 5.3: Development of map when training model

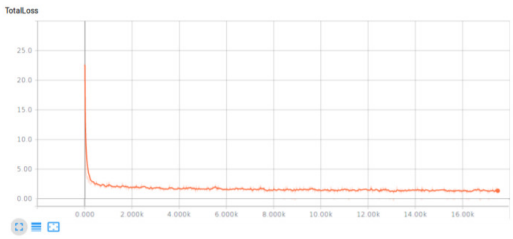

Figure 5.4: Development of loss when training model

A concern rose from the analysis of the results that models had not reached the optimal number of training steps at 17400 time-steps (for instance, the fluctuation in map level between time-steps as in Fig. 5). Therefore, the models were further trained up to 200,000 time-steps to increase value of map.

3. Results:
   The shown results are the results of the customized object after training our model.

Table 5.1: Accuracy

Distance Accuracy				
Objects	**Class**	**NImages**	**Input size**	**Accuracy**
ob.1	**Powerbank**	600	300*300	0.92
ob.2	**Book**	600	300*300	0.87

5.3.3 Distance Measurement

Measuring distance results shown in table 5.2 using equation1 were satisfying in distances between 50 inches and 100 inches, which was sufficient for guiding the user to objects but insufficient for close range obstacles,hence they were addressed in the navigation module using a threshold for the size of the objects detected on the frame.

Table 5.2: Error Ratio

Distance Accuracy		
Real Distance(inch)	**Detected Distance**	**Error Ratio**
100	95	0.05
50	46	0.08

The proposed system has been thoroughly compared with Summan's et al.'s,their system as mentioned earlier in the similar systems section, calculated the distance and fetched the best route to the user with the accuracy mentioned in table 5.3 below.

Table 5.3: Similar System's Calculated Route Distance.

Paths	User 1	User 2	User 3
	Accuracy	Accuracy	Accuracy
Path 101-105 (42.89m)	0.9	0.8	0.7
Path 104-107 (49.1m)	0.55	0.88	0.5

Through a video stream, the proposed system is able to display the best path available, hence, when compared to the other similar systems, the system provides a higher, more precise accuracy, through a better, more reliable approach shown below in table 5.4 .

Table 5.4: Proposed System's Estimated Distance

Paths	User 1		User 2		User 3	
	Est. Length	Acc.	Est. Length	Acc.	Est. Length	Acc.
Path A (2.54m)	2.4	0.95	2.35	0.92	2.2	0.87
Path B (1.27m)	1.2	0.92	1.1	0.88	1.08	0.85

Chapter 6

Conclusion

This Thesis provides a study for building an Android software paper that recognizes indoor objects using a smartphone camera, and returns a navigation guidance to reach the specified object while detecting obstacles, in order to support in-door roaming and locate specified objects processes for visually impaired people. As a future work, implementing this application for iOS developers is also needed so cross platform techniques will be needed[37], we will also improve the performance of the general detection model by increasing the object categories that can be identified. However, if the categories to be detected are increased, the recognition accuracy may be lowered, therefore, some measures for this process are needed and transfer learning methods[38] should be approached. To improve the performance of the obstacle detection model, parallel processing can be used for accelerating the detection process.

6.1 Future directions

Future research should further develop and confirm these initial findings by aiming to increase usability and availability on multiple platforms as well as facilitate the process of adding a personalized object as it requires manual annotation for each object and that can be costly for the user, further research should be conducted on the possibility of accurate room localization using image processing so that the environment could be accurately captured for later use navigation and object allocation. We will also improve the performance of the general detection model by increasing the object categories that can be identified. However, if the categories to be detected are increased, the recognition accuracy may be

lowered, therefore, some measures for this process are needed and transfer learning methods should be approached. To improve the performance of the obstacle detection model, parallel processing can be used for accelerating the detection process.

Bibliography

[1] "Vision impairment and blindness," World Health Organization, https://www.who.int/news-room/fact-sheets/detail/blindness-and-visual-impairment, 2019.

