

**VIRTUAL SHOPPING ASSISTANCE FOR BLIND PEOPLE
WITH RECCOMADATION APPORACH**

Submitted in partial fulfilment of the required for the award of
Masters of Computer Science

by

LALITHA LAKSHMI D

(REGISTER NO 42734009)



**DEPARTMENT OF COMPUTER SCIENCE
SCHOOL OF SCIENCE AND HUMANITIES**

**SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY**

(DEEMED TO BE UNIVERSITY)

Accredited with Grade “A++” by NAAC

JEPPIAR NAGAR, RAJIV GANDHI SALAI, CHENNAI 600 119

APRIL 2024



SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Accredited "A++" Grade by NAAC | 12B Status by UGC Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the bonafide work of **LALITHA LAKSHMI D (42734009)** who carried out the project entitled "**VIRTUAL SHOPPING ASSISTANCE FOR BLIND PEOPLE WITH RECCOMADATION APPORACH**" under our supervision from January 2024 TO April 2024.

Aishwarya B
Internal Guide

Ms. AISHWARYA B, M.C.A., Assistant Professor

Rekha Chakravarthi
Dean And Head Of The Department

Dr. REKHA CHAKRAVARTHI M.E., Ph.D.,

Submitted for Viva voce examination held on 20 - 04 - 2024

Shiv
Internal Examiner

Dish
External Examiner

DECLARATION

I am, **LALITHA LAKSHMI D (42734009)** here by declare that the Project Report entitled **"VIRTUAL SHOPPING ASSISTANCE FOR BLIND PEOPLE WITH RECOMMENDATION APPROACH"** done by us under the guidance of **Ms. AISHWARYA B, M.C.A., Assistant Professor** Department of Computer Science at **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY**, Jeppiaar Nagar, Rajiv Gandhi Salai, Chennai-600 119 is submitted in partial fulfilment of the requirements for the award of Master of Science degree in Computer Science.

DATE: 20-4-2024

PLACE: CHENNAI

D. Lalitha Lakshmi
LALITHA LAKSHMI D

SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to Board of Management of **SATHYABAMA INSTITUTE OF SCIENCE AND TECHNOLOGY** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. REKHA CHAKRAVARTHI M.E., Ph.D., Dean of Arts and Science** for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my project guide **Ms. AISHWARYA B, M.C.A., Assistant Professor, Department of Computer Science** for their valuable guidance, suggestions and constant encouragement paved way for the successful completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff member of the Department of Computer Science who were helpful in many ways for the completion of the project

LALITHA LAKSHMI D

ABSTRACT

Independent shopping is one of the most functionally challenging tasks for visually impaired and blind individuals. Mainly identifying the best outfits for the concern blind people and buying them is not possible. Thus the proposed solution focuses on creating a personal independent shopping system with a recommendation approach for blind people. Here the concern person should stand in front of the camera where the camera identifies the physic of the blind people. After getting out the physical status the recommendation of outfits according to them will be recommended. Thus the people can wear their best outfit. On identifying their physical existence Artificial Neural Network (ANN) is used to get a 70% of physic status. The implementation carried with an assistance environment where the blind people can make a query of navigation regarding the dress they wish to buy. A trained labeled database with recommendation and assistance is done where HMM based data trained extraction is done. Thus the blind people can easily shop according to their wish.

TABLE OF CONTENT

CH.NO	TITLE	PAGE. NO
1	INTRODUCTION	1
1.1	OVERVIEW	1
1.2	OBJECTIVE OF THE SYSTEM	2
1.3	PROBLEM STATEMENT	2
1.4	PROPOSED SYSTEM	3
2	LITERATURE SURVEY	4
3	SYSTEM DESIGN	11
3.1	UML DIAGRAMS	11
3.1.1	Use Case Diagram	11
3.1.2	Sequence Diagram	12
3.1.3	Activity Diagram	13
3.1.4	Package Diagram	14
3.1.5	Component Diagram	16
4	SYSTEM ARCHITECTURE	18
4.1	Architectural Diagram	18
4.2	Architectural Description	19
4.3	SYSTEM REQUIREMENT	20
5	SYSTEM IMPLEMENTATION	21
5.1	MODULES	21
5.2	MODULES DESCRIPTION	21
5.2.1	Video Acquisition	21
5.2.2	Frame Conversion	22
5.2.3	Pre-Processing	23
5.2.4	Feature Extraction	24
5.2.5	Physic Identification	25
6	CODING AND SCREENSHOT	27
6.1	SAMPLE CODING	27
6.2	SAMPLE SCREENSHOT	47
7	CONCLUSION AND FUTUREWORK	51
7.1	CONCLUSION	51
7.2	FUTURE ENHANCEMENT	52
8	REFERENCE	53

LIST OF FIGURES

FIGURE NO	TITLE	PAGE NO
3.1	Use case diagram	12
3.2	Sequence Diagram	13
3.3	Activity Diagram	14
3.4	Package Detection	16
3.5	Component diagram	17
4.1	System Architecture	18
5.1	Video acquisition	22
5.2	Frame conversion process	22
5.3	Pre-processing system	23
5.4	Feature Extraction	24
5.5	HMM conversion system	26
6.2.1	Home page	47
6.2.2	Face detection	48
6.2.3	Feature Extraction	49
6.2.4	Recommendation	50

CHAPTER 1

INTRODUCTION

1.1 OVERVIEW

Online shopping, by reducing the needs for traveling, has become an essential part of lives for people with visual impairments. However, in HCI, research on online shopping for them has only been limited to the analysis of accessibility and usability issues. To develop a broader and better understanding of how visually impaired people shop online and design accordingly, we conducted a qualitative study with twenty blind people. Our study highlighted that blind people's desire of being treated as ordinary had significantly shaped their online shopping practices: very attentive to the visual appearance of the goods even they themselves could not see and taking great pain to find and learn what commodities are visually appropriate for them. This paper reports how their trying to appear ordinary is manifested in online shopping and suggests design implications to support these practices.

Online shopping is increasingly popular and is especially meaningful for people with disabilities who are inconvenient to travel. It enables them to shop easily without going out and greatly enhances their sense of independence and self reliance. It is especially true for visually impaired people. By using screen reader software, they could just read the commodity information directly from the screen without travel, greatly relieving navigation and object recognition issues commonly associated with their offline shopping experiences

1.2 OBJECTIVE OF THE SYSTEM

The main objective of our system is to provide all the information to blind people that they may need to comfortably use public spaces. Our system improves the autonomy, and thus self-esteem, of visually impaired people by enabling them to explore public spaces without the need to ask for help. Another valuable feature is that the user's device is the personnel smartphone, dispensing the cost of learning to use a new device. The whole shopping is equipped with Bluetooth BLE devices that provide shop and services information. The Bluetooth signal is also used to determine the user positioning. All visual content is translated into textual information and stored into a database. An internet of things infrastructure and synthesizer are used to retrieve the most important information from the database and translate it to voice tips

1.3 PROBLEM STATEMENT

In particular, we conducted a semi-structured interview with twenty blind people (7 with congenital blindness, and the rest with acquired blindness) in China. Aging from 17 to 48, they all use screen reader software on their smart phones and regularly shop online in their daily lives. From our study, we found that our blind participants paid great attention to the visual appearance of the commodities although they themselves could not see. Further analysis shows that this is part of their work to appear ordinary, which greatly shapes their online shopping practices in terms of finding out and learning about what appropriate for them. In this paper, we use the "ordinary" to mean how blind people would like to appear as ordinary members of the society, unnoticeable and not be treated differently, similar to "normal" used in some literatures, e.g., how patients with varied ability want to appear normal.

VISUALLY IMPAIRED PEOPLE

As for offline shopping, navigation and products recognition are the two main barriers for people with visual impairment. Much of related work has focused on navigation. ShopTalk, a wearable system consisted of a small OQO computer, a wireless barcode reader and a numeric keypad, and it uses synthetic verbal route directions and descriptions of the store layout to direct visually impaired people to the aisles with target products. A mobile application was also developed based on Shop Talk. Blind Shopping provided a navigation system. It combines a

white cane with a portable RFID reader attached to its tip, a set of RFID taglines distributed throughout the corridors of the supermarket and a Smartphone application which generates verbal navigation commands for users. RoboCart is a robotic shopping cart that can navigate for visually impaired customers and carry purchased items. Extensive work is also done for product recognition. It has been developed a camera-based prototype system that helps the blind recognize clothing patterns and identify clothing colors. Blind Shopping and Trinetra are QR code or barcode-based systems that allow visually impaired users to identify products by scanning the codes with their phones. Leveraged technologies such as Electronic Product Code (EPC) and Radio Frequency Identification (RFID) to develop a wearable wireless device capable of delivering product information at real time. Others attempt to integrate both navigation and product recognition solutions such as iShop

1.4 PROPOSED SYSTEM

In our proposed system, blind people shopping assistance with virtual reality environment is created. Here the blind people physical identity, color identity and gender identity is done using the camera interface and a best dress recommendation system is done.

The implementation is done using the ANN based classification which gives the good interface development. The HMM based system makes a voice assistance.

CHAPTER 2

LITERATURE SURVEY

2.1 TITLE: DEEPLY SUPERVISED SALIENT OBJECT DETECTION WITH SHORT CONNECTIONS

AUTHOR: Qibin Hou, Ming-Ming Cheng, Xiao-Wei Hu

YEAR : 2019

Significant advancements in salient object recognition have recently been made, mostly as a result of the rapid growth of convolutional neural networks (CNNs). Recently, many methods for salient object detection and semantic segmentation have been created using Fully Convolutional Neural Networks (FCNs). Compared to the generic FCN models that do not address the scale-space issue directly, there is still a great deal of space for development. For edge and boundary identification, the Holistically-Nested Edge Detector (HED) offers a skip-layer structure with deep supervision; nevertheless, it is unclear how much better HED performs for saliency detection. In this study, we introduce short connections to the skip-layer structures within the HED design, proposing a novel approach for salient item recognition. Our approach fully utilizes the multi-level and multi-scale characteristics that are derived from FCNs, offering more sophisticated representations at every layer a crucial requirement for segment detection. Our solution outperforms the current methods in terms of efficiency (0.08 seconds per image), efficacy, and simplicity, and yields state-of-the-art results on five extensively tested salient object detection benchmarks. Furthermore, we carry out a thorough examination of the impact of training data on performance. We offer a training set for impartial comparisons and future studies.

