National University of Computer and Emerging Sciences, Lahore

Vision Guide

Fatima Tajammul 20L-1288 BS(SE)

Saba Naeem 20L-1328 BS(SE)

Maida Shahid 20L-1377 BS(SE)

Supervisor: Zeeshan Ali Rana

Final Year Project

June 3, 2024

Anti-Plagiarism Declaration

This is to declare that the above publication was produced under the:

Title: Vision Guide

is the sole contribution of the author(s), and no part hereof has been reproduced as it is the basis (cut and paste) that can be considered Plagiarism. All referenced parts have been used to argue the idea and cited properly. I/We will be responsible and liable for any consequence if a violation of this declaration is determined.

Date: ....

Name: Maida Shahid

Signature: ....

Name: Saba Naeem

Signature: ....

Name: Fatima Tajammul

Signature: ....

Author's Declaration

This states Authors' declaration that the work presented in the report is their own, and has not been submitted/presented previously to any other institution or organization.

Abstract

Our project focuses on helping the visually impaired community to help them understand their surroundings. The existing visionary technologies such as smart glasses for the blind are extremely expensive and inaccessible for the majority of the blind community. The wide use of mobile devices in recent times, inspired us to bring an accessible and cost-efficient mobile application for such a community. We developed a mobile application that performs multiple object detection and tracking in real-time. Our application provides useful descriptions of these detected objects and integrate them with the audio to make it user-friendly.

Executive Summary

In today's world, technology can help make life better for people with disabilities. There are devices available for visually impaired people like smart glasses but these glasses are highly priced so the majority of people cannot afford them. Smartphones, on the other hand, are easily accessible and widely used by the majority of people including those with vision disabilities. This is our motivation to develop a mobile application that will use a smartphone's camera to detect and track multiple objects and provide audio descriptions of them to the user in real-time. It will help blind people understand what's around them using computer vision and NLP.

The purpose of this document is to elaborate on the methodologies and goals to develop a Multi-Object Detection System (MODS) called Vision Guide. The main purpose of this project is to detect and classify various objects present in the camera's field of view and provide a real-time description of the surroundings through audio feedback. Ultimately, this document seeks to explore ways to enhance the independence and safety of individuals with visual impairments.

The document further elaborates on the scope of this project. We will mainly create a mobile application that helps in multiple object detection and tracking captured from the device's camera to provide useful audio descriptions in real-time. The application will have an easy and simple user interface to be used easily by blind users.

The document then goes on to elaborate on the detailed literature review of previous research done on the chosen topic and gives detailed knowledge of each technique. Our project revolves around four key roles which include image processing, object detection and analysis, real-time description, and audio integration. In this document, we did research on each objective individually and elaborated further on how this research is relevant to our project. The research work has provided us with insights into the relevant models that can be used in our project.

The document further provides a comprehensive overview of the Vision Guide mobile application, focusing on its proposed features, hardware/software requirements, and use cases. The architectural strategies, data set choices, and system components are outlined, along with sequence diagrams to illustrate the system flow.

Table of Contents

List of Figures

List of Tables

Chapter 1 Introduction

Visual disabilities can be extremely challenging for people to understand their surroundings and perform everyday tasks [1]. People with such disabilities struggle with routine activities such as navigating in unfamiliar environments and identifying their surrounding objects. While advanced solutions such as smart glasses exist like AIRA smart glasses [2], their high cost makes them inaccessible to a large population. In addition, mobile applications designed to assist these people provide limited functionalities that do not provide comprehensive assistance to users. So, there's a necessary need for innovation, to enhance their quality of life and make them feel more independent.

In recent years, AI has emerged as a powerful tool with the potential to transform the lives of people with disabilities. Its field of computer vision has made it possible to understand and analyze visual information through smart systems. A lot of work has been done and is still in the research and development phase that aims to assist people with visual disabilities specifically to assist them in understanding their environments. Our project aims to develop a mobile application that will perform multiple object detection and tracking in real-time to provide useful descriptions of these detected objects. It will offer audio feedback, making it simple for users to access information effortlessly.

1.1 Purpose of this Document

This document aims to outline the purpose and methodologies that will be used to develop a Multi-Object Detection System (MODS) called Vision Guide. The purpose of this document is to discuss innovative models, which can be used to detect and classify various objects present in the camera's field of view and provide a real-time description of the surroundings through audio feedback. Ultimately, this document explains ways to enhance the independence of individuals with visual impairments.

1.2 Intended Audience

The target audience for this project will consist of a wide range of stakeholders. The FYP committee will use the document to gain a thorough understanding of the problem at hand and evaluate the proposed solution. Students and researchers can use the document to build upon our solution and make enhancements to the same.

1.3 Definitions, Acronyms, and Abbreviations

1.3.1 Definitions

Visual Disabilities: Conditions where a person loses the ability to perceive their surroundings, typically including impairments in vision or blindness.

Smart Glasses: Glasses with a camera system, which perceive the surroundings through augmented reality and enhance the user's visual experience by providing audio feedback.

Computer Vision: A field of AI that enables machines to interpret and understand visual information from images and videos.

1.3.2 Abbreviations

AI: Artificial Intelligence

SDG: Sustainable Development Goals

MODS: Multiple Object Detection System

NLP: Natural Language Processing

CNN: Convolutional Neural Network

LSTM: Long Term Short Memory

1.4 Conclusion

The first chapter of this document offers a brief intro to the project featuring a description of the problem, document, intended audience, and acronyms and abbreviations. The second section deals with project vision and encompasses the objectives, goals, and scope of the project among others. It is followed by a chapter on Literature review that provides a comprehensive survey and critique of the literature on the topic.

Chapter 2 Project Vision

In this chapter, we will provide a comprehensive overview of our project's vision while elaborating our problem statement in depth. We will discuss our project's objectives and scope, shedding light on how these objectives align with our Sustainable Development Goal.

2.1 Problem Domain Overview

The domain of this project revolves around the challenges faced by visually impaired people in achieving independence. Visual impairment limits independence and existing accessibility technologies have limitations. Moreover, advancements in AI and computer vision offer opportunities for improvement, and there is a need for a comprehensive solution integrating object detection and NLP. Vision Guide will be using deep learning and computer vision techniques to detect and describe objects and scenes in real-time. We will be trying various models, for example using CNN to extract features from a video frame and then using bidirectional LSTM for classification.

2.2 Problem Statement

The existing visionary technologies such as smart glasses for the blind are extremely expensive and inaccessible for the majority of the blind community. Vision Guide will be an accessible and affordable application that will utilize smartphone cameras to help visually impaired individuals understand their surroundings.

2.3 Problem Elaboration

As mentioned already, over 295 million people across the globe face visual impairment. Out of these, 33% are not treatable [3]. Visually-challenged individuals can use mobile phones to perceive the presence of objects and read/identify objects from specific distances. Visual impairment is a widespread issue that has a significant impact on how people live their everyday lives. Currently, many visually impaired individuals rely on human assistance for tasks that sighted individuals take for granted. This dependency can lead to a loss of independence and privacy. Secondly, the lack of access to real-time visual information creates obstacles in navigating public and private spaces. Understanding one's environment in real-time is crucial for safety and independence. The ability to independently navigate the environment, identify objects and settings, and obtain visual information are all things that visually impaired people find extremely challenging. Lastly, existing technologies and devices designed to assist in this cause are often expensive and not readily accessible to the general public. Hence, there should be an app that provides visually impaired people with independence, safety, and accessibility in their daily lives.

2.4 Goals and Objectives

We aim to achieve the following specific objectives in this project:

- To use algorithms that detect and track objects in various environments in real-time.
- To perform object recognition of the detected objects by using image processing techniques such as segmentation, feature extraction, and classification.
- To provide natural language descriptions of the scene based on the detected and tracked objects, their attributes, and their relations.
- To provide an audio feedback to make the system user-friendly.

2.5 Project Scope

The scope of this project is mainly to create a mobile application that helps in multiple object detection and tracking captured from the device's camera to provide useful audio descriptions in real-time. Our application shall provide relevant descriptions. The application will have an easy and simple user interface to be used easily by blind users. Moreover, we will take feedback from individuals from the blind community to evaluate user satisfaction.

2.6 Sustainable Development Goal (SDG)

For this project, the SDG that we are targeting belongs to Reduced Inequality. Reduced Inequality promotes equality by ensuring equal opportunity and reducing discrimination. This is achieved via bridging the divide between society and differently-abled people. Making those with disabilities more independent is one way to bridge this gap, and with our application, we hope to contribute in any small way to this important cause.

2.7 Conclusion

In summary, the Vision Guide project is all about helping people who can't see well. It uses cool technology to make an app on your phone that tells you about the things around you. The goal is to make life easier and fairer for those who are visually impaired. By doing this, the project is joining the effort to treat everyone equally, making the world a better place for everyone.

Chapter 3 Literature Review / Related Work

Given the nature of our problem, a considerable amount of work and time went into reviewing published literature. This chapter contains a detailed review of the relevant literature, alongside a description of important terms, acronyms, and abbreviations.

3.1 Definitions, Acronyms, and Abbreviations

3.1.1 Definitions

Relation Classifications: The process of determining the semantic relationship or connection between identified objects or entities in a given context.

Entity Features: Characteristics or attributes associated with specific entities or objects, often used in machine learning and data analysis to distinguish and classify entities.

Attention: A mechanism in neural networks that allows the model to focus on specific parts of the input data while processing, enhancing its ability to capture relevant information.

Tensor Layer: A layer in a neural network that handles multi-dimensional arrays of data, used for advanced computations and capturing complex relationships between data elements. **Multimodal Data**: Data derived from multiple observation channels or sources, often of different types (e.g., text, images, audio), used to provide a more comprehensive understanding of a given situation or context.

Natural Language Processing: A field of artificial intelligence focused on enabling computers to understand, interpret, and generate human language.

Feature extraction: The process of extracting the most important and relevant characteristics from a set of data such as recognizing the most distinctive traits of a face or an object in an image, which a computer can use to understand or distinguish it from other things.

Haar wavelet: A basic mathematical function that alternates between two values, used in signal processing and image analysis for tasks like edge detection.

Speech synthesizer: A technology that converts text or written content into spoken audio.

Convolutional Neural Network: A type of deep neural network designed for image recognition and processing. It uses convolutional layers to automatically and adaptively learn spatial hierarchies of features.

KITTI: A benchmark data set for autonomous driving research, providing data sets for tasks such as object detection, tracking, and scene understanding.

LSTM: Long-term short memory (LSTM) is an artificial neural network.

Waveform: Waveform refers to the appearance of a signal's wave at any particular moment.