[2] R. Y. F. B. Abbas Riazi, Fatemeh Riazi, "Outdoor difficulties experienced by a group of visually impaired iranian people," *journal of current ophthalmology*, p. 85–90., 2016.

[3] "Challenges blind people face when living life," 2019.

[4] E. Y. E. K. J. E. H. M. M. Milios Awad, Tarek Mahmoud, "Intelligent eye: A mobile application for assisting blind people," *2018 9th IEEE Annual Ubiquitous Computing, Electronics Mobile Communication Conference (UEMCON)*, p. 6, June 2018.

[5] M.-C. C. W.-J. C. C.-H. Y. C.-Y. S. Liang-Bi Chen, Jian-Ping Su, "An implementation of an intelligent assistance system for visually impaired/blind people," *2019 IEEE International Conference on Consumer Electronics (ICCE)*, p. 2, 2019.

[6] B. Y. . H. G. LeCun, Y., "Deep learning." *Nature 521*, p. 436–444, 05 2015.

[7] M. K. D.-K. P. K. Dhruv Dahiya, Ashish Issac, "Computer vision technique for scene captioning to provide assistance to visually impaired," *2018 41st International Conference on Telecommunications and Signal Processing (TSP)*, p. 4, September 2018.

[8] P. S. A. G. Laviniu Ţepelea, Virgil Tiponuţ, "Multicore portable system for assisting visually impaired people," *2014 14th International Workshop on Cellular Nanoscale Networks and their Applications (CNNA)*, p. 2, 2014.

[9] M. Rehm, N. Bee, and E. Andre, "Wave like an egyptian: accelerometer based gesture recognition for culture specific interactions," in *Proceedings of the 22nd British HCI*

Group Annual Conference on People and Computers: Culture, Creativity, Interaction - Volume 1, ser. BCS-HCI '08. Swinton, UK, UK: British Computer Society, 2008, pp. 13–22.

[10] M. Hruz, P. Campr, E. Dikici, A. Kındıroğlu, Z. Krňoul, A. Ronzhin, H. Sak, D. Schorno, H. Yalçın, L. Akarun, O. Aran, A. Karpov, M. Saraclar, and M. Železný, "Automatic fingersign-to-speech translation system," *Journal on Multimodal User Interfaces*, vol. 4, pp. 61–79, 07 2011.

[11] S. J. S. J. Varsha Sharma, Chaitanya Sharma, "Assitance application for visually impaired - vision," *International Journal of Scientific Research and Engineering Development*, vol. 2, no. 6, nov 2019.

[12] "Object detection: Tensorflow lite," TensorFlow, https://www.tensorflow.org/lite/models, 2020.

[13] I. T. C. O. K. K. Manabu Shimakawa, Kosei Matsushita, "Smartphone apps of obstacle detection for visually impaired and its evaluation," *ACIT 2019: Proceedings of the 7th ACIS International Conference on Applied Computing and Information Technology*, p. 6, May 2019.

[14] S. A. F. A. Summan Zaib, Shah Khusro, "Smartphone based indoor navigation for blind persons using user profile and simplified building information model," *2019 International Conference on Electrical, Communication, and Computer Engineering (ICECCE)*, p. 6, 2019.

[15] G. Bohouta and V. Këpuska, "Next-generation of virtual personal assistants (microsoft cortana, apple siri, amazon alexa and google home)," 01 2018.

[16] M. Awad, J. Haddad, E. Khneisser, T. Mahmoud, E. Yaacoub, and M. Malli, "Intelligent eye: A mobile application for assisting blind people," 04 2018, pp. 1–6.

[17] J. Huang, V. Rathod, C. Sun, M. Zhu, A. Korattikara, A. Fathi, I. Fischer, Z. Wojna, Y. Song, S. Guadarrama, and K. Murphy, "Speed/accuracy trade-offs for modern convolutional object detectors," 07 2017, pp. 3296–3297.