- **Techniques:** Convolution Neural Network
- **Merits:** The object in the path way is detected.
- **Problem Description:** The system identifies the implementation works with higher process but navigation is not done

2.2 TITLE: SALIENT OBJECT DETECTION VIA RANDOM FOREST

AUTHOR: Shuze Du, Shifeng Chen

YEAR: 2017

The identification of salient objects is crucial to the pre-processing of images. Large objects may not be accurately detected by current methods because they frequently ignore the contours of conspicuous items. Furthermore, their primary objective is to identify a solitary object. In this study, we use the object contour to identify the prominent item. Our suggestion is to utilize the random forest for calculating patch similarities and measuring patch rarity. Based on the rarity of the patch throughout the entire image, a global rarity map is computed. Using an active contour model, the salient object's approximate contour is retrieved based on this rarity map. Next, by comparing the similarities between patches inside and outside the contour, a local saliency map is produced.

- **Techniques:** Contour modeling generation
- **Merits:** The extraction of data will be done with object analysis navigation
- **Problem Description:** Shopping system can be added in this navigation

2.3 TITLE: 3D OBJECT PROPOSALS USING STEREO IMAGERY FOR ACCURATE OBJECT CLASS DETECTION

AUTHOR: Xiaozhi Chen, Kaustav Kundu

YEAR: 2019

The identification of salient objects is crucial to the pre-processing of images. Large objects may not be accurately detected by current methods because they frequently ignore the contours of conspicuous items. Furthermore, their primary objective is to identify a solitary object. In this study, we use the object contour to identify the prominent item. Our suggestion is to utilize the random forest for calculating patch similarities and measuring patch rarity. Based on the rarity of the patch throughout the entire image, a global rarity map is computed. Using an active contour model, the salient object's approximate contour is retrieved based on this rarity map. Next, by comparing the similarities between patches inside and outside the contour, a local saliency map is produced.

- **Techniques:** 3D object Stereo Imagery Classification
- **Merits:** The implementation can be done for product recognition with text
- **Problem Description:** Recognizing the product image does not further helps visually impaired people

2.4 TITLE: SHOPPING AND TOURISM FOR BLIND PEOPLE USING RFID AS AN APPLICATION OF IOT

AUTHOR: Tania Lorigo-Botrán, Unai Lopez-Novoa

YEAR: 2019

Based on the premise that visually impaired people would want to buy alone, without aid from others, this study attempts to develop a service for the blind. In order for this to occur, a blind individual entering a mall would need to be aware of the nearby stores and their offerings. The suggested software, which is based on Radio Frequency Identification (RFID) technology, attempts to help blind individuals shop by leading them to their destinations, displaying their position, and giving them helpful information about each surrounding store. Additionally, the program can provide a notification service based on the time, the anticipated amount of time needed for shopping, and the closing time of a certain store, allowing the user to know long enough to complete their desired purchases. A significant portion of the software generated and attained as a result of this research has proven successful.

- **Techniques:** RFID product identification in shopping
- **Merits:** The system helps in the identification of the shopping generation system
- **Problem Description:** RFID implementation is too costly

2.5 TITLE: MOBILE PHONE BASED INDOOR NAVIGATION SYSTEM FOR BLIND AND VISUALLY IMPAIRED PEOPLE: VUK VISIONLESS SUPPORTING FRAMEWORK

AUTHOR: László Árvai

YEAR: 2020

For those who are blind or visually challenged, getting around in cities can be difficult. It is nearly hard to navigate complicated structures like shopping malls, public transportation, or subways without assistance. In addition to traditional aids, a modern smartphone's enhanced sensory and computational capabilities make it a perfect choice for blind individuals seeking navigation assistance. A carefully thought-out software solution that combines state-of-the-art indoor localization techniques, man-machine interfaces, and unique user needs is needed to fully utilize this gadget. This paper describes the architecture, key components, and interfaces of an indoor navigation system specifically made for blind and visually impaired individuals. It also provides a brief overview of key algorithms and techniques.

- **Techniques:** VUK Visionless supporting framework
- **Merits:** The system gives assistance for people to make an outdoor walk
- **Problem Description:** Normal navigation is done using smart phone with low accuracy.

2.6 TITLE: A LOW COST SMART SHOPPING FACILITATOR FOR VISUALLY IMPAIRED

AUTHOR: M. Mathankumar, N. Sugandhi

YEAR: 2017

When someone is disabled, it means they are unable to take care of themselves. One of a person's limitations is visual impairment. Many strategies have been put out thus far to improve the quality of life for blind and visually impaired individuals. It's difficult for them to continue shopping at the grocery store without assistance from others. They can identify and buy their products in the supermarket with the help of the suggested system. To identify the merchandise, radio frequency identification, or RFID, technology is used. They will receive assistance from the audio instructions in the store depending on current events. The supermarket's billing system is automated to make it run more efficiently. The Zigbee transceiver is used by the autonomous billing system to send and receive product information. As a result, the supermarket's current queuing system is eliminated. This system's ultimate goal is to remove the need for outside assistance for visually impaired shoppers and give them access to a sophisticated and convenient setting. When this method is put into place, it makes shopping easier for blind individuals, saves customers' time, and increases business sales.

- **Techniques:** Zig bee transceiver with RFID technology
- **Merits:** The system assist the blind people by using RFID technology on identifying products
- **Problem Description:** The IOT implementation is too costly compared to all the systems

2.7 TITLE: OBSTACLE AVOIDANCE BASED ON COMPONENT SEPARATION-REFUSION ALGORITHM FOR BLIND PEOPLE

AUTHOR: Xinzhen Ren,Fulong Yao,Tianfang Zhou,Xiaogang Gu

YEAR: 2021

How to guarantee the basic rights and interests of blind people is an urgent problem that needs to be addressed. The visual assistance system for the blind people is a good choice. Obstacle detection in the visual assistance system for blind people is extremely important, however existing obstacle avoidance methods based on machine vision do not consider the impact of shadows in the environment. This paper proposes an obstacle avoidance method based on machine vision, includes an obstacle detection algorithm and a direction planning strategy, to assist blind people to move freely. Firstly, a component separation- refusion algorithm based on the HSV model is proposed to reduce the influence of shadows in the environment. Secondly, an obstacle avoidance method, includes an obstacle detection based on image blocks and a direction planning strategy, is developed to help blind people avoid obstacles both indoors and outsides. Finally, the proposed method is tested under various light intensities and venues, experiments show that this method can effectively help blind people avoid obstacles and improve mobility.

- **Techniques:** Component seperation refusion algorithm
- **Merits:** The obstacle analysis is done
- **Problem Description:** Analysis in shopping zone is not done

2.8 A REAL-TIME SYSTEM TO ASSIST BLIND PEOPLE THROUGH FACE RECOGNITION AND EMOTION DETECTION

AUTHOR: Kartheek Garapati,Sri Satya Maram

YEAR: 2023

Globally the number of people with visual impairment is very high and among them majority

have low vision. They face difficulties during social interactions. In order to assist visually impaired people a new system is introduced to identify the expression or emotion of the confronting person and hence enable a better communication. In this system the blind person uses camera assistance for image acquisition of the person who is interacting with him and by using an audio device the facial emotion is detected and conveyed. This system identifies the emotion such as happy, sad and surprise, thereby the blind person can have an effective social interaction. This system identifies all the basic emotions and hence is an efficient assistive tool for the blind.

- **Techniques:** Multi Layer Perceptron (MLP)
- **Merits:** Voice assistance with emotion detection
- **Problem Description:** Accuracy level is completely low

2.9 TITLE: AN INTELLIGENT AND MULTI-FUNCTIONAL STICK FOR BLIND PEOPLE USING IOT

AUTHOR: Shamim Ahmed,Md. Munam Shaharier,Srijan Roy

YEAR: 2022

The primary goal of this study is to aid blind people who do not require human assistance. People who cannot see the world have difficulty perceiving the obstacles in front of them, putting their lives in jeopardy. Using an intelligent and multipurpose bright blind stick will allow them to recognize their environment. In this study, we presented a low-cost, innovative, and versatile stick for blind individuals in our community adopting IoT to better their lives. A person can walk more confidently with the help of this brilliant blind stick. This stick can identify impediments in the route of blind people thanks to an Arduino UNO, ultrasonic sensor, IR sensor, water sensor, GPS and GSM modules, four-channel wireless remote control, and Buzzer. This stick identifies any object or barrier in front of the user, such as water, holes, stairs, or cars, and if any are detected, the system will inform the blind person via a buzzer. Our system's processing unit, the Arduino UNO, processes all input and delivers outputs. The walking stick has an ultrasonic sensor and an infrared sensor to identify obstructions in front of blind or disabled people. The water sensor can detect the presence of water and emit an alarm to inform people who are blind. Friends and family members of blind people may easily monitor them using our technology, and blind people can also use GPS and GSM to locate the stick by pressing

a button on the remote, which will cause the Buzzer to ring continuously for 5 seconds, indicating its location. We also researched and compared various current systems in this study, which led us to build a cost-effective, quick-responding, portable, intelligent, and bright multipurpose stick for blind persons utilizing IoT.

- **Techniques:** Arduino UNO
- **Merits:** Blind people walking assistance using IOT
- **Problem Description:** The hardware components is completely costly

2.10 TITLE: ADVANCED AUDIO AID FOR BLIND PEOPLE

AUTHOR: Savera Sarwar, Muhammad Turab, Danish Channa

YEAR: 2022

One of the most important senses in human life is vision, without it life is totally filled with darkness. According to WHO globally millions of people are visually impaired estimated there are 285 million, of whom some millions are blind. Unfortunately, there are around 2.4 million people are blind in our beloved country Pakistan. Human are a crucial part of society and the blind community is a main part of society. The technologies are grown so far to make the life of humans easier more comfortable and more reliable for. However, this disability of the blind community would reduce their chance of using such innovative products. Therefore, the visually impaired community believe that they are burden to other societies and they do not capture in normal activities separates the blind people from society and because of this believe did not participate in the normally tasks of society . The visual impair people mainly face most of the problems in this real-time The aim of this work is to turn the real time world into an audio world by telling blind person about the objects in their way and can read printed text. This will enable blind persons to identify the things and read the text without any external help just by using the object detection and reading system in real time. Objective of this work: i) Object detection ii) Read printed text, using state-of-the-art (SOTA) technology.