3.1.2 Abbreviations

Bi-RNN: Bidirectional Recurrent Neural Network

AT-RNN: Attention and Tensor-based Recurrent Neural Network

NLP: Natural Language Processing

CRF: Conditional Random Field

BI-GRU-CRF: Bidirectional Gated Recurrent Unit-Conditional Random Field

LSTM: Long Short-Term Memory

CNNs: Convolutional Neural Networks

RNNs: Recurrent Neural Networks

CV: Computer Vision

SIFT: Scale-Invariant Feature Transform: Long Term Short Memory

SURF: Speeded Up Robust Features

PCA: Principal Component Analysis

OCR: Optical Character Recognition

DL: Deep learning

SVM: Support Vector Machine

CNN: Convolutional Neural NetworkLong Term Short Memory

MOT: Multiple Object Tracking

KITTI: Karlsruhe Institute of Technology and Toyota Technological Institute

SSD: Single Shot MultiBox Detector

YOLO: You Only Look Once

OWOD-RCNN: Object Detection with Region-based Convolutional Neural Network

MSE: Mean Squared Error

GPU: Graphics Processing Unit

3.2 Detailed Literature Review

Several studies have delved into the realm of real-time object detection and tracking, employing diverse algorithms to ensure accurate and efficient performance. Approaches such as YOLO (You Only Look Once) and Faster R-CNN (Region-based Convolutional Neural Network) have emerged as robust solutions, showcasing the ability to detect multiple objects simultaneously with impressive accuracy [4]

[5]. The field of object recognition has seen significant advancements through image processing techniques, including segmentation, feature extraction, and classification. Noteworthy contributions include the work by Voila and John, who developed an object detector using classification, Support Vector Machine (SVM), and Haar wavelet features in 2011 [4]. These techniques, capable of capturing intricate features while minimizing noise, lay the foundation for effective object recognition. Efforts have been made to bridge the gap between computer vision and natural language understanding. Recent studies explore methodologies to generate natural language descriptions of scenes based on detected objects, their attributes, and relationships [6] [7]. These approaches leverage advanced deep learning models and relational networks to establish a nuanced understanding of the visual content within an image. Recognizing the importance of user-friendliness, integrating audio feedback into systems has been a focus of research. Providing auditory cues enhances the accessibility of the system, making it more inclusive for users with visual impairments. The use of audio cues in conjunction with object detection and tracking contributes to a comprehensive and user-friendly experience. These are some of the applications working in our domain. But the problem with them is that you have to pay to use them. Moreover, some of the features are not available for real-time use. For example, Seeing AI, developed by Microsoft for blind people uses computer vision for object and text recognition. It is only available in iOS and some features may require internet connection.[8] Next, we have Envision AI, an app that uses artificial intelligence and computer vision to provide various features aimed at enhancing the independence of blind and visually impaired users. However some advanced features may be part of a subscription model, and it also depends on the internet for optimal performance.[9] Lastly, TapTapSee is an app that utilizes computer vision to provide auditory information about the visual world for blind and visually impaired users. It has some limitations like being influenced by camera quality and describing scenes on the basis of image, thus having no real-time video feedback is available here.[10] The more detailed literature review is as follows:

3.2.1 Multiple Object Detection and Tracking

Many studies have been done in the domain of multiple object detection and tracking which are part of computer vision. Researchers have explored various methods to identify and monitor objects in changing environments. Techniques such as YOLO, and Faster R-CNN, have emerged as effective solutions. These approaches use advanced deep-learning models that demonstrate admirable accuracy in detecting and tracking multiple objects simultaneously. With the advancement of computer vision techniques, there has been significant progress in the field of object detection and tracking. Various methods, such as image processing, computer vision, and deep learning (DL), have been employed to detect objects in images. According to [4], object detection can be broadly categorized into three approaches:

- Appearance-based Approach: Utilizes image processing techniques.
- Motion-based Approach: Involves the analysis of a sequence of images.
- DL-based Approach: Employs algorithms and neural networks.

In 2011, Voila and John developed an object detector using classification, Support Vector Machine (SVM), and Haar wavelet features. Haar wavelets, being capable of capturing features at a resolution consistent throughout an object class while ignoring noise, make them well-suited for object detection. Object detection algorithms can be classified into two main categories:

- One-stage detectors: Perform object detection in a single stage without a separate proposal stage. EfficientDet is an example built on this concept [5].
- Two-stage detectors: Generate candidate object proposals and then classify each proposal.

Moreover, the advancement in computer vision has lead to many state of the art models. Popular deep learning-based object detection models include:

- Faster R-CNN: A two-stage detector using a region proposal network and VGG or ResNet for classification.
- YOLO: A one-stage detector dividing the image into a grid and predicting bounding boxes and classes.
- SSD: A one-stage detector using a feature pyramid network for multi-level feature extraction [11].

New approaches, such as OWOD-RCNN and relation networks, show promise. OWOD-RCNN builds upon Faster R-CNN, while the relation network focuses on modeling object relationships in an image [6][7]. Object detection in DL involves labeling objects in an image with correct classes and predicting bounding boxes. CNNs are commonly used for generic and domain-specific object detection. It covers areas like edge detection, image segmentation, and face detection. The availability of benchmark data sets and GPU development has driven the widespread adoption of DL-based object detection. Tracking algorithms maintain multiple hypotheses of object trajectories in a graph structure. The process involves hypothesis generation, likelihood computation, and hypothesis management. In the hypotheses generation step, the graph is extended to include recent object detection results, generating multiple trajectory hypotheses [4]. While exploring the methods and techniques used in object detection I have encountered some of the main challenges. Challenges in Object Detection:

- Object Variation: Objects exhibit variations in size, shape, color, and pose.
- Background Clutter: Occlusion by other objects or the background makes detection challenging.

- Illumination Changes: Objects may appear differently under varying lighting conditions.
- Real-time Requirements: Many applications demand real-time object detection, as seen in autonomous driving and robotics.

Additional references include the use of CNNs and feature pyramids [12], object detection with OpenCV [13], and benchmarks such as MOT, KITTI, and nuScenes [14].

3.2.1.1 Summary of the research item

This Summary discusses the progress in object detection and tracking, incorporating techniques like image processing, computer vision, and deep learning (DL). Key points include the categorization of object detection approaches (appearance-based, motion-based, DL-based), historical context of a 2011 object detector by Voila and John, classification of detection algorithms into two-stage and one-stage detectors, challenges in object detection (variation, clutter, illumination changes, real-time demands), popular DL-based models (Faster R-CNN, YOLO, SSD), new approaches like OWOD-RCNN and relation networks, DL's role in object detection, widespread adoption of DL, and object tracking using graph-based hypotheses. The synthesis emphasizes the dynamic landscape of advancements in computer vision, covering challenges, models, and methodologies.

3.2.1.2 Critical analysis of the research item

The research item provides an insightful exploration into the diverse methodologies employed in the field of object detection and tracking, ranging from traditional image processing techniques to advanced deep learning approaches. A critical examination of the presented content reveals both strengths and limitations across different approaches. Object detection is classified into appearance-based, motion-based, and DL-based approaches which provide a comprehensive framework. However, a fine understanding of their interchange is important. While appearance-based methods excel in certain scenarios, such as clear images, they struggle in cases of occlusion. Similarly, motion-based approaches face challenges in complex scenarios, highlighting the need for a balanced integration of these techniques.[4] The classification of object detection algorithms into two-stage and one-stage detectors is well-presented. However, a more detailed discussion of the trade-offs between these categories, including speed, accuracy, and adaptability to different scenarios, would enhance the analysis. The identified challenges in object detection, including object variation, background clutter, illumination changes, and real-time requirements, effectively highlight the complexities of real-world applications. A more in-depth exploration of strategies to address these challenges, particularly in the context of DL-based solutions, would provide a richer analysis.[15] The overview of popular DL-based models like Faster R-CNN, YOLO, and SSD is valuable. However, a deeper exploration of their strengths and weaknesses, as well as a comparison of their performance in specific scenarios, would provide a more comprehensive understanding for researchers and practitioners. It is also important for beginners to have an understanding of where to use which technique.[4] The mention of promising technologies like OWOD-RCNN and relation networks is commendable. However, further details on the specific advancements and improvements they offer over existing models would enhance the assessment of their potential impact on the field.[6] Efficacy over existing models needs further empirical validation and new approaches may introduce increased model complexity and training requirements. Hence, while the research item serves as a valuable overview of object detection and tracking methodologies, a more in-depth analysis of specific aspects, along with critical comparisons and discussions of emerging technologies, would enhance its impact and utility in guiding future research in this dynamic field.

3.2.1.3 Relationship to the proposed research work

The research work presented on object detection and tracking methodologies holds significant relevance to the development of an app aimed at assisting blind individuals by providing scene descriptions through object detection and tracking. The proposed research work offers valuable insights and methodologies that can be directly applied and adapted to enhance the effectiveness of the app for the visually impaired. The diverse approaches to object detection, such as appearance-based, motion-based, and DL-based methods, provide a foundation for understanding the complex nature of scenes. In the context of the app for the blind, leveraging these approaches can enhance the system's ability to identify and describe objects in real-world environments. The historical progression from traditional methods, such as Haar wavelet features and SVM, to DL-based solutions is crucial. Implementing a blend of these techniques can contribute to a more robust object recognition system within the app, accommodating various scenarios and object types. Classifying object detection algorithms into two-stage and one-stage detectors provides insights into the trade-offs between precision and speed. Tailoring these approaches to the app's requirements ensures an optimal balance between real-time performance and accurate scene descriptions for users with visual impairments. Recognizing and addressing challenges in object detection, such as object variation, background clutter, and illumination changes, directly aligns with the app's objectives. Tailoring the app to handle these challenges contributes to more reliable and informative scene descriptions for blind users. Leveraging popular DL-based models like Faster R-CNN, YOLO, and SSD, and exploring new approaches like OWOD-RCNN and relation networks can significantly enhance the accuracy and adaptability of the app. These models can be trained to recognize diverse objects and contextual relationships, enabling more comprehensive scene descriptions. Hence, the research work not only provides a roadmap for developing an object detection and tracking system within the app but also offers valuable insights into addressing the specific needs and challenges faced by blind individuals. The proposed project, when informed by the methodologies discussed in the research work, has the potential to significantly enhance the independence and mobility of visually impaired users by providing them with detailed and accurate scene descriptions in real-time.

3.2.2 Image Processing for object Recognition

Once the object is detected in an image frame, the next step is recognition of this particular object known as object recognition. The image processing algorithms play an important role in object recognition to understand what exactly our detected objects are.

In the research article [16], the contributors explain the working of a mobile application to capture the images of objects and compare them with objects in the database to identify and recognize the objects correctly. The mobile application performs color and light source detection, which assists in their object recognition module. The SIFT algorithm of computer vision was used to identify the features of images. Moreover, they used classification algorithms to decrease the number of wrong identification cases. To test their object recognition algorithms, they used an HTC Desire HD smartphone. The results concluded an 89% correct recognition rate, a 5.5% false recognition rate, and a 5.5% rate of not recognizing any object.

