University of Kerbala
College of Computer Science & Information Technology
Computer Science Department

Developing An Internet of Things System for Paralyzed Patients Using Eyebrow Movement and Voice Assessment Technologies

A Thesis

Submitted to the Council of the College of Computer Science & Information Technology / University of Kerbala in Partial Fulfillment of the Requirements for the Master Degree in Computer Science

Written by

Ali Thabit Jabbar

Supervised by

Prof. Dr. Baheeja Kudayir Shukur

Prof. Dr. Nidhal Khdhair El Abbadi

2024 A.D. 1446 A.H.

بِسْمِ ٱللَّهِ ٱلرَّحْمَٰنِ ٱلرَّحِيمِ

((يَرْفَعِ اللَّهُ الَّذِينَ آمَنُوا مِنْكُمْ وَالَّذِينَ أُوتُوا
الْعِلْمَ دَرَجَتٍ وَاللَّهُ بِمَا تَعْمَلُونَ خَبِيرٌ))

صَـدَقَ أللهُ ٱلعَلِـيُّ ٱلعَظْيِـم

المجـادلـة – الجزء الثامن والعشرون ــ الآية (11)

Supervisor Certification

I certify that the thesis entitled (**Developing An Internet of Things System for Paralyzed Patients Using Eyebrow Movement and Voice Assessment Technologies**) was prepared under my supervision at the department of Computer Science / College of Computer Science & Information Technology / University of Kerbala as partial fulfillment of the requirements of the degree of Master in Computer Science.

Signature:
Prof. Dr. Baheeja Kudayir Shukur.
Date: / /2024

Signature:
Prof. Dr. Nidhal Khdhair El Abbadi.
Date: / /2024

The Head of the Department Certification

In view of the available recommendations, I forward the thesis entitled "**Developing An Internet of Things System for Paralyzed Patients Using Eyebrow Movement and Voice Assessment Technologies**" for debate by the examination committee.

Signature:
Assist. Prof. Dr. Muhannad Kamil Abdulhameed
Head of Computer Science Department
Date: / / 2024

Certification of the Examination Committee

We hereby certify that we have studied the dissertation entitled (**Developing An Internet of Things System for Paralyzed Patients Using Eyebrow Movement and Voice Assessment Technologies**) presented by the student (**Ali Thabit Jabbar**) and examined him/her in its content and what is related to it, and that, in our opinion, it is adequate with (**Very good**) standing as a thesis for the Master degree in Computer Science.

Signature:
Name: Ashwan Anwer Abdulmunem
Title: Assist Prof. Dr.
Date: / / 2024
(Chairman)

Signature:
Name: Meeras Salman Juwad
Title: Dr. Lect.
Date: / / 2024
(Member)

Signature:
Name: Nidhal Khdhair El Abbadi
Title: Prof. Dr.
Date: / / 2024
(Member and Supervisor)

Signature:
Name: Ali Abdul Azeez Mohammed
Title: Assist Prof. Dr.
Date: / / 2024
(Member)

Signature:
Name: Baheeja Kudayir Shukur
Title: Prof. Dr.
Date: / / 2024
(Member and Supervisor)

Approved by the Dean of the College of Computer Science & Information Technology, University of Kerbala.

Signature:
Assist. Prof. Dr. Mowafak Khadom Mohsen
Date: / / 2024
(Dean of College of Computer Science & Information Technology)

Dedication

I dedicate my efforts to the martyrs of Iraq, especially my martyred brother, Ameer Thabit. I also thank those who have given me all their support and love, my mother and sister, who stood by me and supported me throughout my academic journey. Thank you for everything.

Acknowledgement

First, I would like to thank God Almighty for His many blessings that enabled me to complete my research successfully.

I extend all thanks to our noble Prophet, the first teacher of humanity, Prophet Muhammad (peace be upon him and his family), and to his pure and virtuous household. I also express my gratitude to my master, Imam Hussein (peace be upon him), and to the Awaited Imam (peace be upon him).

I extend my sincere thanks to Prof. Dr. Nidhal Khdhair El Abbadi and Prof. Dr. Baheeja Kudayir Shukur, my thesis supervisors, for their hard work, scientific guidance, and insightful consultations, and for their patience with my less-than-serious efforts.

I want to extend my gratitude and acknowledgment to the Dean of the College of Computer Science and Information Technology, all the faculty members, the Head of the Computer Science Department, and my fellow graduate students in the College of Computer Science and Information Technology at the University of Kerbala, as well as my colleagues.