- **Techniques:** Stateof Art technology
- **Merits :** The systemshows a audio based assistance using object detection
- **Problem Description:** Normalroad side assistance is done

CHAPTER 3

SYSTEM DESIGN

3.1 UML DIAGRAMS

A UML (Unified Modeling Language) diagram is a graphical representation of a system or process using standardized symbols and notation. UML diagrams are widely used in software engineering and other fields to visually depict various aspects of a system, such as its structure, behavior, interactions, and architecture. There are several types of UML diagrams, each serving a different purpose and focusing on different aspects of the system being modeled. Some common types of UML diagrams include: Unified Modeling Language (UML) diagram for the proposed automated college timetable scheduling system. Here's an example of a high-level UML class diagram for the system:

3.1.1 Use Case Diagram

A Use Case Diagram is a type of UML (Unified Modeling Language) diagram that illustrates the interactions between actors (users or external systems) and a system under consideration. It provides a high-level view of the functionalities or features of a system from the perspective of its users. Use case diagrams are often used during the requirements analysis phase of software development to capture and communicate the system's behavior and requirements.

This diagram provides a high-level overview of the interactions between users, administrators, and the animal detection system, as well as the main functionalities supported by the system.

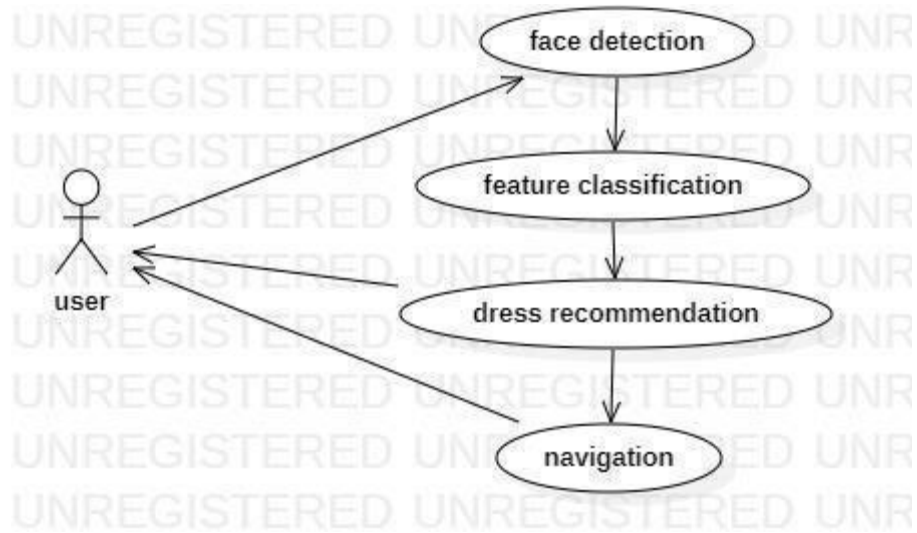


Fig 3.1 Use case diagram

3.1.2 Sequence Diagram

A Sequence Diagram is a type of UML (Unified Modeling Language) diagram that depicts the interactions between objects or components within a system over time. It illustrates the sequence of messages exchanged between these objects to accomplish a specific behavior or scenario. Sequence diagrams are particularly useful for visualizing the flow of control and collaboration among objects during the execution of a use case or scenario.

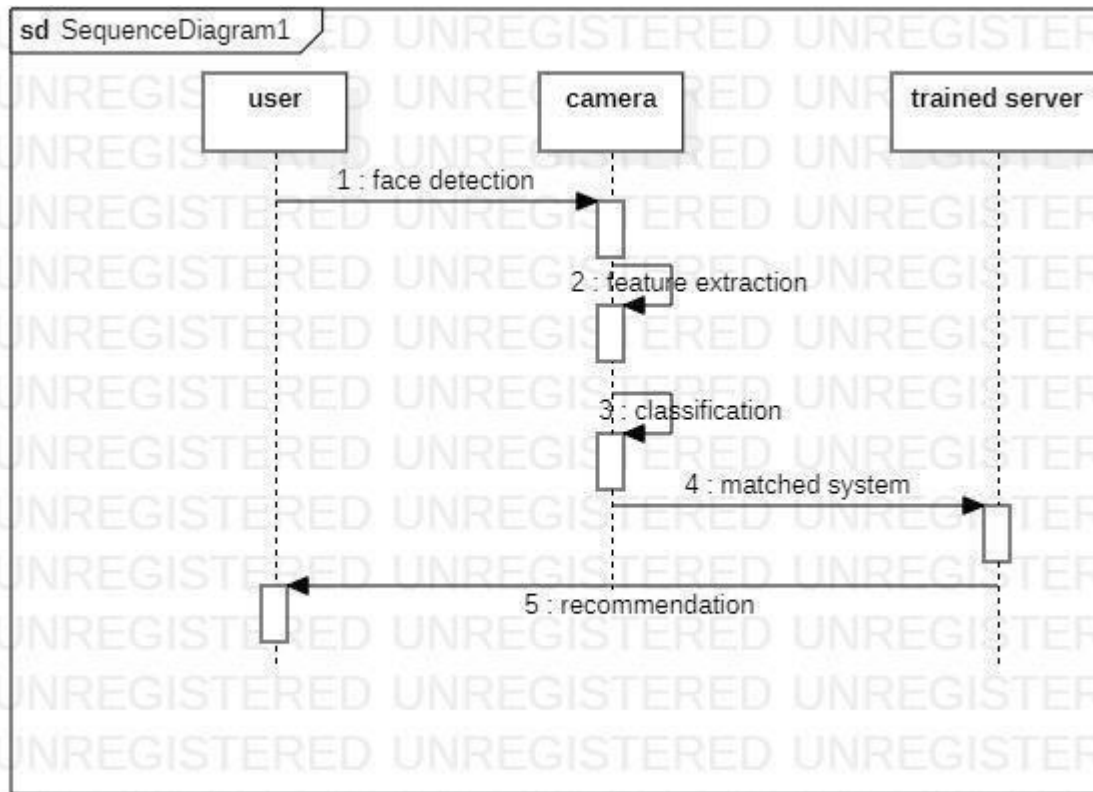


Fig 3.2 Sequence Diagram

3.1.3 Activity Diagram

An Activity Diagram is a type of UML (Unified Modeling Language) diagram that illustrates the flow of activities or actions within a system or a specific business process. It visually represents the sequence of actions, decisions, and transitions from one activity to another, allowing stakeholders to understand the workflow or behavior of a system or process.

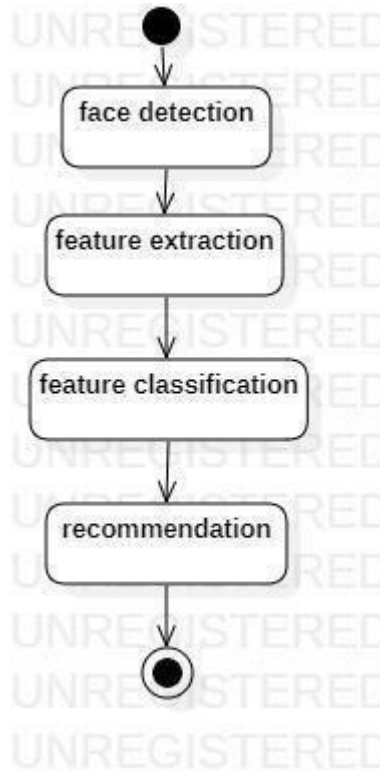


Fig 3.3 Activity Diagram

3.1.4 Package Diagram

A Package Diagram is a type of UML (Unified Modeling Language) diagram that represents the organization and dependencies among packages in a system. In software engineering, a package is a collection of related elements, such as classes, interfaces, and other packages, grouped together to provide a cohesive unit of functionality. Package diagrams provide a high-level overview of the structure and organization of a system, helping to manage complexity and promote modularity.

Key components of a package diagram include:

Packages: Represented by rectangles with folded corners, packages are used to encapsulate and organize related elements within a system. Each package may contain classes, interfaces, other packages, and dependencies.

Package Dependencies: Arrows between packages indicate dependencies between them. A dependency implies that elements in one package depend on elements in another package. Dependencies can be either directed or bidirectional.

Package Relationships: Different types of relationships between packages can be depicted, including:

Association: Indicates a semantic relationship between packages, such as a usage dependency.

Generalization: Represents a specialization/generalization relationship between packages, where one package specializes another.

Import: Signifies that one package imports elements from another package, allowing access to its contents.

Merge: Used to combine the contents of two or more packages into a single package.

Package Visibility: Packages may have different levels of visibility, such as public, private, or protected, which determine the accessibility of their contents to other packages.

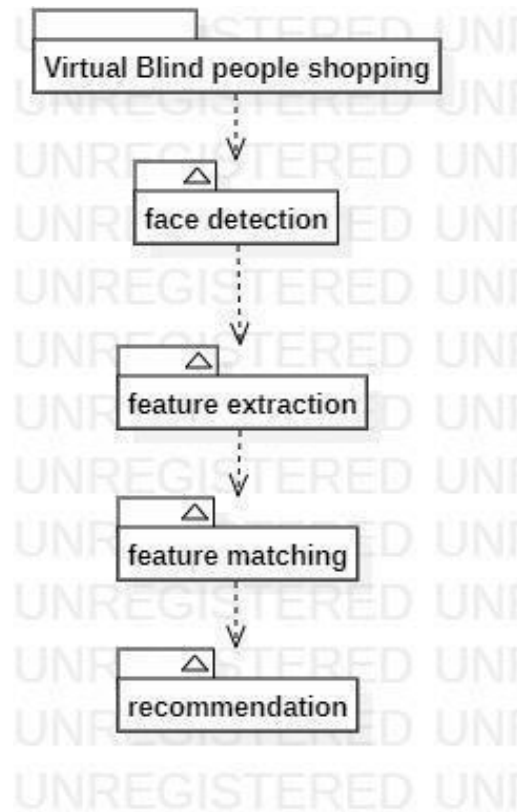


Fig 3.4 Package Detection

This package diagram provides a high-level overview of the package structure of the automated college timetable scheduling system, organizing classes and components into logical groupings based on their functionalities.