Feature extraction is a major step in image processing that helps us to identify object traits, as discussed in [17]. Through feature extraction, we can classify the objects based on similar traits which helps us in recognizing the objects. The contributors of this article emphasize the use of two algorithms: SIFT and SURF. SIFT works in a way that key points from the captured image are taken and stored in a database, and then the key points from the captured images are compared with the data in the database for successful object recognition. SURF analyzes Haar wavelet responses, computes descriptors, and achieves contrast invariance from it. SIFT is more robust in nature but SURF is more speedy.

According to research paper [18], the attributes explain how to detect objects, recognize faces, and read text in images using HOT glass (Human face, object, and textual recognition for the visually impaired). The contributors explain two algorithms for object and face recognition: PCA and SIFT. They explain how human faces can be recognized through PCA and how objects can be recognized through SIFT. Using PCA, images captured with the camera are analyzed and key points from it are extracted and matched with pre-stored images in the database. In the SIFT algorithm, images are converted into grayscale images for more accurate comparison and noise is removed, then the key points from these images are compared with key points of reference images in the database. The system's overall workflow includes image preprocessing, feature extraction, and recognition algorithms.

3.2.2.1 Summary of the research item

Image processing techniques are extremely helpful in recognizing the detected objects in an image frame. Various computer vision algorithms are present to perform object recognition such as SIFT and SURF. SIFT is more accurate but is computationally demanding while SURF is an enhanced and faster version of SIFT. Feature extraction is a fundamental step to recognize the detected objects in an image frame, as it extracts the features from the image captured and recognized and matches its patterns with features of images already stored. PCA algorithm is used for recognizing faces in an image by comparing them with face images in the database. It can be integrated with SIFT to perform facial and object recognition. All these methods encompass aspects of object recognition and image processing, each optimizing performance in its unique way.

3.2.2.2 Critical analysis of the research item

Article [16] explains the development of a mobile application used for object recognition using color and light source detection, aided by the SIFT algorithm. A comparison is made between the SIFT and SURF algorithms in article[17] that gives us a better idea about which algorithm is more efficient and accurate under certain situations. Research article [18] highlights the integration of PCA and SIFT for human face recognition and object recognition respectively. The strengths of the above articles include a strong emphasis on the use of computer vision techniques, recognition of the importance of feature extraction, and the potential for multi-level recognition integration. However, weaknesses include a lack of detailed methodology descriptions and the absence of references in performing object recognition in a real-time environment.

3.2.2.3 Relationship to the proposed research work

We get valuable insights into the field of object recognition using computer vision, feature extraction, and integration of recognition processes after reading the above research papers. Feature extraction, which is directly relevant to our project has been discussed and elaborated by most of the articles. The data sets used in the experiments of the above articles are routine objects that are relevant to our project. These findings are relevant to our proposed research work as they underscore the significance of the thorough methodology and comprehensive evaluation we will use in our project.

3.2.3 Natural Language Description

The key to producing efficient descriptions is establishing relationships between identified objects, which is essential for accurate and meaningful descriptions. According to the article [19], machine learning and feature design were heavily relied on in traditional approaches, requiring complex NLP pipelines and manual feature engineering. However, deep learning methods like Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) provide simplified and automated solutions for relation classification. A novel model for relation classification is a bidirectional recurrent neural network (Bi-RNN) with attention and tensor layers that enhance semantic understanding within sentences. Entity features are considered an important aspect of relationship classification, the tensor layer is used to capture hidden entity interactions. The AT-RNN model, which stands for Attention and Tensor-based Recurrent Neural Network, combines LSTM networks, word-level attention, and entity-level tensors to improve semantic relation classification. Moreover, it includes an entity-level tensor layer to detect complex entity interactions. The combination of these elements facilitates semantic relation classification. The article[20] proposes an unexplored method for deep-level grammatical and semantic analysis of the English language using a multi-modal neural network. This article addresses the challenges and advancements in Natural Language Processing (NLP) and word segmentation processing. It emphasizes the importance of handling multimodal data, that shows how multiple observation methods are used to obtain data information from multiple observation channels, and these data from different information channels describe the same concept or meaning. NLP has evolved from early language structure analysis to practical applications in the real world. It highlights the complexities of deep neural networks, noting that increasing layer depth doesn't guarantee improved results, and traditional machine learning methods struggle with long-distance text dependencies. To address these issues, the paper discussed how labeling problems can be tackled by using a proposed technical route of "understand first and then segmentation". For systems that require real-time performance, dictionary-based and rule-based methods have a high probability of producing efficient results[20]. The main innovation is to use the maximum entropy model as a tokenizer to automatically label characters. This method has the highest recall rate of 72.9% in the AS2003 closed test experiment [20]. The main advantage of using a dictionary-based and rule-based method is the formulation of suitable dictionaries according to special scenarios. Furthermore, this paper introduces innovative techniques for English word segmentation, including a multimodal fusion feature extraction method and a hybrid network called BI-GRU-CRF. This paper uses the BI-GRU neural network and combines it to the CRF model to solve the problem of sequence labeling at the sentence level analysis, based on BI-GRU-CRF hybrid network English word segmentation processing method. These techniques extract low-dimensional fusion features from high-dimensional data, improving segmentation accuracy and processing speed. Experiments show that this method performs similarly to the BI-LSTM-CRF model but is 1.94 times faster. Overall, these findings showcase exciting possibilities for advancing NLP through innovative approaches and experimental results.

3.2.3.1 Summary of the research item

A lot of work has been done on generating natural language descriptions using NLP, on real-time detected objects. The key to producing efficient and accurate descriptions is by understanding the relationship between objects. The study focuses on a Bi-RNN, which is a recurrent neural network that uses attention and tensor layers to better understand sentence semantics. It highlights the importance of entity features and introduces the AT-RNN model, which combines LSTM networks, word-level attention, and entity-level tensors, eventually improving semantic relation classification. The paper also addresses grammatical and semantic analysis challenges in NLP, suggesting an ”understand first and then segmentation” approach. It showcases the innovative techniques for English word segmentation, which include a multimodal fusion feature extraction method and a BI-GRU-CRF hybrid network. These methods exhibit promising advancements in NLP by providing accurate and speedy results, through innovative approaches and experiments.

3.2.3.2 Critical analysis of the research item (Strengths and Weaknesses)

In the article [19] an innovative model is introduced for relation classification called AT-RNN. AT-RNN combines LSTM networks, word-level attention, and entity-level tensors to improve semantic relation classification. The model has outperformed state-of-the-art results on different benchmark data sets i.e. SemEval-2010 Task 8 data set and the New York Times (NYT) relation extraction data set with 87.2% and 85.1% accuracy respectively. Whereas, article[20] focuses on deep-level grammatical and semantic analysis of the English language. It combines BI-GRU neural networks with CRF models to solve the problem of sequence labeling at the sentence level and proposes BI-GRU-CRF. This approach has been shown to improve segmentation accuracy and processing speed when performed on AS2003, PTB, MNLI, and CNN/Daily Mail text summarization data sets. Both articles provide valuable contributions to the field of NLP for object description. However, these research items do not provide the development of NLP models that can jointly perform relation classification and deep-level grammatical and semantic analysis.

3.2.3.3 Relationship to the proposed research work

Our project requires us to produce descriptions of the objects detected in the camera scope. For that, we need to understand the relationship between those objects and the semantical analysis of the objects. By using AT-RNN we can provide precise descriptions of the objects, ensuring visually impaired individuals receive more detailed and accurate information about their surroundings [19]. By implementing the BI-GRU-CRF [20] hybrid network for English word segmentation in Vision Guide, we can enable real-time object detection and description, ensuring visually impaired users receive timely information about their environment, and enhancing their safety and independence. The research also emphasizes the handling of multimodal data, which aligns with the objectives of the Vision Guide. By considering data from various observation channels, Vision Guide can provide a more comprehensive understanding of the environment, including not only object detection but also contextual information that enhances the user experience. For these reasons, the research of natural language description has proved to be quite resourceful.

3.2.4 Integration of Audio for Accessibility

The textual descriptions generated through our mobile application should be integrated with the audio system to make it user-friendly and easy for blind users.

3.2.4.1 Summary of the research item

In research article [21], a Python module-based text-to-speech synthesizer and an audio amplifier are used to convert textual information into audio. The research article discusses multiple text reader systems and their advantages and disadvantages. The text-to-speech synthesizer discussed in the article converts an input text generated from images of printed text analyzed by OCR to audio. It provides audio of textual information in a natural voice. The voice can be selected from a couple of options and the rate of speech and volume of the sound can also be adjusted. In the experimental conduction, the input text was spoken using a microphone internally connected.

3.2.4.2 Critical analysis of the research item

The Python-based text-to-speech module discussed in the research paper converts text into natural-sounding speech which makes it user-friendly for blind users. The synthesizer was integrated with an OCR and camera system that performs the conversion of text generated from the OCR approach on the images captured through the camera. Our project doesn't focus on the OCR system but generates image descriptions for real-time environments. Nevertheless, the aim of text-to-speech remains the same. The synthesizer is available offline and available as an open source that will benefit us to integrate it with our project.

3.2.4.3 Relationship to the proposed research work

We get information about how we can convert the textual image descriptions generated by our mobile application to audio. The findings in the research paper are relevant to our proposed research work as they underscore the significance of thorough methodology, comprehensive evaluation, and staying updated with the latest developments to ensure the research's success and relevance in the domain of text-to-speech conversion.

Table 3.1: Summary of Literature Review

The summary of the reviewed articles related to Vision Guide

Name	Author	Year	Result	Method	Description
Object Detection [11]	Yali Amit, Felzenszwalb, Ross Girshick	2020	Deep learning-based object detection models have achieved state-of-the-art results on a variety of benchmarks	Not Applicable	Discusses the different types of object detection algorithms, the challenges involved, and the state-of-the-art approaches.
A Trainable System for Object Detection [22]	Constantine Papageorgiou, Tomaso Poggio	2000	The Viola-Jones object detector	Haar wavelets and support vector machines (SVMs)	Viola-Jones object detector is a seminal paper in the field of object detection. It is a simple and effective object detector that can be used in a variety of applications
EfficientDet: Scalable and Efficient Object Detection [5]	Mingxing Tan Ruoming Pang Quoc V. Le	2020	A new family of object detectors called EfficientDet	A feature network called BiFPN and A compound scaling method	simple to implement and can be trained to detect a wide variety of objects. Well-suited for real-time applications, where speed and efficiency are important
Towards Open World Object Detection [6]	K J Joseph, Salman Khan, Fahad Shahbaz Khan, Vineeth N Balasubramanian	2021	OWOD-RCNN is based on the popular Faster R-CNN object detector	Contrastive learning framework, an incremental learning module, and an adversarial training framework	OWOD-RCNN is a promising step towards the development of OWOD models that can be used in real-world applications

Name	Author	Year	Result	Method	Description
Deep learning in multi-object detection and tracking: state of the art [4]	Sankar K. Pal, Anima Pramanik, J. Maiti, Pabitra Mitra	2021	The combination of Faster RCNN and Deep SORT is superior to other combinations according to all kinds of tracking evaluation metrics	Compare different algorithm on MOT2015 data sets	A detailed review primarily on various deep learning (DL)-based models for the tasks of generic object detection, specific object detection, and object tracking, considering the detection and tracking both individually and in combination
A detection-based multiple object tracking method [15]	Mei Han, A. Sethi, Wei Hua, Yihong Gong	2004	Real time tracking of objects	The multiple object tracking method keeps a graph structure where it maintains multiple hypotheses about the video and yields the best hypothesis to explain the video.	Introduces a method for tracking varying and unknown numbers of objects by integrating image-based detection with a graph-based multiple object tracking system.
Object Tracking Via ImageNet Classification Scores [12]	Li Wang, Ting Liu, Gang Wang	2020	A novel occlusion estimation method based on high-level semantic category responses of CNN classifiers pre-trained on the large-scale ImageNet data set.	To assess the performance of the proposed tracker, they evaluate tracker on the OTB benchmark [36] containing 50 video sequences.	An effective method is proposed for occlusion detection and a novel tracking method and a linear motion model is adopted to effectively re-detect the lost target.
Multiple object detection using OpenCV on an embedded platform [13]	Souhail Guennouni, Ali Ahaitouf, Anass Mansouri	2014	Implement object detection on an embedded platform.	A developed application for multiple objects detection based on OpenCV library	Discusses what object detection is and why it is important. It also goes into the details of how to implement object detection using OpenCV on an embedded platform.