[18] N. K. P. Jayawardhana, A. Aponso and A. Rathnayakem, "An intelligent approach of text-to-speech synthesizers for english and sinhala languages," *2019 IEEE 2nd Interna-*

tional Conference on Information and Computer Technologies (ICICT), pp. 229–234, 2019.

[19] M. Richards, *Software Architecture Patterns*. O'Reilly Media, Inc., 2015.

[20] A. Samkaria, "Object detection using convolutional neural networks in tensor flow," *International journal of innovative research in technology*, vol. 5, no. 4, p. 351–353, sep 2018.

[21] "Detect and recognize faces and facial features with luxand facesdk," Luxand, Inc., https://www.luxand.com/facesdk/facedetection, 2020.

[22] "Common objects in context," COCO, http://cocodataset.org/explore, 2017.

[23] S. Kak, F. Mustafa, and P. Valente, "A review of person recognition based on face model," vol. 4, pp. 157–168, 01 2018.

[24] A. Samkaria, "Object detection using convolutional neural networks in tensor flow," *INTERNATIONAL JOURNAL OF INNOVATIVE RESEARCH IN TECHNOLOGY*, vol. 5, no. 3, pp. 351–353, September 2018.

[25] N. M. D. R. Yamashita, R., "Convolutional neural networks: an overview and application in radiology." *Insights Imaging 9*, p. 611–629, 06 2018.

[26] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, M. Kudlur, J. Levenberg, R. Monga, S. Moore, D. G. Murray, B. Steiner, P. Tucker, V. Vasudevan, P. Warden, M. Wicke, Y. Yu, and X. Zheng, "Tensorflow: A system for large-scale machine learning," in *12th USENIX Symposium on Operating Systems Design and Implementation (OSDI 16)*, 2016, pp. 265–283.

[27] Y. Li, H. Huang, Q. Xie, and L. Yao, "Research on a surface defect detection algorithm based on mobilenet-ssd," *Applied Sciences*, vol. 8, p. 1678, 09 2018.

[28] E. Dong, Y. Lu, and S. Du, "An improved ssd algorithm and its mobile terminal implementation," 08 2019, pp. 2319–2324.

[29] D. Biswas, H. Su, C. Wang, A. Stevanovic, and W. Wang, "An automatic traffic density estimation using single shot detection (ssd) and mobilenet-ssd," *Physics and Chemistry of the Earth, Parts A/B/C*, 12 2018.

[30] R. P. Prasetya and F. Utaminingrum, "Triangle similarity approach for detecting eyeball movement," *2017 5th International Symposium on Computational and Business Intelligence (ISCBI)*, pp. 37–40, 2017.

[31] W. Qi, F. Li, and L. Zhenzhong, "Review on camera calibration," 06 2010, pp. 3354 – 3358.

[32] K. K. J. H. Vít Třebický, Jitka Fialová, "Focal length affects depicted shape and perception of facial images," *PLOS ONE*, 02 2016.

[33] R. Saborido Infantes, R. Morales, F. Khomh, Y.-G. Guéhéneuc, and G. Antoniol, "Getting the most from map data structures in android," *Empirical Software Engineering*, 03 2018.

[34] M. W. . M. K. . C. Marouane, "Indoor positioning using smartphone camera," *2011 International Conference on Indoor Positioning and Indoor Navigation*, november 2011.

[35] J. W. Francis, "Ssd-mobilenet v2 trained on ms-coco data," https://resources.wolframcloud.com/NeuralNetRepository/resources/, 2019.

[36] C. Thirumoorthi and K. Thirunavu, "Easy optimization of image transformation using sfft algorithm with halide language," 12 2014.

[37] N. Hui, L. Chieng, W. Ting, H. Mohamed, and M. Mohd Arshad, "Cross-platform mobile applications for android and ios," 04 2013, pp. 1–4.

[38] K. T. . W. D. Weiss, K., "A survey of transfer learning." *J Big Data 3*, 05 2016.