3.1.5 Component Diagram

Below is a simplified component diagram representing the components and their interactions in the automated college timetable scheduling system:

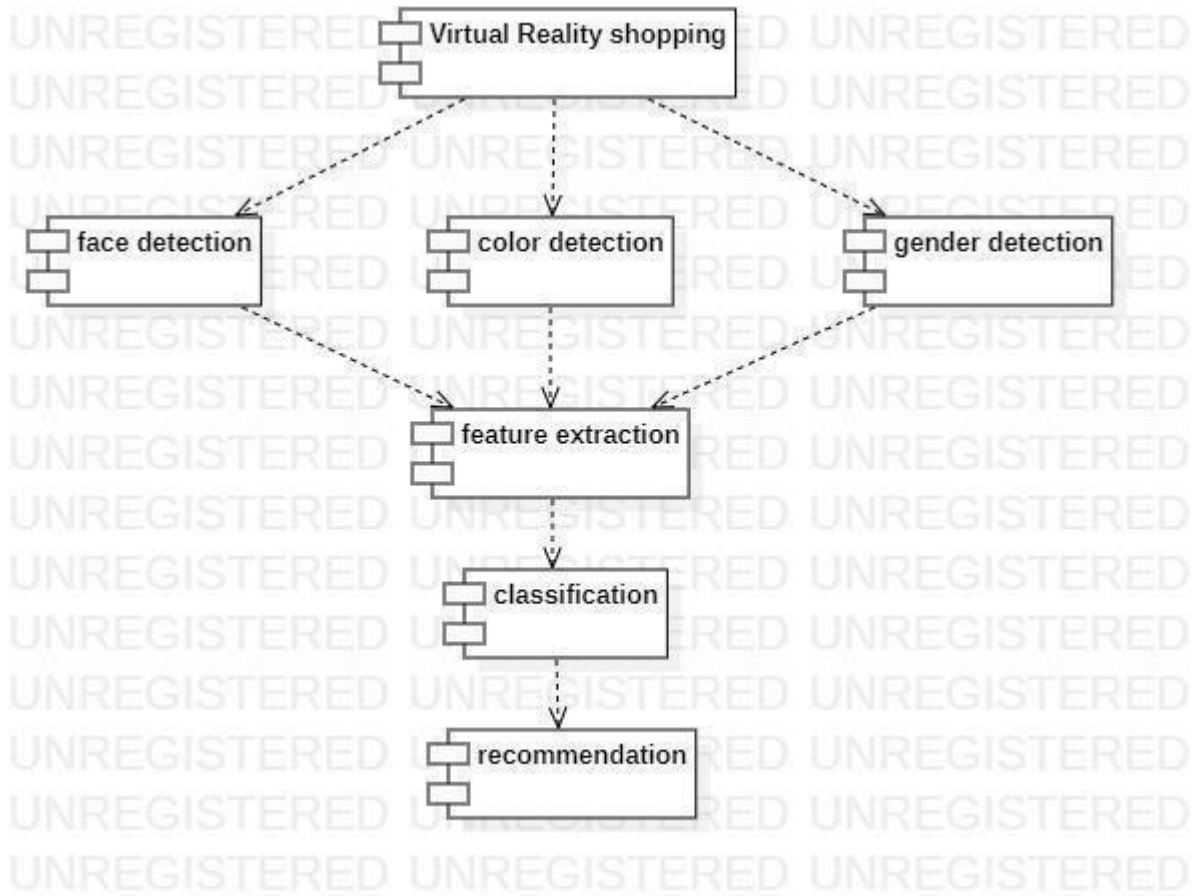


Fig 3.5 Component diagram

CHAPTER 4

SYSTEM ARCHITECTURE

4.1 Architectural Diagram

The system architecture for animal detection typically involves a combination of hardware and software components designed to identify and classify recommendatio based on input data, such as images or sensor readings. Below is a conceptual overview of the system architecture for virtual blind people detection:

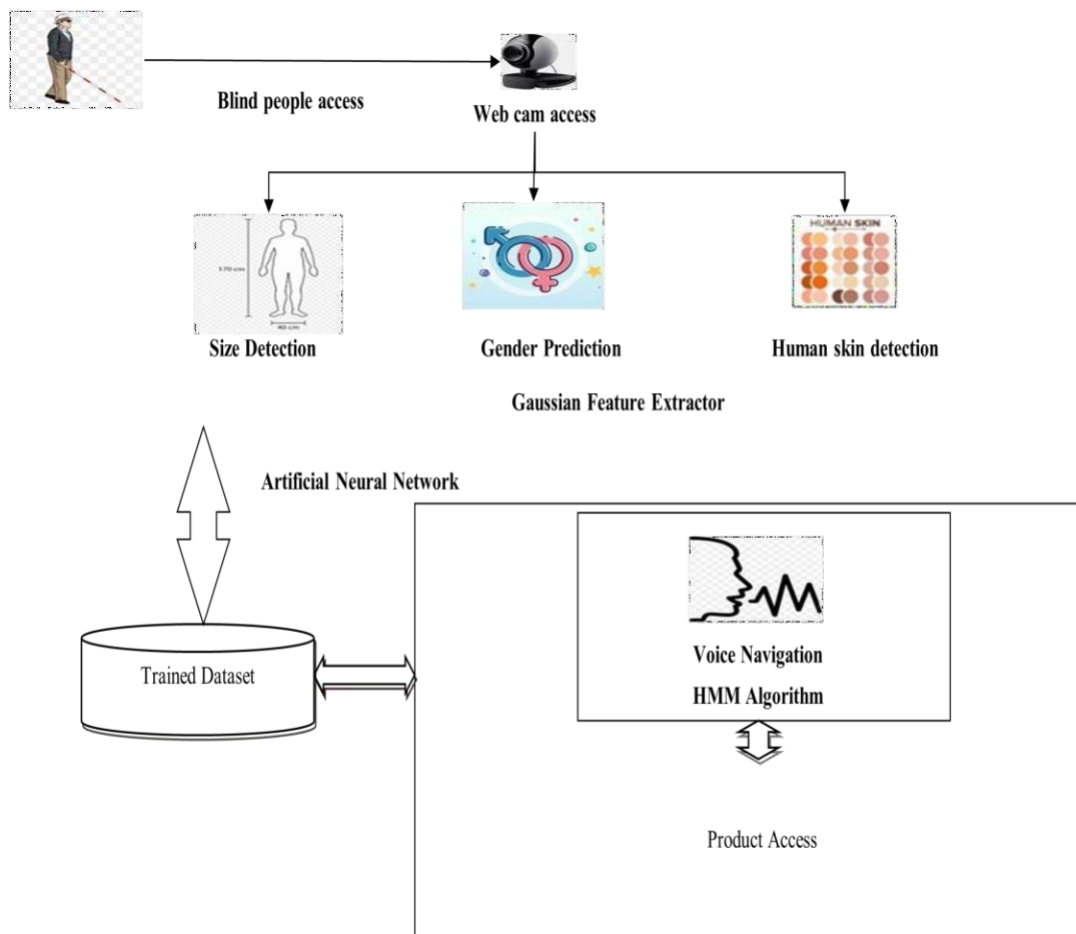


Fig 4.1 System Architecture

4.2 Architectural Description

Normally blind people will make access with camera where three type of features can be detected using Gaussian Feature extractor. The recommendation can be done with voice assistance using HMM and also the classification can be identified with ANN.

The system is targeted to enhance the quality of the videos from the thermal imagers system and equip the imagers with the capability of video acquisition, video streaming and video analytics. Video Capturing Unit and Video Processing Unit will be implemented in the PL portion of the FPGA.

Pre- processing is the process of which the best frame on maintaining median filtering technique is used. The median filtering can be done with the extraction of unwanted noise, blur patterns will be eliminated.

Frame conversion is the process in which the video can be classified as images to get good feature. Mostly analyzing the feature from video is a complex thing thus the image based analysis reduce the complexity of the detection. Thus 10fps(Frames Per Second) has been done thus this implementation can be carried for easy analysis.

The Haar cascade feature set extracts using Gaussian Feature Extractor Algorithm. The algorithm detects the face color values, shoulder size with face based gender system. Thus this algorithm exactly detects the feature value from the human

The classification is done using ANN classification algorithm where the complete analysis of the feature with exact range of recommendation can be done. sometimes also referred to as 'identified skeletal collections'. The observed biological variation in these known populations is described statistically and an identification technique is produced and then applied to the ancient skeletal remains

4.3 SYSTEM REQUIREMENT

HARDWARE REQUIREMENTS

- Processor : Dualcore processor 2.6.0 GHZ
- RAM : 1GB
- Hard disk : 160 GB
- Compact Disk : 650 Mb
- Keyboard : Standard keyboard
- Monitor : 15inch color monitor

SOFTWARE REQUIREMENTS

- Front End PYTHON
- CSV DATASET

CHAPTER 5

SYSTEM IMPLEMENTATION

5.1 MODULES

- Video acquisition
- Frame Conversion
- Pre-processing
- Feature Extraction
- Physic identification
- Outfit recommendation

5.1 MODULES DESCRIPTION

5.1.1 Video Acquisition

In video acquisition a normal video based accessing system where a web cam with video sequence will be played. The system is targeted to enhance the quality of the videos from the thermal imagers system and equip the imagers with the capability of video acquisition, video streaming and video analytics. Video Capturing Unit and Video Processing Unit will be implemented in the PL portion of the FPGA. These units will be responsible for receiving PAL video data @25fps and applying some video processing like 2D-FFT/IFFT and Bilateral/Gaussian filtering.

The raw data rate of the frames generated for PAL video @25fps is 20 M-Byte/sec, Ethernet/USB cannot support such high speed. Hence, it is necessary to compress the video data before streaming it over Ethernet. We'll use JPEG compression. The JPEG compressed video

can be either saved on board in a SD Card present on the board. The saved video will be in the form of sequence of JPAG images. The video can later be retrieved in a host PC using an USB interface. The JPEG compressed video can be streamed over Ethernet interface to a PC. A socket connection will be created between host PC and ZYNQ board for streaming.