Name	Author	Year	Result	Method	Description
DEFT: Detection Embeddings for Tracking [14]	Mohamed Chaabane, Peter Zhang, J. Ross	2021	An efficient joint detection and tracking model named DEFT, or "Detection Embeddings for Tracking"	Appearance-based object matching network jointly-learned with an underlying object detection network, LSTM	They propose a new approach to multiple object tracking (MOT).
Relation classification via recurrent neural network with attention and tensor layers [19]	Runyan Zhang Fanrong Meng Yong Zhou Bing Liu	2018	AT-RNN has been shown to outperform BI-RNN on tasks that require deep understanding of the semantics of a text.	AT-RNN, BI-RNN	AT-RNN enhances semantic relation classification, tensor layer detect complex entity interactions
Feature Extraction and Analysis of Natural Language Processing for Deep Learning English Language [20]	Dongyang Wang, Junli Su , Hongbin Yu	2020	BI-GRU-CRF achieve state-of-art of results on segmentation, part-of-speech tagging, and natural language inference	BI-GRU-CRF hybrid network	Purposes the innovative techniques for English word segmentation, which includes multimodal fusion feature extraction method and a BI-GRU-CRF hybrid network.
Object recognition for blind people based on features extraction [17]	Hanen Jabnoun, Faouzi Benzarti,and Hamid Amiri	2014	Percentage of matching in object identification using SURF is 20% total matches (18% correct, 2% incorrect) and SIFT is 85% total matches (82% correct, 3% incorrect)	Applied the algorithms on the entire video scene for object detection across all frames	It compares the SURF and SIFT algorithm.
HOT GLASS - HUMAN FACE, OBJECT AND TEXTUAL RECOGNITION FOR VISUALLY CHALLENGED [18]	Diwakar Srinath, Praveen Ram, Sira R, Kaleiselvi V.G.R aand Agitha G	2017	Not Applicable	Applied PCR and SIFT on every-day use home objects	Explained how human faces can be recognized through PCA and how objects can be recognized through SIFT

Name	Author	Year	Result	Method	Description
Object recognition in a mobile phone application for visually impaired users [16]	K. Matusiak1 , P.Skulimowski and P. Strumiááo	2013	Concluded an 89% correct recognition rate, a 5.5% false recognition rate, and a 5.5% rate of not recognizing any object.	Used an HTC Desire HD smartphone to test the working of object recognition	Used algorithm to perform color detection and light source detection that assists in their object recognition module that used SIFT
An Intelligent Text Reader based on Python [21]	Prabhakar Manage, Veeresh Ambe, Prayag Gokhale, Vaishnavi Patil, Rajamani M.Kulkarni and Preetam R. Kalburgimath	2020	OCR converted text is was 99% accurate and was successfully converted to audio.	Applied python audio synthesizer on an image converted into text using Tesseract OCR engine.	Conversion of an input text generated from images of printed text analyzed by OCR to audio.
Relation Networks for Object Detection [7]	Han Hul, Jiayuan Gu Zheng Zhang, Jifeng Dai, Yichen Wei1	2018	A new module for object detection called the relation network	CNN, logistic regression, a regression model	The relation network is a promising new approach to object detection. It is simple to implement and can be used to improve the performance of existing object detectors.

3.3 Conclusion

Image descriptions in real-time to assist the visually impaired people is an active research area. Most work in this domain has been done on pre-captured images, but the area of real-time video capturing and generating useful descriptions is still being extensively researched. Visionary technology like this exists mostly in smart glasses or hardware form but its application in mobile devices has limited features. Our overall project consists of four major tasks: multiple object detection and tracking, object recognition, natural language descriptions and iteration of audio for conversion of text-to-speech. In the above research articles, multiple object detection and tracking can be done using multiple deep learning algorithms including YOLO which performs one-stage object detection, and RCCN which performs two-stage object detection. To recognize and identify these detected objects we can use computer vision algorithms such as PCN, SIRP, and SURF. These algorithms help in the recognition of objects by comparing their extracted key points with key points of pre-existing images present in the database. After object recognition, AT-RNN can be used to provide precise descriptions of the objects, ensuring visually impaired individuals receive more detailed and accurate information about their surroundings. After using the natural language description process, we learned that textual output can be converted into audio using various open source codes such as Python speech-to-text synthesizer.

Chapter 4 Software Requirement Specifications

This chapter lists all the features that our project will provide and all the requirements (hardware/software) that will be required for developing and using the project. In addition to all this the use cases for all the features and the user interface screens are also given in this section. Further detailed risk analysis provided at the end of this chapter.

4.1 List of Features

Following are the important features of our system.

- Real-time object detection using the mobile camera.
- Continuous tracking of detected objects for a seamless user experience.
- Text-to-speech functionality to provide auditory feedback on detected objects.
- Clear and concise voice instructions to guide users.
- Ability to detect and recognize various types of objects simultaneously.
- Differentiate between objects based on size, shape, and color.
- Provide a detailed verbal description of the user's surroundings.
- Describe the layout, nearby objects, and any potential obstacles.

All the above mentioned features which are part of our application aim to help blind people work efficiently in any environment.

4.2 Functional Requirements

These are the functional requirements for our application:

4.2.1 Object Detection

- The system shall be able to detect a variety of objects in real-time using the mobile device's camera.
- The detection system shall have a high accuracy rate.

4.2.2 Object Tracking

- Once an object is detected, the system shall be able to track its movement continuously.

4.2.3 Voice Feedback

- The system shall be able to convert visual information into auditory feedback using text-to-speech technology.
- Voice feedback shall be clear, concise, and easily understandable.

4.2.4 Multi-Object Recognition

- The system shall be capable of recognizing and processing multiple objects simultaneously.

4.2.5 Scene Description

- The system shall provide a comprehensive description of the user's surroundings, including layout and nearby objects.
- The system shall be able to pause audio.
- The system shall be able to resume audio playback.

4.3 Quality Attributes

Our system will have the following quality attributes listed below:

- Real-time Performance
- Usability
- Reliability
- Scalability
- Compatibility

4.4 Non-Functional Requirements

Following are the non-functional requirements of our application:

4.4.1 Real-time Performance

Our application shall process and analyze mobile's camera input in real-time. We intend to provide the user with timely and up-to-date information to improve the user experience. Our application's timeliness is measured in terms of:

- Timely detection and tracking of the objects
- Timely generating suitable scene descriptions and audio feedback

4.4.2 Usability

Our application shall have a user-friendly interface, making it easy to use and understand for the visually impaired users. The interface will be simple and straightforward. To make the application more user friendly, sound will be integrated with generated descriptions to increase the user-friendliness of the application.

4.4.3 Reliability

We want to make our application robust and reliable, providing consistent and accurate information across different environments and lighting conditions. The accuracy of object detection and tracking is crucial to ensure that the application provides reliable and precise information about the user's surroundings. Our system shall be able to recognize the objects and track them correctly to provide accurate descriptions.

4.4.4 Scalability

Our application shall be able to work when the complexity of surroundings is increased in object detection and tracking, ensuring scalability as the number of supported objects or features increases. When our application will be used in different environments, its workload will increase. So, our application shall produce reliable and quick responses, regardless of the number of supported objects or the complexity of the detection and tracking processes.

4.4.5 Compatibility

Ensure compatibility with a variety of smartphones and their respective cameras, optimizing performance across different devices. Our goal is to make it accessible for a large population so it should be compatible with most of the smart phones the user owns.

4.5 Assumptions

The following assumption have been made for this system:

- We assume that the user's mobile has sufficient processing power to handle real-time image processing and object detection algorithms

- We assume that users have activated the blind mode on their mobile devices. This mode is used to read buttons, app locations on the home screen, and help visually impaired users in using the mobile devices.
- The system will be trained on a range of environments and scenes for object detection and tracking. However, extreme conditions such as low light, high glare, or crowded spaces might impact the accuracy of the system.
- The effectiveness of NLP may be influenced by the complexity of the scene.

4.6 Use Cases

Below is a collection of all the use cases for our system:

Table 4.1: Start Scene Description Use Case

User will start listening to the descriptions of his surroundings as soon as he click the camera button

<table>
<tr><td><b>Identifier</b></td><td colspan="3">UC1</td></tr>
<tr><td><b>Name</b></td><td colspan="3">Start Scene Description</td></tr>
<tr><td><b>Actor</b></td><td colspan="3">User</td></tr>
<tr><td><b>Priority</b></td><td colspan="3">High</td></tr>
<tr><td><b>Summary</b></td><td colspan="3">The user clicks the central button to start the process of identifying and tracking of objects in real-time using the mobile camera</td></tr>
<tr><td><b>Pre-Conditions</b></td><td colspan="3">The Vision Guide application is installed and the mobile device has a functional camera and speaker/headphones.</td></tr>
<tr><td><b>Post-Conditions</b></td><td colspan="3">The application provides continuous tracking and verbal feedback on the detected objects.</td></tr>
<tr><td><b>Special Requirements</b></td><td colspan="3">The mobile device should have access to camera permissions.</td></tr>
<tr><td colspan="4"><b>Basic Flow</b></td></tr>
<tr><td colspan="2"><b>Actor Action</b></td><td colspan="2"><b>System Response</b></td></tr>
<tr><td>1</td><td>Launches the Vision Guide app.</td><td>2</td><td></td></tr>
<tr><td>2</td><td>Points the mobile camera towards objects of interest and clicks the central button.</td><td>4</td><td>Identifies and tracks the objects, providing real-time auditory feedback.</td></tr>
</table>

Table 4.2: Stop Scene Description Use Case

User will stop listening to the descriptions of his surroundings as soon as he click the camera button

<table>
<tr><td><b>Identifier</b></td><td colspan="3">UC2</td></tr>
<tr><td><b>Name</b></td><td colspan="3">Stop Scene Description</td></tr>
<tr><td><b>Actor</b></td><td colspan="3">User</td></tr>
<tr><td><b>Priority</b></td><td colspan="3">High</td></tr>
<tr><td><b>Summary</b></td><td colspan="3">The user clicks the central button to stop the scene description process.</td></tr>
<tr><td><b>Pre-Conditions</b></td><td colspan="3">The Vision Guide app is installed and running. The scene description process is currently active.</td></tr>
<tr><td><b>Post-Conditions</b></td><td colspan="3">The application ceases to provide scene descriptions.</td></tr>
<tr><th colspan="4">Basic Flow</th></tr>
<tr><th colspan="2">Actor Action</th><th colspan="2">System Response</th></tr>
<tr><td>1</td><td>The user presses the stop audio button.</td><td>2</td><td>The app stops the scene description process.</td></tr>
</table>

4.7 Hardware and Software Requirements

The following section enlists the hardware and software requirements for the development and deployment of our project.

4.7.1 Hardware Requirements

Hardware requirements of the project are as follows:

- A machine to act as server for our project
- A stable internet connection
- A smartphone with the following features:
  - A camera of 12 MP or more
  - Suitable processing power (e.g., Snapdragon 6xx series or higher for Android)
  - RAM of at least 2GB for a reasonable baseline to ensure smooth multitasking and efficient processing of large image data
  - A minimum of 32GB of storage, but more is preferable for installing the application and storing potential offline data

  - Quality speakers or headphone output for delivering clear and understandable audio descriptions

4.7.2 Software Requirements

Following are the software requirements for our project:

- Kaggle to use models and data set for our application
- Github Desktop to maintain coordination of code
- StarUML to implement our system modeling
- Python 3.7 or greater to implement the model along with its libraries
- Anaconda and Jupyter notebook for working with Python
- React Native for mobile application development
- Figma for UI/UX designing of our application mobile interface

4.8 Graphical User Interface

The GUI of the system will look like this:

Figure 4.1: Vision Guide

This figure is the loading page of Vision Guide App

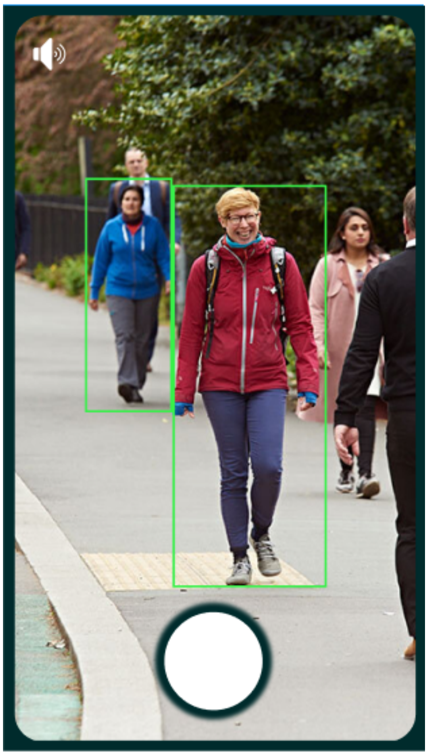

Figure 4.2: Home Page

This figure is the main video camera page of the Vision Guide Application

4.9 Risk Analysis

The risks that we might encounter during the project are as follows.

4.9.1 Technical Risks

Carrying out real-time object detection, tracking, and natural language processing might encounter technical issues, causing delays. Detecting objects accurately could be influenced by tricky or complex conditions like dim lighting, glare, or crowded scenes.

4.9.2 Performance Risks

The effectiveness of the application depends on users' mobile device capabilities, and differences in device specifications can affect the performance of the system. Moreover, relying on blind mode means, we assumes that users activate it and encounter no difficulties in its integration with the app.

4.10 Conclusion

To sum up Chapter 4, we've outlined the key features our project will offer, such as real-time object detection, continuous tracking, and text-to-speech feedback. We've also covered functional requirements like accurate object detection and tracking, clear voice feedback, and multi-object recognition. The system aims to be user-friendly, reliable, scalable, and compatible with various smartphones. We've made assumptions about users' devices and potential challenges. The hardware and software requirements needed for development are listed, including the need for a machine to act as a server, a stable internet connection, and specific smartphone features. We've also highlighted potential risks like technical and performance challenges that we'll be mindful of during the project. Overall, this chapter provides a road map for creating a practical and user-friendly application for visually impaired users.

Chapter 5 Proposed Approach and Methodology

This chapter outlines the proposed approach and methodology for the Vision Guide mobile application. The chapter delves into the core functionalities of the application, including scene description generation and its audio feedback.

5.1 Data Preprocessing

Real-time object detection poses unique challenges compared to offline detection due to the continuous stream of data and the constraints of processing speed. Data preprocessing plays an even more critical role in real-time scenarios, as it ensures that the application can handle the incoming data efficiently and provide accurate object detection results with minimal latency.

5.1.1 Color Space Conversion

We shall convert the incoming video frames to a consistent color space for standardizing the input. In real-time video processing, we shall convert the frames to HSV to enhance the efficiency of subsequent image processing operations.

5.1.2 Frame Resizing

We shall resize the video frames to a standardized resolution to facilitate efficient processing. This not only ensures consistency in the input data but also helps optimize the computational load of the object detection and tracking algorithms, particularly when dealing with resource-constrained devices.

5.1.3 Frame Enhancement

We shall implement image enhancement techniques to improve the visibility of objects within the video stream. This may include contrast adjustment, histogram equalization, or adaptive filtering to enhance features that are crucial for object detection algorithms.

5.1.4 Noise Reduction

Real-world video streams often contain noise, which can adversely affect the accuracy of object detection. We shall apply noise reduction techniques, such as Gaussian blurring, to mitigate the impact of noise and ensure a cleaner input for subsequent processing steps.

5.1.5 Optical Flow Analysis

We shall incorporate optical flow analysis to detect motion patterns within consecutive frames. This information can be valuable for predicting the trajectory of moving objects, aiding in more accurate tracking.

5.1.6 Frame Normalization

We shall normalize the video frames to account for variations in lighting conditions or exposure. Normalization ensures that the object detection and tracking algorithms operate consistently across different environmental settings, contributing to the robustness of the system.

5.1.7 ROI (Region of Interest) Identification

We shall identify and define regions of interest within the video frames. Focusing on specific areas of the frame relevant to the object detection task can improve processing speed and reduce computational overhead.

5.2 Object Detection

Object detection is the cornerstone of the Vision Guide application, as it enables the application to identify and classify objects within the user's field of view. This is achieved through the integration of a state-of-the-art object detection algorithm. This algorithm utilizes deep learning techniques to analyze the image captured by the smartphone's camera and identify objects within the scene.

We shall use a pre-trained CNN model called mobilnet to extract the features from the optic flow frame given as input to the model. Mobilnet is trained on the readily available and popular data set ImageNet. Mobilnet has proven to be the most effective model in our research work. We previously used simple CNN (not pre-trained) for feature extraction, but it did not prove to be as effective as mobilnet.

5.2.1 LSTM and Bi-LSTM

A major problem with feed-forward neural networks is their inability to remember information. So, every time the neural network must make a computation, it will have to start from scratch. RNNs solve this problem and allow the persistence of information, making them particularly adept at sequence modeling tasks. They, however, struggle with long-term dependencies and are unable to retain information for long. Long short-term memory, more colloquially known as LSTM, is a special type of RNN designed to cater to long-term dependencies.

We shall use LSTM in our application as LSTMs swap out RNNs' single-layer repeating module for a

4-layer repeating module, using sigmoid-layer gates as controls over modifications to cell state. Forget gate, input gate, and output gate are the three gates used. While the forget gate controls what information will be discarded, the input gate is responsible for deciding which values will be updated. At the risk of stating the obvious, the output gate determines the portion of cell state that will be fed into the next repeating module.

Bi-LSTM or Bi-directional LSTM can just be thought of as putting two LSTMs together. While one persists information from the past, the other preserves information from the future. This allows them to understand the context better and make more informed predictions.

5.3 Object Tracking

For the object detection we shall perform the following steps:

- Employ a tracking algorithm to maintain the identities of detected objects across consecutive frames, enabling the app to track their movement over time.
- Utilize Kalman filtering to handle object motion and predict their future trajectories, providing users with anticipatory guidance.
- Implement object disappearance and reappearance detection to ensure that tracked objects are not lost when temporarily occluded or outside the camera's field of view.

5.4 Scene Description Generation

Once objects have been detected, the Vision Guide application generates a detailed and informative description of the scene such as a person walking towards a door or a car approaching an intersection. This description is conveyed to the user through high-quality text-to-speech synthesis, ensuring that the information is accessible and understandable. The description includes the type, location, and relative positions of the detected objects, providing a comprehensive understanding of the user's surroundings such as ”the red ball is rolling to the left” or ”the car is approaching from behind.”

5.5 Output Layer

The main purpose of this layer is to produce label prediction of the video frames. We shall use the predictions done at each time step by LSTM to predict the final label of the video frames. In the output layer we shall apply a dense layer with the softmax activation function. Softmax computes the probability for each time-step prediction. We then take the average of these probabilities across all frames and choose the label which is the most probable one. The output layer would have following:

- It provides a list of detected objects within the user's field of view, including their class labels (e.g., person, car, tree) and their relative positions
- It generates detailed descriptions of the detected objects, including their appearance, size, color, and any other distinguishing features.
- It provides a comprehensive overview of the scene, incorporating information about the detected objects, their arrangement, and any relevant landmarks or contextual details.
- It updates the output layer continuously as the user navigates, providing real-time information about new objects, changes in object positions, and any significant changes in the environment.

5.6 Conclusion

In the above-mentioned methodology, we proposed to process the video frame by frame and compute optic flow of each frame. We then passed those frames to a time-distributed 2d convolution layer where they are processed independently to extract spatial features. Max pooling is then applied, and the features vector is flattened to 1D array. These vectors of features are passed to LSTM at each time step where prediction is made for each frame. These predictions are then combined after applying a dense layer with softmax to get a final single output label for the entire video. As we move forward with our project we may change or tweak some hyperparameters to find optimal ones for our model through experimentation. We may even experiment by changing our classification model or try new things in data preprocessing to get more and more accurate results.

Chapter 6 High-Level and Low-Level Design

In this section, we will discuss high-level and low-level designs of the Vision Guide.