Fig 5.1 Video acquisition

5.1.2 Frame Conversion

Frame conversion is the process in which the video can be classified as images to get good feature. Mostly analyzing the feature from video is a complex thing thus the image based analysis reduce the complexity of the detection. Thus 10fps(Frames Per Second) has been done thus this implementation can be carried for easy analysis

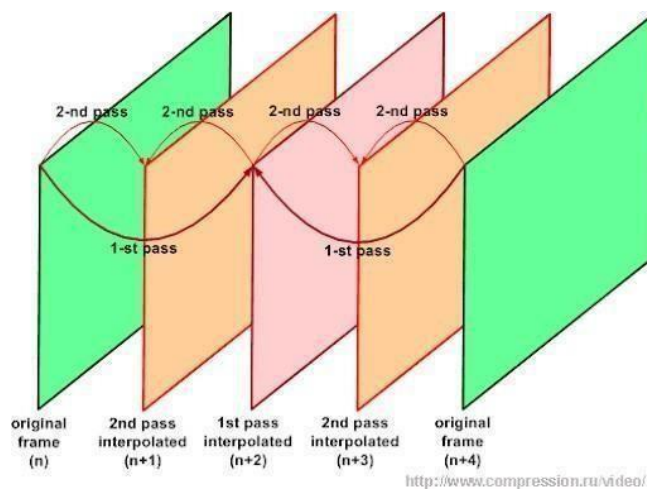


Fig 5.2 Frame conversion process

The Frame Conversion block passes the input through to the output and sets the output sampling mode to the value of the Sampling mode of output signal parameter, which can be either Frame-based or Sample-based. The output sampling mode can also be inherited from the signal at the Ref (reference) input port, which you make visible by selecting the Inherit output sampling mode from <Ref> input port check box.

The Frame Conversion block does not make any changes to the input signal other than the sampling mode. In particular, the block does not rebuffer or resize 2-D inputs. Because 1-D vectors cannot be frame based, when the input is a length-M 1-D vector and the block is in Frame-based mode, the output is a frame-based M-by-1 matrix that is, a single channel.

5.1.3 Pre-Processing

Pre- processing is the process of which the best frame on maintaining median filtering technique is used. The median filtering can be done with the extraction of unwanted noise, blur patterns will be eliminated.

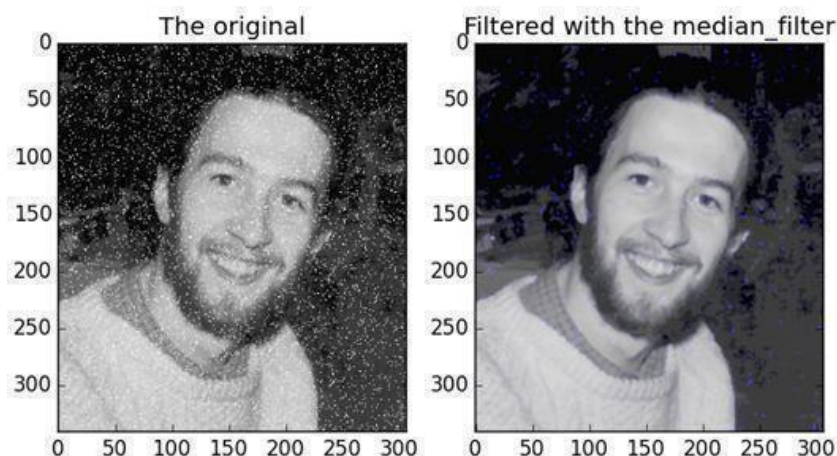


Fig 5.3 Pre-processing system

Image preprocessing is quite useful to improve the quality of images and thus boost them for analysis and further processing. Some powerful image preprocessing techniques

include noise reduction, contrast enhancement, image resizing, color correction, segmentation, feature extraction, etc. It is an essential step in image analysis that helps enhance the data in images and reduce clutter.

5.1.4 Feature Extraction

Feature Extraction is the module where the feature value from the frames can be extracted. In this module three kinds of features can be extracted with Face color, size and shape, gender. The Haar cascade feature set extracts using Gaussian Feature Extractor Algorithm. The algorithm detects the face color values, shoulder size with face based gender system. Thus this algorithm exactly detects the feature value from the human.

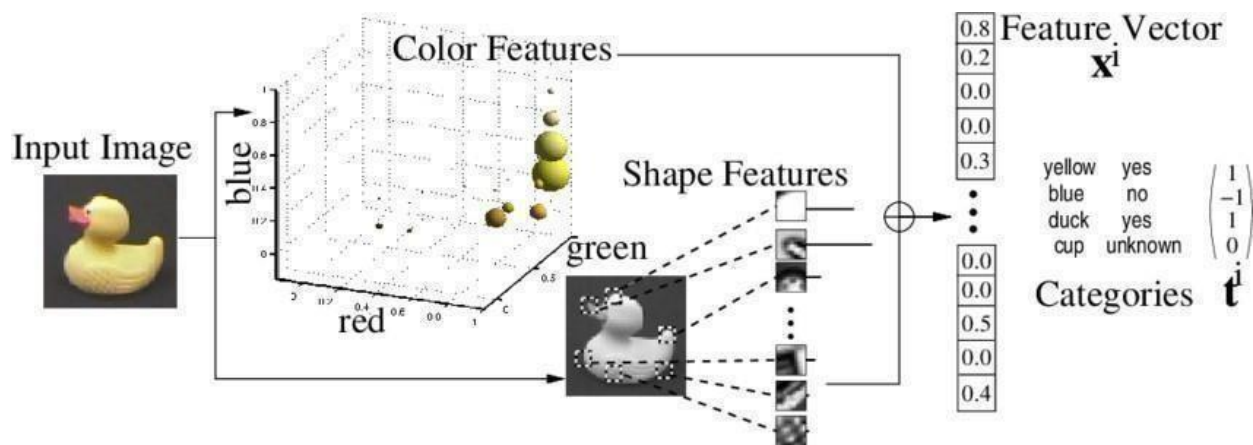


Fig 5.4 Feature Extraction

HAAR cascade: Haar cascade is an algorithm that can detect objects in images, irrespective of their scale in image and location. This algorithm is not so complex and can run in real-time. We can train a haar-cascade detector to detect various objects like cars, bikes, buildings, fruits, etc.

Haar cascade works as a classifier. It classifies positive data points → that are part of our detected object and negative data points → that don't contain our object.

- Haar cascades are fast and can work well in real-time.
- Haar cascade is not as accurate as modern object detection techniques are.

- Haar cascade has a downside. It predicts many false positives.
- Simple to implement, less computing power required.

Pre-trained Haar Cascades

The OpenCV library manages a repository containing all popular haar cascades that can be used for:

- Human face detection
- Eye detection
- Nose / Mouth detection
- Vehicle detection

Gaussian Feature Extractor:

In Gaussian Feature Extractor will helps the image based feature value extractor. Specifically, first, the HSI is decomposed into several Gaussian pyramids to extract multi-scale features, and then the matrix of spectral angle distance (mSAD) is used to generate weight coefficients to evaluate each feature. The extracted feature points will identify the color, face gender feature and size can be extracted.

Feature Extraction aims to reduce the number of features in a dataset by creating new features from the existing ones (and then discarding the original features). These new reduced set of features should then be able to summarize most of the information contained in the original set of features. In this way, a summarized version of the original features can be created from a combination of the original set.

5.1.5 Physic Identification

Thus the extracted feature classifies the size with S, XL, L etc. Color classified as Brown, Black, and Cream White etc. And the identification of the gender can be classified

with the extractor. The classification is done using ANN classification algorithm where the complete analysis of the feature with exact range of recommendation can be done. sometimes also referred to as ‘identified skeletal collections’. The observed biological variation in these known populations is described statistically and an identification technique is produced and then applied to the ancient skeletal remains. This sounds straightforward, but, as many archaeologists have observed previously, it involves a whole host of assumptions concerning the universality of these physical features. Bioarchaeologists are well aware that the skeleton is very far from universal; indeed it is the skeleton’s very plasticity that is harnessed to infer information about past living environments. Yet, in terms of the basic categories of identification, such concerns regarding the uniformitarian assumptions inherent in many techniques are generally brushed aside

5.1.6 Outfit Recommendation

The outfit recommendation can be done with merging ANN classification algorithm in which the outranges can be done with gender, color and size progressions. The recommendation will be like a voice assistant system using HMM algorithm which helps as a assistant for shopping.

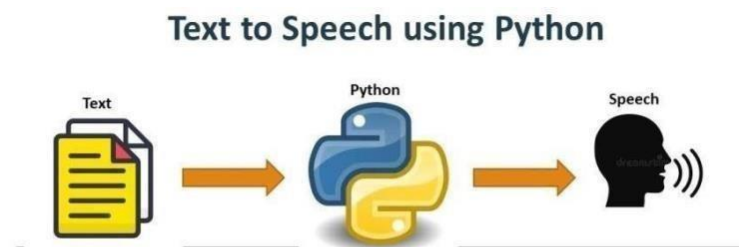


Fig 5.5 HMM conversion system

CHAPTER 6

CODING AND SCREENSHOT

6.1 SAMPLE CODING

Home.py

```
fromtkinter import *
import numpy as np
fromPIL import Image, ImageTk
import cv2
import mediapipe as mp

mp_pose = mp.solutions.pose
pose = mp_pose.Pose()
class tk_master:
    name='ar_master'
    gender=""
    size=""
    color=""
    def _init_(self):
        self.master = 'ar_master'
        self.title = 'Vr Blind Shopping'
        self.titlec = 'VR BLIND SHOPPING'
        self.background_color = '#2F4F4F'
        self.text_color = '#FFF'
        self.background_image = 'images/background_hd1.jpg'
    def get_title(self):
        return self.title
    def get_titlec(self):
```

```

        return self.titlec
def get_background_color(self):
    return self.background_color
def get_text_color(self):
    return self.text_color
def get_background_image(self):
    return self.background_image
def set_window_design(self):
    root = Tk()
    w = 780
    h = 500
    ws = root.winfo_screenwidth()
    hs = root.winfo_screenheight()
    x = (ws / 2) - (w / 2)
    y = (hs / 2) - (h / 2)
    root.geometry('%dx%d+%d+%d' % (w, h, x, y))
    self.bg = ImageTk.PhotoImage(file='images/background_hd1.jpg')
    root.title(self.title)
    root.resizable(False, False)
    bg = ImageTk.PhotoImage(file=self.background_image)
    canvas = Canvas(root, width=200, height=300)
    canvas.pack(fill="both", expand=True)
    canvas.create_image(0, 0, image=bg, anchor=NW)
    canvas.create_text(390, 20, text=self.title, font=("Times New Roman", 24),
def clickHandler(event):
    tt = tk_master
    tt.login_home(event)
image = Image.open('images/logo.png')
img = image.resize((225, 225))
my_img = ImageTk.PhotoImage(img)