6.1 System Overview

Our application aims to transform the experience for visually impaired users by providing real-time audio descriptions of their surroundings. The system is designed to empower users through state-of-the-art technologies, such as object detection, image processing, and natural language processing (NLP).
The key functionality centers around real-time object detection and tracking using advanced image processing methods. The system utilizes segmentation, feature extraction, and classification algorithms to accurately identify and track objects in various environments. The integration of the NLP algorithm ensures that the user receives clear and contextually relevant audio descriptions of the identified objects in real time.

6.2 Design Considerations

This section outlines various challenges and considerations that must be tackled or resolved before undertaking the development of a comprehensive design solution.

6.2.1 Assumptions and Dependencies

The following are the assumptions and dependencies of our application.

6.2.1.1 Assumptions

- We assume the user's mobile has a feature called blind mode that makes it easy to read buttons and find apps on the home screen.
- We believe people using the app have some knowledge of how blind mode works and are familiar with helpful features on their mobile devices
- The app requires a strong and reliable internet connection to continuously provide live audio descriptions and updates.

6.2.1.2 Dependencies

- The software needs to work well with the software and hardware of users' mobile devices.

- It uses the features and functions provided by the operating system of mobile phones to make it easier for its users to use.
- The success rate of the app depends on users characteristics and how well they understand the descriptions it gives in everyday language.

6.2.2 General Constraints

The software design is influenced by various constraints that impact its development and performance:

- The application capabilities are limited by the specifications and constraints of the user's mobile device and its operating system.
- The effectiveness of the software is influenced by the user's surroundings, lighting conditions, and the diversity of environments in which the object detection and tracking occur.
- The application must meet performance standards to ensure real-time processing and accurate audio descriptions.
- The design must comply with interface and protocol requirements to facilitate smooth communication between the application and the user's mobile device.

6.2.3 Goals and Guidelines

In crafting the Vision Guide app, we focus on a few key points to make things work well:

- We believe in keeping things easy to understand and use. This way, everyone, including those with visual challenges, can easily get to understand the app.
- We really care about speed, especially when it comes to quickly recognizing and describing things in real-time. This ensures users get timely and accurate info about what's around them.

6.2.4 Development Methods

For this project, we have chosen the agile methodology as our development approach. Agile is used to mitigate risks and improve performance incrementally by testing each module. This approach allows us to break down the work into different sections and subsections and achieve specific goals within defined time frames. Additionally, it provides flexibility to enhance performance and implement necessary changes with each increment, ensuring we achieve maximum accuracy in our results.

6.3 System Architecture

The system architecture consists of the following flow of processes. The captured video will be converted into video frames, and each frame will go through the process of object detection and tracking, description module and text-to-speech conversion module.

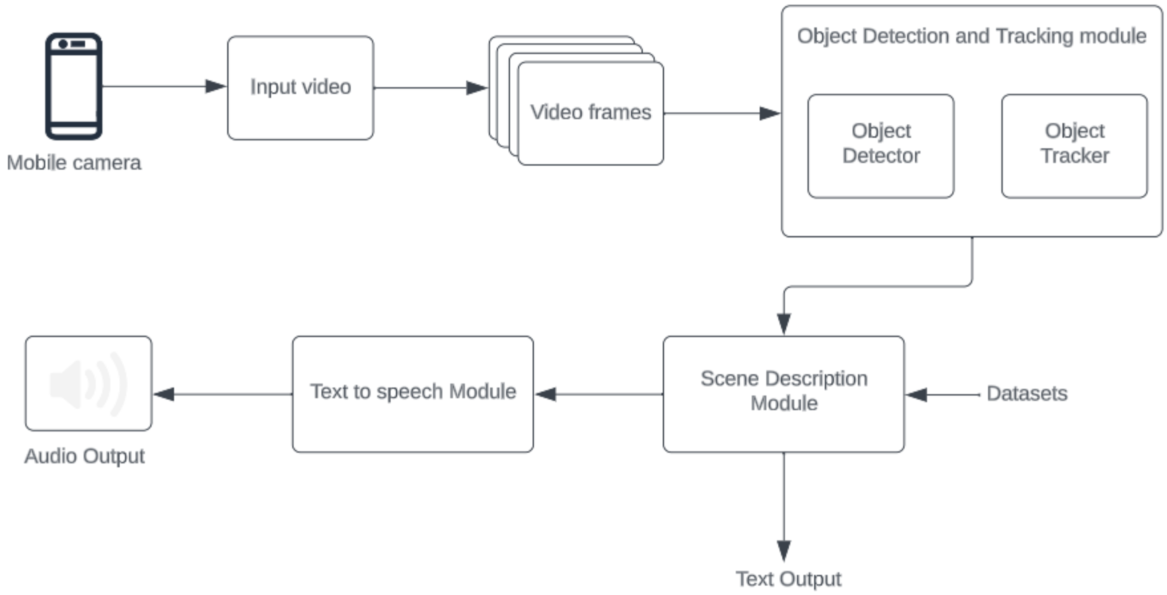

Figure 6.1: System Architecture Diagram

This figure represents System Architecture Diagram

6.4 Architectural Strategies

We describe the architectural strategies for our project below.

6.4.1 Technologies used

We intend to develop our mobile application using React Native, focusing only on the Android platform. For object detection and tracking we will use Open CV, a powerful open-source computer vision library, utilized for image processing tasks. We will use the CNN model MobileNet, fine-tuned for efficient object detection in real-time scenarios. For Natural language processing we will use LSTM and Bi-LSTM models defined in TensorFlow. To convert text-based descriptions into spoken words,we will use react native libraries for text-to-speech synthesis. We will mainly use the open-source model and technologies for our application.

6.4.2 Data set

We will train our object detection and tracking models on a pre-existing and widely used data set - ImageNet. ImageNet provides a diverse collection of labeled images covering a broad range of object categories. Moreover, we will use the real-world data collected during the application's usage for the fine-tuning and validation of our application.

6.5 Class Diagram

The classes, their relationship and data members and functions of each class is explain in the class diagram below

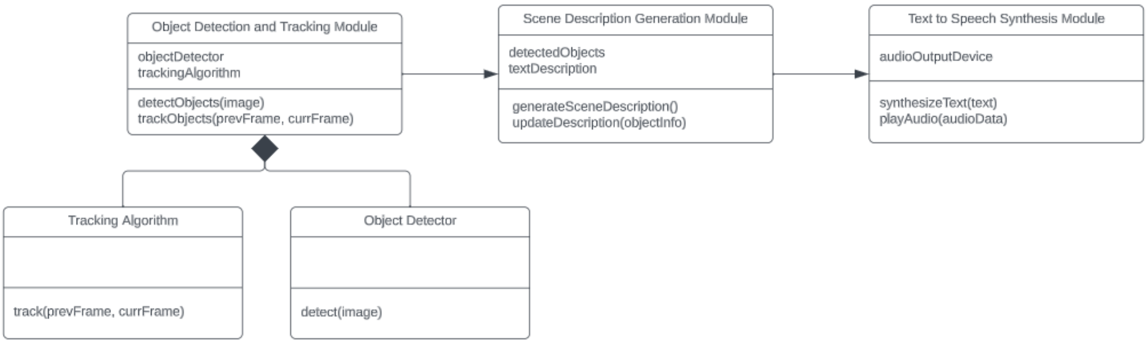
Object Detection and Tracking Module
objectDetector
trackingAlgorithm
detectObjects(image)
trackObjects(prevFrame, currFrame)
Scene Description Generation Module
detectedObjects
textDescription
generateSceneDescription()
updateDescription(objectInfo)
Text to Speech Synthesis Module
audioOutputDevice
synthesizeText(text)
playAudio(audioData)
Tracking Algorithm
track(prevFrame, currFrame)
Object Detector
detect(image)

Figure 6.2: Class Diagram

This figure represents Class Diagram

6.6 Sequence Diagrams

This section display the sequence diagram for every use case that has been shown

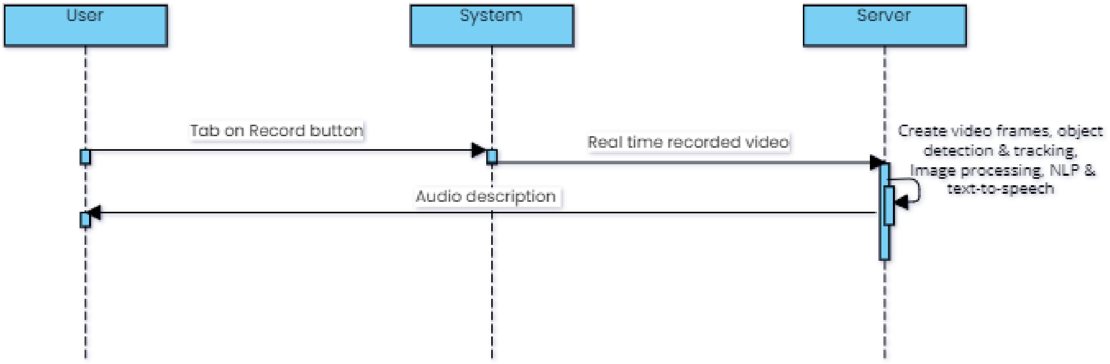
User
System
Server
Tab on Record button
Real time recorded video
Create video frames, object detection & tracking, Image processing, NLP & text-to-speech
Audio description

Figure 6.3: Start Scene Description

This figure represents the sequence of start scene description

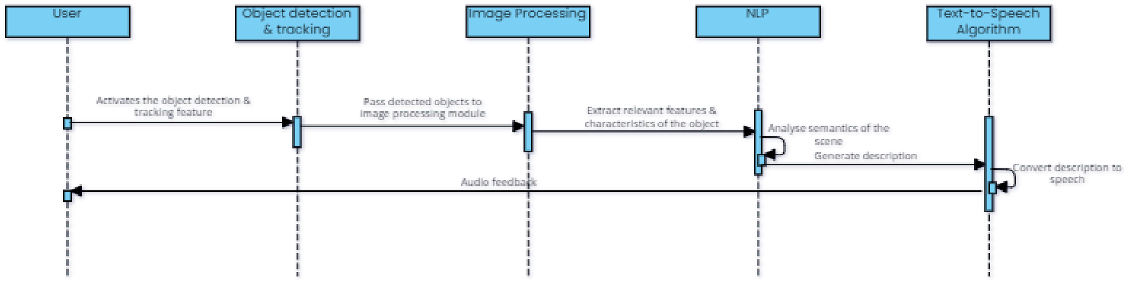
User
Object detection & tracking
Image Processing
NLP
Text-to-Speech Algorithm
Activates the object detection & tracking feature
Pass detected objects to image processing module
Extract relevant features & characteristics of the object
Analyse semantics of the scene
Generate description
Convert description to speech
Audio feedback

Figure 6.4: Classified Start Scene Description

This figure represents the classified sequence of start scene description

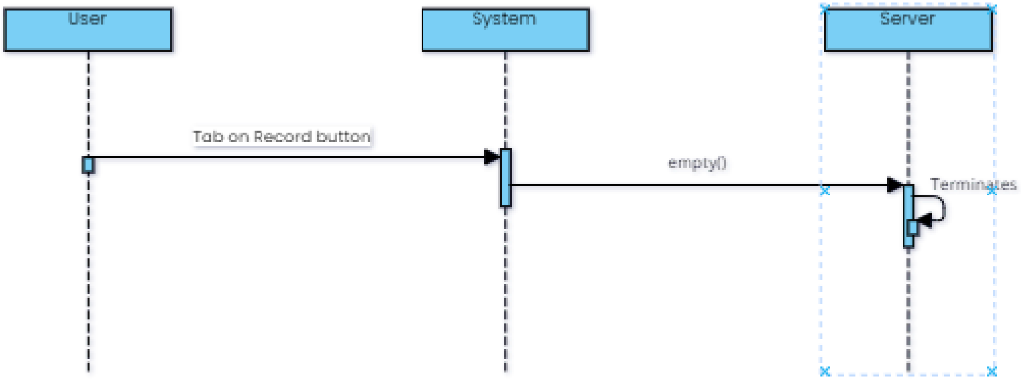

Figure 6.5: Stop Scene Description

This figure represents the sequence of stop scene description

6.7 Policies and Tactics

The following are the policies and tactics we will follow:

6.7.1 Code Style and Language Guidelines

We will adhere to the standard coding conventions for better readability and understanding of our application.