```

```

image_id = canvas.create_image(380, 200, image=my_img)
canvas.tag_bind(image_id, "<1>", clickHandler)
admin_id = canvas.create_text(380, 360, text="Start", font=("Times New Roman", 28),
canvas.tag_bind(admin_id, "<1>", clickHandler)
root.mainloop()
def login_home(self):
    def categorize_skin_color(image, face_region):
        x, y, w, h = face_region
        roi = image[y:y + h, x:x + w]
        ycrb = cv2.cvtColor(roi, cv2.COLOR_BGR2YCrCb)
        y_mean = np.mean(ycrb[:, :, 0])
        cr_mean = np.mean(ycrb[:, :, 1])
        cb_mean = np.mean(ycrb[:, :, 2])
        # print(y_mean, cr_mean, cb_mean)
        if y_mean > 180 and cr_mean > 130 and cr_mean < 170 and cb_mean > 77 and cb_mean
        return "Medium"
    def highlightFace(net, frame, conf_threshold=0.7):
        frameOpencvDnn = frame.copy()
        frameHeight = frameOpencvDnn.shape[0]
        frameWidth = frameOpencvDnn.shape[1]

```

```

root = Tk()
w = 780
h = 500
ws = root.winfo_screenwidth()
hs = root.winfo_screenheight()
x = (ws / 2) - (w / 2)
y = (hs / 2) - (h / 2)
root.geometry('%dx%d+%d+%d' % (w, h, x, y))
self.bg = ImageTk.PhotoImage(file='images/background_hd1.jpg')
root.title(self.title)
root.resizable(False, False)
bg = ImageTk.PhotoImage(file=self.background_image)
canvas = Canvas(root, width=200, height=300)
canvas.pack(fill="both", expand=True)
canvas.create_image(0, 0, image=bg, anchor=NW)
canvas.create_text(390, 20, text=self.title, font=("Times New Roman", 24), fill=self.text_color)
def clickHandler(event):
    tt = tk_master
    tt.login_home(event)
image = Image.open('images/logo.png')
img = image.resize((225, 225))
my_img = ImageTk.PhotoImage(img)
image_id = canvas.create_image(380, 200, image=my_img)
canvas.tag_bind(image_id, "<1>", clickHandler)
admin_id = canvas.create_text(380, 360, text="Start", font=("Times New Roman", 28),
canvas.tag_bind(admin_id, "<1>", clickHandler)
root.mainloop()
def categorize_skin_color(image, face_region):
    x, y, w, h = face_region
    roi = image[y:y + h, x:x + w]
    ycrb = cv2.cvtColor(roi, cv2.COLOR_BGR2YCrCb)
    y_mean = np.mean(ycrb[:, :, 0])

```

```

cr_mean = np.mean(ycrcb[:, :, 1])
cb_mean = np.mean(ycrcb[:, :, 2])
# print(y_mean,cr_mean,cb_mean)
if y_mean > 180 and cr_mean > 130 and cr_mean < 170 and cb_mean > 77 and cb_mean < 127:
    return "White"
elif y_mean > 120 and cr_mean > 100 and cr_mean < 150 and cb_mean > 90 and cb_mean <
    return "Brown"
elif y_mean < 120 and cr_mean > 130 and cr_mean < 170 and cb_mean > 77 and cb_mean <
    return "Black"
else:
    return "Medium"
def highlightFace(net, frame, conf_threshold=0.7):
    frameOpencvDnn = frame.copy()
    frameHeight = frameOpencvDnn.shape[0]
    frameWidth = frameOpencvDnn.shape[1]
    blob = cv2.dnn.blobFromImage(frameOpencvDnn, 1.0, (300, 300), [104, 117, 123], True, False)
    net.setInput(blob)
    detections = net.forward()
    faceBoxes = []
    for i in range(detections.shape[2]):
        confidence = detections[0, 0, i, 2]
        if confidence > conf_threshold:
            x1 = int(detections[0, 0, i, 3] * frameWidth)
            y1 = int(detections[0, 0, i, 4] * frameHeight)
            x2 = int(detections[0, 0, i, 5] * frameWidth)
            y2 = int(detections[0, 0, i, 6] * frameHeight)
            faceBoxes.append([x1, y1, x2, y2])
    return frameOpencvDnn, faceBoxes
total_frame_count=30
count=0
faceProto = "ar_model_gender/opencv_face_detector.pbtxt"
faceModel = "ar_model_gender/opencv_face_detector_uint8.pb"

```

```

genderProto = "ar_model_gender/gender_deploy.prototxt"
genderModel = "ar_model_gender/gender_net.caffemodel"
MODEL_MEAN_VALUES = (78.4263377603, 87.7689143744, 114.895847746)
genderList = ['Male', 'Female']
faceNet = cv2.dnn.readNet(faceModel, faceProto)
genderNet = cv2.dnn.readNet(genderModel, genderProto)
video = cv2.VideoCapture('http://192.168.1.3:8080/video')
# video = cv2.VideoCapture(0)
padding = 20
color = (0, 0, 255)
thickness = 1
size_dict = {"s": 0, "m": 0, "l": 0, "xl": 0, "xxl": 0, "xxxl": 0}
gender_dict = {"Male": 0, "Female": 0}
color_dict = {"White": 0, "Black": 0, "Brown": 0, "Medium": 0, "Other": 0}
upper_body_cascade =
cascPath = "data/haarcascades/haarcascade_frontalface_default.xml"
faceCascade = cv2.CascadeClassifier(cascPath)
while 1:
    ret, frame = video.read()
    height, width = frame.shape[:2]
    gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
    resultImg, faceBoxes = highlightFace(faceNet, frame)
    frame_center_w = width / 2
    frame_left2_w=frame_center_w/2
    frame_left1_w=frame_left2_w/2
    left_edge_w=frame_left1_w+frame_left2_w
    start_point_h = (int(left_edge_w), 10)
    end_point_h = (int(left_edge_w), height-10)
    cv2.line(resultImg, start_point_h, end_point_h, (0,0,0), thickness)

    if check_distance==0:

```

```

check_distance=0
upper_bodies = upper_body_cascade.detectMultiScale(gray, scaleFactor=1.1, =5,
for (x, y, w, h) in upper_bodies:
    cv2.rectangle(resultImg, (x, y), (x + w, y+ h), (0, 255, 255), 2)
    body_type='wait'
    if left_edge_w+10<x:
        body_type='Move Right'
    elif left_edge_w-10>x:
        body_type = 'Move Left'
    else:
        body_type='Ok'
        find_sholder=1
        if find_sholder==1:
            find_sholder=0
            sholder = tmp_sholder
            sholder_result = "unknown"
            if (sholder < 42):
                sholder_result = 's'
            elif (sholder <= 50):
                sholder_result = 'm'
            elif (sholder <= 54):
                sholder_result = 'l'
            elif (sholder <= 58):
                sholder_result = 'xl'
            elif (sholder <= 70):
                sholder_result = 'xxl'
            else:
                sholder_result = 'xxxl'
            xx = size_dict[sholder_result]
            xx = xx + 1
            size_dict[sholder_result] = xx
            count = count + 1

```

```

        if count > 40:
            Keymax = max(zip(size_dict.values(), size_dict.keys()))[1]
            tk_master.size=str(Keymax)
    else:
    for faceBox in faceBoxes:
        face = frame[max(0, faceBox[1] - padding):

        genderNet.setInput(blob)
        genderPreds = genderNet.forward()
        gender = genderList[genderPreds[0].argmax()]
        xx = gender_dict[gender]
        xx = xx + 1
        gender_dict[gender] = xx
        count = count + 1
    if count > total_frame_count:
        Keymax = max(zip(gender_dict.values(), gender_dict.keys()))[1]
        tk_master.gender=str(Keymax)
    cv2.putText(resultImg, f'{gender}', (faceBox[0], faceBox[1] - 10),

    faces = faceCascade.detectMultiScale(frame, 1.1, 3, minSize=(100, 100))
    for (x, y, w, h) in faces:
        category = categorize_skin_color(frame, (x, y, w, h))
        xx = color_dict[category]
        xx = xx + 1
        color_dict[category] = xx
    cv2.imshow("Image1", resultImg)
    if tk_master.gender!=" and tk_master.size!=":
        Keymax2 = max(zip(color_dict.values(), color_dict.keys()))[1]
        tk_master.color = str(Keymax2)
        break
    k = cv2.waitKey(30) & 0xff
    if k == 27:

```

```

        break
    file1 = open("user_data.txt", "w")
    L = [tk_master.gender + "#" + tk_master.size + "#" + tk_master.color]
    file1.writelines(L)
    file1.close()
    video.release()
    cv2.destroyAllWindows()
    if tk_master.size!="":
        import dress_shopping
        ar = dress_shopping.shopping()
        ar.show_window()

ar = tk_master()
root = ar.set_window_design()

```

Dress shopping

```

import csv
import time
from tkinter import *
from tkinter.ttk import Treeview
import pyttsx3
import speech_recognition as sr
import numpy as np
from PIL import Image, ImageTk
import cv2
import mediapipe as mp
class shopping:
    gender = 'Female'
    size = 'xl'
    color = 'white'
    category=""
    name=""

```

```

type="
rcolor="
pattern="
navigation="
description="
price="
offer="

def __init__(self):
    f = open("user_data.txt", "r")
    user_data=(f.read())
    xx = user_data.split('#')
    shopping.gender=xx[0]
    shopping.size=xx[1]
    shopping.color=xx[2]

    self.master = 'ar_master'
    self.title='Blind Shopping'
    self.background_color = '#2F4F4F'
    self.text_color = '#FFF'
    self.background_image = 'images/background_hd1.jpg'
def show_window(self):
    def get_voice_input():
        try:
            r = sr.Recognizer()
            with sr.Microphone() as source:
                audio_data = r.record(source, duration=5)
                text = r.recognize_google(audio_data)
                # print(text)
            return text
        except Exception as error:
            print(error)

```

```

    SpeakText('Incorrect Data')
    print("***")
    get_voice_input()
def get_csv_dress_category():
    data=[]
    shopping.gender=shopping.gender.lower().strip()
    shopping.size=shopping.size.lower().strip()
    shopping.color=shopping.color.lower().strip()
    file = 'vr_dataset.csv'
    with open(file) as f:
        reader = csv.DictReader(f, delimiter=',')
        for row in reader:
            t1 = row['Gender'].lower().strip()
            t2 = row['Size'].lower().strip()
            t3 = row['Color'].lower().strip()
            t4 = row['Dress Category'].lower()
            if((t1==shopping.gender) and (t2==shopping.size) and (t3==shopping.color)):
                data.append(t4.lower())
    data = list(set(data))
    return data
def get_csv_dress_name():
    data=[]
    shopping.gender=shopping.gender.lower().strip()
    shopping.size=shopping.size.lower().strip()
    shopping.color=shopping.color.lower().strip()
    shopping.category=shopping.category.lower().strip()
    file = 'vr_dataset.csv'
    with open(file) as f:
        reader = csv.DictReader(f, delimiter=',')
        for row in reader:
            t1 = row['Gender'].lower().strip()
            t2 = row['Size'].lower().strip()

```

```

        t3 = row['Color'].lower().strip()
        t4 = row['Dress Category'].lower()
        t5 = row['Dress name'].lower()
        if((t1==shopping.gender) and (t2==shopping.size) and (t3==shopping.color) and
            data.append(t5.lower())
data = list(set(data))
return data
def get_csv_dress_type():
data=[]
shopping.gender=shopping.gender.lower().strip()
shopping.size=shopping.size.lower().strip()
shopping.color=shopping.color.lower().strip()
shopping.category=shopping.category.lower().strip()
shopping.name=shopping.name.lower().strip()
file = 'vr_dataset.csv'
with open(file) as f:
    reader = csv.DictReader(f, delimiter=',')
    for row in reader:
        t1 = row['Gender'].lower().strip()
        t2 = row['Size'].lower().strip()
        t3 = row['Color'].lower().strip()
        t4 = row['Dress Category'].lower().strip()
        t5 = row['Dress name'].lower().strip()
        t6 = row['Dress type'].lower().strip()
        if((t1==shopping.gender) and (t2==shopping.size) and (t3==shopping.color) and
            data = list(set(data))
return data
def get_csv_dress_recommendation():
data=[]
shopping.gender=shopping.gender.lower().strip()
shopping.size=shopping.size.lower().strip()
shopping.color=shopping.color.lower().strip()

```

```

shopping.category=shopping.category.lower().strip()
shopping.name=shopping.name.lower().strip()
shopping.type=shopping.type.lower().strip()
file = 'vr_dataset.csv'
with open(file) as f:
    reader = csv.DictReader(f, delimiter=',')
    for row in reader:
        t1 = row['Gender'].lower().strip()
        t2 = row['Size'].lower().strip()
        t3 = row['Color'].lower().strip()
        t4 = row['Dress Category'].lower().strip()
        t5 = row['Dress name'].lower().strip()
        t6 = row['Dress type'].lower().strip()
        t7 = row['color'].lower().strip()
        t8 = row['Pattern'].lower().strip()
        t9 = row['Navigation'].lower().strip()
        t10 = row['Description'].lower().strip()
        t11 = row['Price'].lower().strip()
        t12 = row['Offers'].lower().strip()
        if((t1==shopping.gender) and (t2==shopping.size) and (t3==shopping.color) and
(t4==shopping.category) and (t5==shopping.name) and (t6==shopping.type)):
            shopping.rcolor=t7.lower()
            shopping.pattern=t8.lower()
            shopping.navigation=t9.lower()
            shopping.description=t10.lower()
            shopping.price=t11.lower()
            shopping.offer=t12.lower()

data = list(set(data))
return data

def SpeakText(command):

```

```

engine = pyttsx3.init()
engine.say(command)
engine.runAndWait()
engine.stop()

def select_dress_category():
    data = get_csv_dress_category()
    SpeakText("listen Your Dress Category List.")
    for x in data:
        SpeakText(x)
        time.sleep(1)
    SpeakText("Please, Enter Your, Choice.")
    dress_category=get_voice_input()
    return dress_category

def select_dress_name():
    data = get_csv_dress_name()
    SpeakText("listen Your Dress Name List")
    for x in data:
        SpeakText(x)
        time.sleep(1)
    SpeakText("Please, Enter Your, Choice.")
    dress_category=get_voice_input()
    return dress_category

```

```

def select_dress_type():
    data = get_csv_dress_type()
    SpeakText("listen Your Dress Type List")
    for x in data:
        SpeakText(x)
        time.sleep(1)
    SpeakText("Please, Enter Your, Choice.")
    dress_type=get_voice_input()
    return dress_type

    t1 = row['Gender'].lower().strip()
    t2 = row['Size'].lower().strip()
    t3 = row['Color'].lower().strip()
    t4 = row['Dress Category'].lower().strip()
    t5 = row['Dress name'].lower().strip()
    t6 = row['Dress type'].lower().strip()
    if((t1==shopping.gender) and (t2==shopping.size) and (t3==shopping.color) and
        data.append(t6.lower()))
    data = list(set(data))
    return data

def get_csv_dress_recommendation():
    data=[]
    shopping.gender=shopping.gender.lower().strip()
    shopping.size=shopping.size.lower().strip()
    shopping.color=shopping.color.lower().strip()
    shopping.category=shopping.category.lower().strip()
    shopping.name=shopping.name.lower().strip()
    shopping.type=shopping.type.lower().strip()
    file = 'vr_dataset.csv'
    with open(file) as f:
        reader = csv.DictReader(f, delimiter=',')

```

```

for row in reader:
    t1 = row['Gender'].lower().strip()
    t2 = row['Size'].lower().strip()
    t3 = row['Color'].lower().strip()
    t4 = row['Dress Category'].lower().strip()
    t5 = row['Dress name'].lower().strip()
    t6 = row['Dress type'].lower().strip()
    t7 = row['color'].lower().strip()
    t8 = row['Pattern'].lower().strip()
    t9 = row['Navigation'].lower().strip()
    t10 = row['Description'].lower().strip()
    t11 = row['Price'].lower().strip()
    t12 = row['Offers'].lower().strip()
    if((t1==shopping.gender) and (t2==shopping.size) and (t3==shopping.color) and
        shopping.rcolor=t7.lower()
        shopping.pattern=t8.lower()
        shopping.navigation=t9.lower()
        shopping.description=t10.lower()
        shopping.price=t11.lower()
        shopping.offer=t12.lower()

data = list(set(data))
return data

def SpeakText(command):
    engine = pyttsx3.init()
    engine.say(command)
    engine.runAndWait()
    engine.stop

```

```

def select_dress_category():
    data = get_csv_dress_category()
    SpeakText("listen Your Dress Category List.")
    for x in data:
        SpeakText(x)
        time.sleep(1)
    SpeakText("Please, Enter Your, Choice.")
    dress_category=get_voice_input()
    return dress_category

```

```

def select_dress_name():
    data = get_csv_dress_name()
    SpeakText("listen Your Dress Name List")
    for x in data:
        SpeakText(x)
        time.sleep(1)
    SpeakText("Please, Enter Your, Choice.")
    dress_category=get_voice_input()
    return dress_category

```

```

def select_dress_type():
    data = get_csv_dress_type()
    SpeakText("listen Your Dress Type List")
    for x in data:
        SpeakText(x)
        time.sleep(1)
    SpeakText("Please, Enter Your, Choice.")
    dress_type=get_voice_input()
    return dress_type

```

```

ar=shopping()
ar.show_window()
canvas

import cv2
import mediapipe as mp

# Initialize MediaPipe Pose
mp_pose = mp.solutions.pose
pose = mp_pose.Pose()

# Function to identify if a person is thin, medium, or fat
def identify_body_type(image_path):
    # Read the image
    image = cv2.imread(image_path)
    image_rgb = cv2.cvtColor(image, cv2.COLOR_BGR2RGB)

    # Process the image and get the pose estimation results
    results = pose.process(image_rgb)

    if results.pose_landmarks:
        landmarks = results.pose_landmarks.landmark

```

```

# Get the y-coordinate of the shoulder and hip landmarks
shoulder_y = int(landmarks[mp_pose.PoseLandmark.LEFT_SHOULDER].y *
hip_y = int(landmarks[mp_pose.PoseLandmark.LEFT_HIP].y * image.shape[0])

# Get the x-coordinate of the left and right hips
left_hip_x = int(landmarks[mp_pose.PoseLandmark.LEFT_HIP].x * image.shape[1])
right_hip_x = int(landmarks[mp_pose.PoseLandmark.RIGHT_HIP].x * image.shape[1])

# Calculate torso height and hip width
torso_height = hip_y - shoulder_y
hip_width = right_hip_x - left_hip_x

# Determine body type based on the ratio of torso height to hip width
ratio = torso_height / hip_width
print(ratio)

if ratio < 1.0: # Adjust this threshold based on your observations
    body_type = "Thin"
elif ratio > 1.2: # Adjust this threshold based on your observations
    body_type = "Fat"
else:
    body_type = "Medium"

# Draw landmarks and body type on the image
# mp.drawing_utils.draw_landmarks(image, results.pose_landmarks,
mp_pose.POSE_CONNECTIONS)
cv2.putText(image, f'Body Type: {body_type}', (10, 30),
cv2.FONT_HERSHEY_SIMPLEX, 1, (0, 255, 0), 2)

# Display the image with body type information

```

```
cv2.imshow('Body Type Identification', image)
cv2.waitKey(0)
cv2.destroyAllWindows()
else:
    print("No pose landmarks detected.")

# Example usage
image_path = 'test_data/v2.jpeg' # Provide the path to the image you want to test
identify_body_type(image_path)
```