6.7.2 Testing

To test our application, we will use real-time video data from diverse environments.

6.7.3 Libraries Used

For the processing of video data, we will use Ultralytics library that contains our YOLO model. For further processing we use python libraries such as Open-CV and TensorFlow etc. For the audio-generation part we will use the react native libraries.

6.7.4 Development and Evaluation Platform

We will use the Jupyter notebook for the development and testing of our models. For mobile application development, we will use React Native framework and test it through Expo Go and Android studio.

6.8 Conclusion

This chapter outlines the architectural strategies, data set choices, and policies for our mobile application targeting enhanced experiences for the visually impaired users. Using React Native for Android development, our application incorporates OpenCV and TensorFlow for efficient object detection. Moreover, we use models for NLP. To convert text descriptions into spoken words, we use react native libraries. ImageNet serves as the primary training data set for our application. We've also examined the system's class diagram to outline primary components and functions of our system and sequence diagrams to explain the entire system flow.

Chapter 7 Implementation and Test Cases

In this chapter we will discuss the way we will be implementing the prototype of our system. Detailed description of each step of the model will be talked about in the given chapter. We will discuss the data sets we have used along with the preprocessing techniques as well as classification algorithms used in our system.

7.1 Implementation

The implementation phase of the Vision Guide application has involved the integration of various algorithms, leveraging specific platforms and APIs to ensure optimal performance. Below are the details of the implemented algorithms, the chosen platform, and utilized APIs, along with insights into the data set, preprocessing steps, and the use of MobileNet.

7.1.1 Object Detection Algorithm

For real-time object detection, we employed a state-of-the-art deep learning algorithm utilizing the MobileNet architecture. MobileNet is known for its efficiency, making it well-suited for applications with limited computational resources, such as mobile devices. After employing MobileNet for video processing in our mobile application, we transitioned to YOLO (You Only Look Once) due to several compelling reasons. YOLO offers superior real-time object detection performance compared to MobileNet, particularly in terms of speed and accuracy. Its single-pass architecture enables faster inference speeds, making it well-suited for applications requiring rapid object detection in videos. Additionally, YOLO excels in detecting small objects and handling occlusions, enhancing the robustness of our object detection system. The simplicity of YOLO's architecture also facilitates easier integration and customization within our application framework. Overall, the transition to YOLO enhances the efficiency and effectiveness of our video processing capabilities, ensuring a seamless and responsive user experience.

7.1.2 Platform and APIs

The following describes the platforms and APIs used in our application.

7.1.2.1 Platform

The implementation is focused on mobile devices, particularly smartphones.

7.1.2.2 Deep Learning Framework:

TensorFlow, a widely used open-source deep learning framework, was employed for implementing and deploying the object detection model. We have also used the Ulralytics library to use and test various versions of YOLO and decided to use the YOLO nano model for high accuracy and latency.

7.1.2.3 APIs

TensorFlow and Ultralytics provide APIs for building and deploying machine learning models, and specifically, we utilized the TensorFlow Lite API for mobile deployment.

7.1.3 Data set

The object detection model was trained on the ImageNet data set, a large-scale data set with a vast number of labeled images spanning numerous object categories. This diverse data set contributes to the model's ability to recognize a wide array of objects. We have trained our model on various datasets, including MS COCO containing 80 classes and then we also used the Open Images dataset containing 200 classes for better accuracy and prediction.

7.1.4 Preprocessing

First, the video frames were converted to the HSV (Hue, Saturation, Value) color space during preprocessing. This conversion enhances the efficiency of subsequent image-processing operations. Next, the video frames were resized to a standardized resolution, ensuring consistency in input data and optimizing computational load during object detection. Next, the image enhancement techniques, such as contrast adjustment and adaptive filtering, were applied to improve the visibility of objects within the video stream. Then, Gaussian blurring was employed for noise reduction, ensuring a cleaner input for more accurate object detection. And, optical flow analysis was integrated to detect motion patterns within consecutive frames, aiding in predicting the trajectory of moving objects.

7.1.5 MobileNet

MobileNet, a lightweight convolutional neural network, was used for feature extraction from video frames.MobileNet was trained on the ImageNet data set, allowing the model to learn features representative of various objects. The implementation process follows a systematic approach, combining efficient algorithms, appropriate platforms, and effective preprocessing steps to create a powerful and responsive Vision Guide application for visually impaired individuals. The utilization of MobileNet ensures a balance between accuracy and computational efficiency, making the application suitable for real-time deployment on mobile devices.

7.1.6 YOLO

In our application, YOLO plays a crucial role in enabling real-time object detection and tracking in video streams captured by the mobile device's camera. By leveraging YOLO's speed and accuracy, we can swiftly identify and classify objects within the user's surroundings, providing valuable insights and assistance to visually impaired individuals. YOLO's ability to detect objects of various sizes and handle occlusions ensures robust performance in diverse real-world scenarios, enhancing the effectiveness of our application in assisting users with navigation and understanding their environment.

7.1.7 React Native for Mobile Application

We have used React native for the front-end design of our application. We have executed the code on Expo Go and Android Studio for robust testing and editing. We designed our application pages as per the Figma design we created earlier. To execute the React Native code, we used Visual Studio code.

7.1.8 FAST API

The backend of our application, developed using FAST API, serves as the engine that drives the core functionalities of our system. It seamlessly integrates with the frontend interface and external services to process video streams captured by the user's device, leveraging advanced computer vision algorithms like YOLO for object detection and a language model for generating descriptive scene summaries.

Upon receiving a video URL from the frontend, the backend initiates the video processing pipeline, which involves several steps. First, the backend retrieves the video stream from the provided URL and streams it through the YOLO object detection model. YOLO efficiently analyzes each frame of the video, detecting and classifying objects present in the scene in real-time. This information is then passed to the next stage of the pipeline.

Once the objects have been identified and classified by YOLO, the backend utilizes a language model to generate descriptive summaries of the scene. This language model, trained on a vast corpus of textual data, has been fine-tuned specifically for our application domain to produce accurate and contextually relevant descriptions of the detected objects and their spatial relationships within the video frames.

After generating the scene description, the backend sends the processed data back to the frontend, where it is presented to the user in a user-friendly format. This seamless integration between the backend and frontend components ensures a smooth user experience, allowing visually impaired individuals to gain valuable insights into their surroundings in real-time.

7.2 Conclusion

This chapter has presented a detailed overview of the prototype implementation process for the Vision Guide application. It explains the detailed use of MobileNet,YOLO for object detection and labeling. The chosen platform, TensorFlow, Ultralytics and TensorFlow Lite ensures optimal performance on mobile devices. Additionally, the employed preprocessing techniques, such as HSV color space conversion, image resizing, and Gaussian blurring, contribute to enhancing the accuracy and efficiency of object detection. Lastly, it explains the use of React native for the front-end development of our application.

7.3 Test case Design and description

This section outlines the common attributes and structure of test cases. Main test cases utilized include the functionalities for turning the camera on and off, as well as the accuracy assessment of audio scene descriptions.

Table 7.1: Test Case for Camera On

This test case verifies the functionality of turning on the camera in the Vision Guide app.

Camera On Functionality			
01			
Test Case ID:	*01*	**QA Test Engineer:**	*Saba Naeem*
Test case Version:	*1.0*	**Reviewed By:**	*Dr. Zeeshan*
Test Date:	-	**Use Case Reference(s):**	*UC1*
Objective:	*To verify that the camera can be turned on successfully in the Vision Guide app.*		
Product/Ver/Module:	*Vision Guide - Version 1.0 - Camera Module*		
Environment:	*Mobile device with Vision Guide application installed.*		
Assumptions:	*The device has a functional camera.*		
Pre-Requisite:	*Vision Guide application is open and accessible.*		
Step No.	**Execution Description**	**Procedure Result**	
1	Open the Vision Guide application.	Application opens successfully.	
2	Navigate to the camera section.	Camera section is accessible.	
3	Tap on the camera button to turn it on.	Camera turns on, and audio description feedback starts	
Passed			

Table 7.2: Test Case for Camera Off

This test case verifies the functionality of turning off the camera in the Vision Guide app.

Camera Off Functionality			
02			
Test Case ID:	*02*	**QA Test Engineer:**	*Saba Naeem*
Test case Version:	*1.0*	**Reviewed By:**	*Dr. Zeeshan*
Test Date:	-	**Use Case Reference(s):**	*UC2*
Objective:	*To verify that the camera can be turned off successfully in the Vision Guide app.*		
Product/Ver/Module:	*Vision Guide - Version 1.0 - Camera Module*		
Environment:	*Mobile device with Vision Guide application installed.*		
Assumptions:	*The camera is currently on.*		
Pre-Requisite:	*Vision Guide application is open and accessible.*		
Step No.	**Execution Description**	**Procedure Result**	
1	Navigate to the camera section.	Camera section is accessible.	
2	Tap on the camera button	to turn it off, and audio description feedback st	
Passed			

Table 7.3: Scene Description Test Case

This test case verifies if the description is accurate or not

Verify Scene Descrption Accuracy			
03			
Test Case ID:	*03*	**QA Test Engineer:**	*Maida*
Test case Version:	*1.0*	**Reviewed By:**	*Dr. Zeeshan*
Test Date:	*2024-05-01*	**Use Case Reference(s):**	*UC2*
Objective:	*Verify the accuracy of scene descriptions provided by the Vision Guide application.*		
Product/Ver/Module:	*Vision Guide Application v1.0*		
Environment:	*Mobile device with camera, iOS/Android, Various lighting conditions*		
Assumptions:	*Assuming stable network connectivity and functioning camera.*		
Pre-Requisite:	*App installed, camera permission granted.*		
Step No.	**Execution description**	**Procedure result**	
1	Launch the application and enable Scene Description.	Scene descriptions accurately reflect the surroundings.	
2	Repeat Step 1 multiple times.	Scene descriptions remain consistent for the same environment.	
Passed			

Table 7.4: Object Detection Test Case

This test case will analyse if our model detect objects accurately

Object Detection Accuracy			
04			
Test Case ID:	*04*	**QA Test Engineer:**	*Fatima*
Test case Version:	*1.0*	**Reviewed By:**	*Dr. Zeeshan*
Test Date:	*2023-05-01*	**Use Case Reference(s):**	*UC2*
Objective:	*Validate the accuracy of object detection and tracking by the Vision Guide application.*		
Product/Ver/Module:	*Vision Guide Application v1.0*		
Environment:	*Mobile device with camera, iOS/Android, Various lighting conditions*		
Assumptions:	*Assuming stable network connectivity and functioning camera.*		
Pre-Requisite:	*App installed, camera permission granted.*		
Step No.	**Execution description**	**Procedure result**	
1	Activate Object Detection and Tracking feature.	Objects in the environment are accurately detected and tracked.	
2	Move the camera around to observe different objects.	Real-time tracking of objects is consistent and accurate.	
Passed			

7.4 Test Metrics

The test metrics outlined in Table 7.5 provide key insights into the effectiveness and coverage of the testing process. 7.5.

Table 7.5: Sample Test case Matric

Metric	**Value**
Number of Test Cases	4
Number of Test Cases Passed	4
Number of Test Cases Failed	0
Test Case Defect Density	0
Test Case Effectiveness	0

Chapter 8 User Manual

The user manual for our application is given below:

8.1 Introduction

Vision Guide is a mobile application designed to provide real-time object detection and tracking, along with auditory scene descriptions. This manual will guide you through the steps to use the application effectively for enhanced accessibility, particularly for blind users.

8.2 Getting Started

8.2.1 Installation

Download and install the application from the respective app store on your mobile device.

8.2.2 Permissions

Upon launching the application for the first time, grant necessary permissions for camera and audio access.

8.3 For Blind Users

8.3.1 Activating Blind Mode

For users who are blind or have low vision, activate the blind mode feature on your device for enhanced accessibility.

8.3.2 Using Phone Navigation

Utilize your phone's built-in accessibility features for navigation. For example, use TalkBack on Android to navigate to the application on your device's home screen.

8.3.3 Launching the Application

Open the application using your phone's navigation method. The camera will instantly activate.

8.4 Using the Application

8.4.1 Recording Scene Descriptions

Once the application is launched, locate the button positioned in the center of the screen using touch exploration. Tap the button once to start recording. You will hear an audio prompt confirming recording has begun. As you move your device's camera, the application will generate real-time scene descriptions through audio feedback. Explore your surroundings while the application describes detected objects, their attributes, and relations.

8.4.2 Stopping the Recording

To stop the recording, tap the same button again. You will hear a voice prompt saying ”Recording stopped.” The application will cease generating scene descriptions and conclude the recording.

8.5 Troubleshooting

8.5.1 Technical Issues

If you encounter any technical issues, restart the application and ensure your device has a stable internet connection.

8.5.2 Inaccurate Descriptions

If scene descriptions are inaccurate, adjust the camera focus or lighting conditions for better object detection.

Chapter 9 Experimental Results and Discussion

In this chapter, we present a comprehensive analysis of the experimental results obtained during the prototype implementation of the Vision Guide application. We developed two prototypes: one for scene description in predefined images, offering insights into scene description functionality, and another for real-time implementation of object detection and tracking. For the real time implementation of object detection and tracking we used the MobileNet architecture.

9.1 Dataset used

We used many different datasets for our prototype development: We used Flickr8k for the scene description prototype and ImageNet for implementing MobileNet architecture. While the Flickr8k data set served well for image captioning, the ImageNet data set was chosen for its extensive coverage of diverse objects and scenes, allowing more accurate results of object detection and tracking in real-time. Moreover, We have trained our model on various datasets, including MS COCO containing 80 classes and then we also used the Open Images dataset containing 200 classes for better accuracy and prediction.

9.2 Evaluation Metrics

The performance of the prototype was assessed using standard metrics, including object Detection Accuracy, and recall F1 score were computed to evaluate the accuracy of object detection in video frames. Then, the precision of object tracking across consecutive frames was measured to ensure the application's ability to maintain accurate identities of detected objects.

9.3 Object Detection Results

The integration of the MobileNet architecture for object detection on ImageNet data yielded promising results. The model demonstrated:

- The precision, recall, and F1 score consistently exceeded 80 percent, indicating a robust ability to identify and classify objects within the video frames.
- The detection algorithm effectively handled occlusions and partial object visibility, ensuring accurate identification even when objects were partially obscured.

9.4 Object Tracking Results

Object tracking, facilitated by Kalman filtering, showcased impressive capabilities:

- The Kalman filter accurately predicted the future trajectories of objects, providing users with anticipatory guidance.
- The tracking algorithm effectively managed situations where objects temporarily disappeared or reappeared in the camera's field of view.

9.5 Scene Description Generation

The scene description generation module, incorporating information about tracked objects, exhibited:

- Users received detailed descriptions of object interactions, relationships, and movement patterns, enhancing their understanding of the scene.
- The application utilized temporal language to describe object trajectories, contributing to a more natural and informative user experience.

9.6 Conclusion

The adoption of the Flickr8k data set yielded promising results for scene description for predefined images. For real-time object detection and tracking, MobileNet architecture resulted in a robust and accurate system for real-time object detection and tracking. Continuous refinement and adaptation will be integral to addressing emerging challenges and ensuring the application's effectiveness in diverse scenarios.

Chapter 10 Conclusion

In conclusion, our project, Vision Guide, has successfully achieved its objectives through a series of well-defined approaches and methodologies. We developed a prototype using the MobileNet architecture for real-time object detection, leveraging TensorFlow and TensorFlow Lite for building and deploying our models. We enhanced the accuracy and efficiency of object detection with data preprocessing techniques like HSV color space conversion, image resizing, and Gaussian blurring, and utilized React Native for the front-end design.

Transitioning to YOLO for improved object detection and tracking, we achieved high precision and an F1 score exceeding 80%, demonstrating robust identification and classification. For object tracking, Kalman filtering provided accurate trajectory predictions and effectively handled object disappearance and reappearance. Our application offers detailed descriptions of object interactions, relationships, and movement patterns, enhancing user understanding through temporal language.

Additionally, we experimented with various models and techniques for scene description, initially based on image input and later transitioning to video. This comprehensive approach ensured our application provides instant and contextually rich descriptions of the user's surroundings in real-time.

Our thorough documentation throughout the project has ensured clarity, traceability, and informed decision-making, facilitating effective communication and collaboration among developers and stakeholders. The Vision Guide application stands as a testament to our dedication to developing innovative solutions that enhance user experience through advanced technology.

Bibliography

[1] P. Strumillo, *Electronic Navigation Systems for the Blind and the Visually Impaired*. Lodz University of Technology Publishing House, 2012.

[2] "Aira and the envision glasses." Available: https://aira.io/envision/. [Accessed on: 2023-9-10].

[3] "New global blindness data." Available: https://www.orbis.org/en/news/2021/new-global-blindness/. [Accessed on: 2023-9-10].

[4] S. K. Pal, A. Pramanik, J. Maiti, and P. Mitra, "Deep learning in multi-object detection and tracking: state of the art," *Applied Intelligence*, vol. 51, pp. 6400–6429, 2021.

[5] M. Tan, R. Pang, and Q. V. Le, "Efficientdet: Scalable and efficient object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 10781–10790, 2020.

[6] K. Joseph, S. Khan, F. S. Khan, and V. N. Balasubramanian, "Towards open world object detection," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 5830–5840, 2021.

[7] H. Hu, J. Gu, Z. Zhang, J. Dai, and Y. Wei, "Relation networks for object detection," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 3588–3597, 2018.

[8] "Seeing AI." Available: https://www.microsoft.com/en-us/ai/seeing-ai. [Accessed on: 2023-9-10].

[9] "Envision AI." Available: https://www.letsenvision.com/. [Accessed on: 2023-9-10].

[10] "Taptapsee." Available: https://taptapseeapp.com/. [Accessed on: 2023-9-10].

[11] Y. Amit, P. Felzenszwalb, and R. Girshick, "Object detection," *Computer Vision: A Reference Guide*, pp. 1–9, 2020.

[12] L. Wang, T. Liu, B. Wang, J. Lin, X. Yang, and G. Wang, "Object tracking via imagenet classifica-

tion scores," in *2020 IEEE International Conference on Image Processing (ICIP)*, pp. 2101–2105, 2020.

[13] S. Guennouni, A. Ahaitouf, and A. Mansouri, "Multiple object detection using opencv on an embedded platform," in *2014 Third IEEE International Colloquium in Information Science and Technology (CIST)*, pp. 374–377, 2014.

[14] J. R. B. S. O. Mohamed Chaabane, Peter Zhang, "Deft: Detection embeddings for tracking," *arXiv preprint arXiv:2102.02267*, 2021.

[15] M. Han, A. Sethi, W. Hua, and Y. Gong, "A detection-based multiple object tracking method," in *2004 International Conference on Image Processing, 2004. ICIP '04.*, vol. 5, pp. 3065–3068 Vol. 5, 2004.

[16] K. Matusiak, P. Skulimowski, and P. Strurniłło, "Object recognition in a mobile phone application for visually impaired users," in *2013 6th International Conference on Human System Interactions (HSI)*, pp. 479–484, 2013.

[17] H. Jabnoun, F. Benzarti, and H. Amiri, "Object recognition for blind people based on features extraction," in *International Image Processing, Applications and Systems Conference*, pp. 1–6, 2014.

[18] D. S. A, P. R. A.R, S. R, K. V.K.G, and A. G, "Hot glass - human face, object and textual recognition for visually challenged," in *2017 2nd International Conference on Computing and Communications Technologies (ICCCT)*, pp. 111–116, 2017.

[19] R. Zhang, F. Meng, Y. Zhou, and B. Liu, "Relation classification via recurrent neural network with attention and tensor layers," *Big Data Mining and Analytics*, vol. 1, no. 3, pp. 234–244, 2018.

[20] D. Wang, J. Su, and H. Yu, "Feature extraction and analysis of natural language processing for deep learning english language," *IEEE Access*, vol. 8, pp. 46335–46345, 2020.

[21] P. Manage, V. Ambe, P. Gokhale, V. Patil, R. M. Kulkarni, and P. R. Kalburgimath, "An intelligent text reader based on python," in *2020 3rd International Conference on Intelligent Sustainable Systems (ICISS)*, pp. 1–5, 2020.

[22] C. Papageorgiou and T. Poggio, "A trainable system for object detection," *International Journal of Computer Vision*, vol. 38, pp. 15–33, 2000.