6.2 SAMPLE SCREENSHOT



Fig 6.2.1 Home page

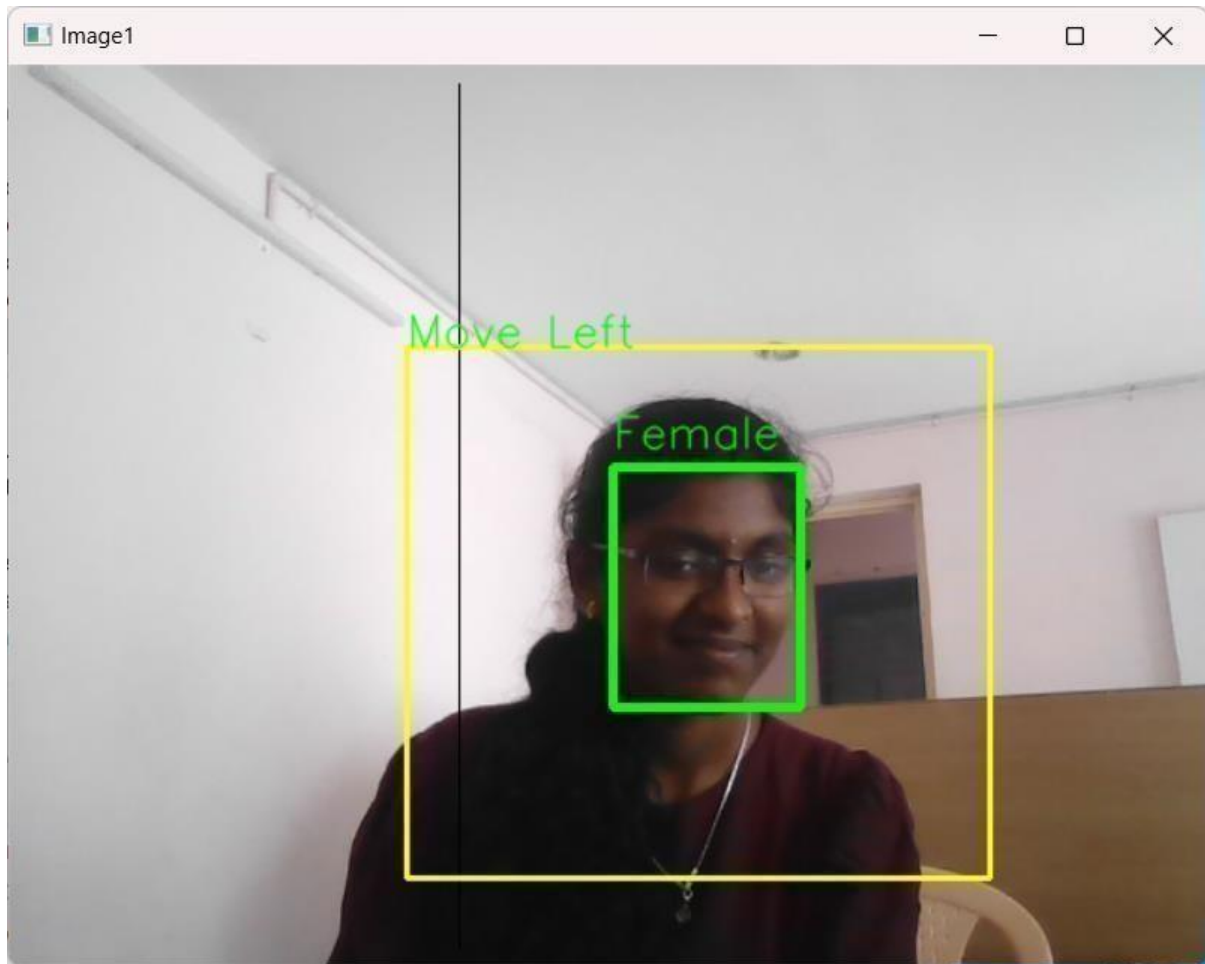


Fig 6.2.2 Face detection

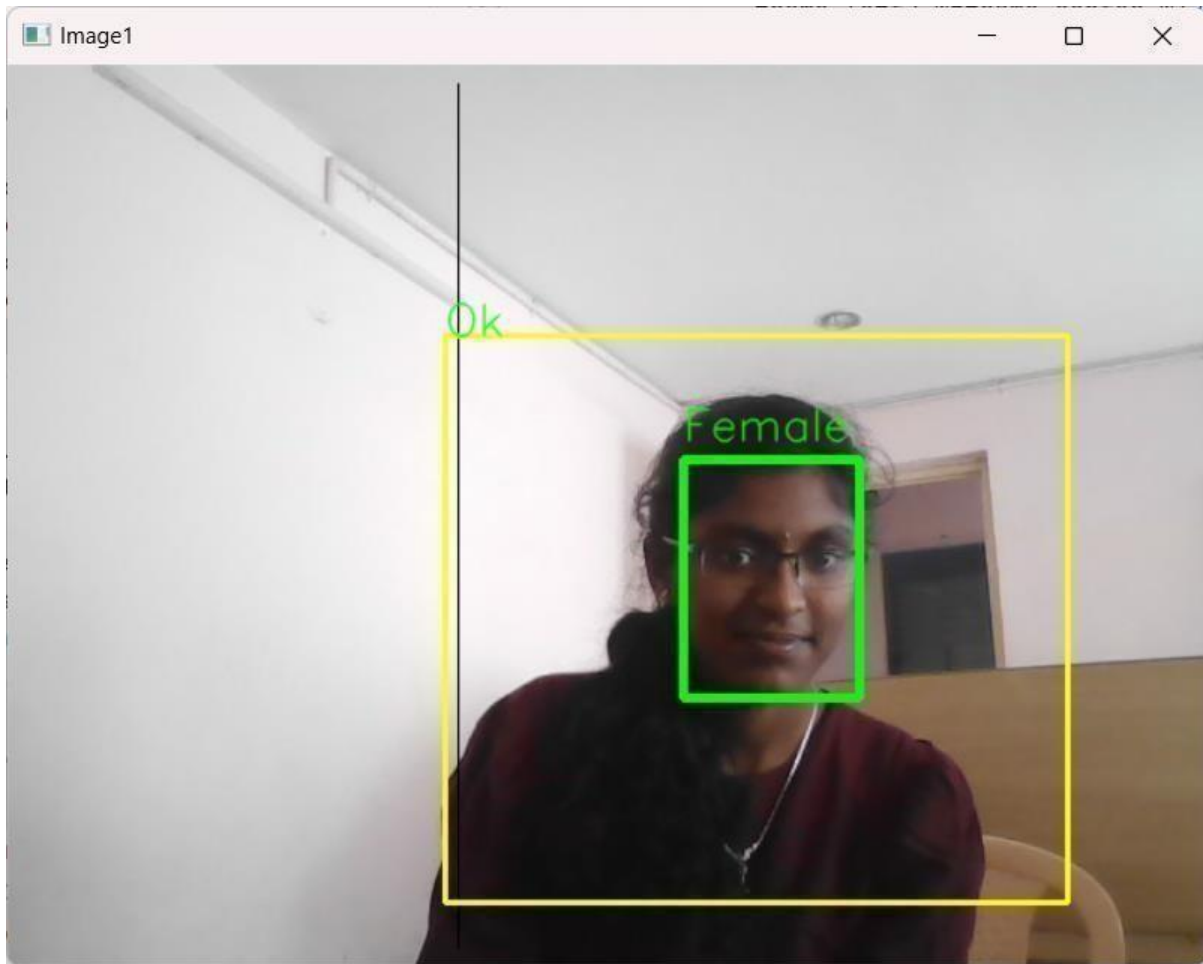


Fig 6.2.3 Feature Extraction



Fig 6.2.24 Recommendation

CHAPTER 7

CONCLUSION AND FUTURE WORK

7.1 CONCLUSION

The developed software program leads a visually challenged person through a shopping area and onto a store shelf where they found a special item. The system demonstrated how we integrated product localization software with creative and forward-thinking ideas to produce an interior and object navigational aid. A substantial amount of integration effort was required to bring all of the vision-founded algorithms together and generate a consistent estimate of the individual's function in order to develop this method. Algorithms were also chosen and changed to satisfy the demands for real-time execution, which is crucial for assistive technologies that operate on headphones. The additional effort in this project is to design an object detection system employing voice recognition technology and a deep learning object recognition technique. For visually handicapped people, this system's voice synthesis offers convenient functions. Our study was carried out with an emphasis on the most efficient ways to assist the blind, as this is one of the fields in which deep learning technology can be used. Consequently, the system was enhanced with voice recognition and voice direction technologies, and its functionality was evaluated.

7.2 FUTURE ENHANCEMENT

This study will help the blind people live more comfortably and privately in the future. It is also expected to be employed in industrial environments with low visibility, such as seabed regions and coal mines, to greatly assist in production and industrial development in highly. Future systems may employ the implementation in a smart phone system that is easily accessible to blind users through their smartphones. The smart phone that will be used to run this program must be turned on and have a sufficient amount of battery life, among other restrictions on the system. The user is required to always carry it with them. In that scenario, a wearable gadget is more practical as it frees up our hands.

CHAPTER 8

REFERENCE

- [1] Qibin Hou ; Ming-Ming Cheng ; Xiaowei Hu ; Ali Borji ; Zhuowen Tu ; Philip H. S. Torr, Deeply Supervised Salient Object Detection with Short Connections, IEEE Transactions on Pattern Analysis and Machine Intelligence (Volume: 41 , Issue: 4 , April 1 2019)
- [2] Shuze Du ; Shifeng Chen, Salient Object Detection via Random Forest, IEEE Signal Processing Letters (Volume: 21 , Issue: 1 , Jan. 2014)
- [3] Xiaozhi Chen ; Kaustav Kundu ; Yukun Zhu ; Huimin Ma ; Sanja Fidler ; Raquel Urtasun, 3D Object Proposals Using Stereo Imagery for Accurate Object Class Detection, IEEE Transactions on Pattern Analysis and Machine Intelligence (Volume: 40 , Issue: 5 , May 1 2018)
- [4] Yunhang Shen ; Rongrong Ji ; Changhu Wang ; Xi Li ; Xuelong Li , Weakly Supervised Object Detection via Object-Specific Pixel Gradient, IEEE Transactions on Neural Networks and Learning Systems (Volume: 29 , Issue: 12 , Dec. 2018)
- [5] Yuxing Tang ; Josiah Wang ; Boyang Gao ; Emmanuel Dellandréa ; Robert Gaizauskas ; Liming Chen, Large Scale Semi-Supervised Object Detection Using Visual and Semantic Knowledge Transfer, IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016
- [6] Kulyukin, V., Kutiyawala, A.: Accessible Shopping Systems for Blind and Visually Impaired Individuals: Design Requirements and the State of the Art. The Open Rehabilitation Journal 3, 158–168 (2020)
- [7] Kulyukin, V., Gharpure, C., Nicholson, J.: RoboCart: Toward robot- assisted navigation of grocery stores by the visually impaired. In: Proceedings of the IEEE/RSJ IROS, pp. 2845– 2850. IEEE Press, Edmonton (2019)
- [8] Nicholson, J., Kulyukin, V.: ShopTalk: Independent blind shopping = verbal route directions + barcode scans. In: Proceedings of the 30th Rehabilitation Engineering and Assistive Technology Society of North America (RESNA), Phoenix, USA (2017)

- [9] Merler, M., Galleguillos, C., Belongie, S.: Recognizing groceries in situ using in vitro training data. SLAM, Minneapolis (2019)
- [10] Lanigan, P.E., Paulos, A.M., Williams, A.W., Rossi, D., Narasimhan, P.: Trinetra: Assistive technologies for grocery shopping for the blind. In: International IEEE-BAIS Symposium on Research on Assistive Technologies (RAT). Dayton, OH (2020)