Abstract

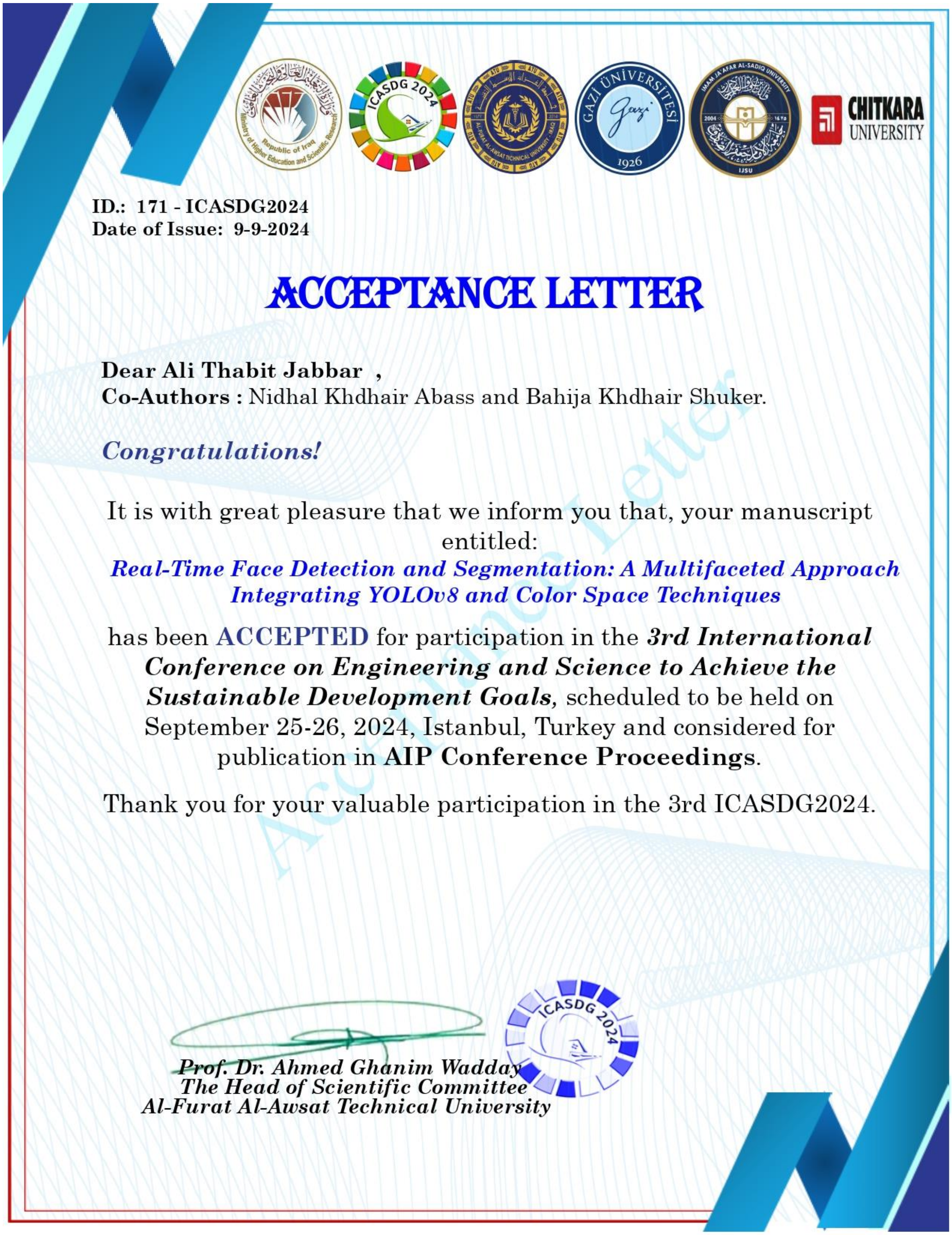
This thesis presents an advanced IoT-based system designed to control smart home devices, specifically developed to meet the needs of people with paralysis or disabilities. It addresses the critical role of IoT in facilitating independent living for individuals with physical limitations by enabling smart home applications that respond to various user inputs. Two types of paralyzed users are considered: those who are unable to speak, using eyebrow movements to control the smart home, and those who can speak, utilizing voice commands converted to text to control devices or send messages.

The system incorporates the YOLO model, which has been specially trained to detect landmarks of the face with high accuracy by processing 9,333 images from two image datasets, the Wider Face and Celebrity Face datasets. After applying augmentation and data cleaning it was turned into 62,938 images. Using the MediaPipe library, 102 facial landmarks were extracted, covering regions such as the eyebrows, eyes, upper and lower lips, and other facial landmarks. Each dataset image was labeled and used to train the YOLOv8n-pose model.

Key features of the system include the ability to control household appliances—such as lights, televisions, fans, and air conditioners—through simple eyebrow movements. For users who can speak, the Whisper library was employed to convert voice to text, enabling device control via voice commands. The Gemini API model serves two functions: detecting whether a user is awake or asleep and engaging in conversation with a custom AI assistant. The former assists speaking users, while the latter is designed for non-speaking users.

The model demonstrates exceptional performance metrics, achieving a facial detection accuracy of 98%, a pose estimation for 102 landmarks accuracy of 94%, and a voice command accuracy of 95%.

Declaration Associated with this Thesis



ID.: 171 - ICASDG2024
Date of Issue: 9-9-2024

ACCEPTANCE LETTER

Dear Ali Thabit Jabbar ,
Co-Authors : Nidhal Khdhair Abass and Bahija Khdhair Shuker.

Congratulations!

It is with great pleasure that we inform you that, your manuscript entitled:
Real-Time Face Detection and Segmentation: A Multifaceted Approach Integrating YOLOv8 and Color Space Techniques

has been **ACCEPTED** for participation in the ***3rd International Conference on Engineering and Science to Achieve the Sustainable Development Goals,*** scheduled to be held on September 25-26, 2024, Istanbul, Turkey and considered for publication in **AIP Conference Proceedings**.

Thank you for your valuable participation in the 3rd ICASDG2024.

ICASDG 2024

Prof. Dr. Ahmed Ghanim Wadday
The Head of Scientific Committee
Al-Furat Al-Awsat Technical University

Table of Contents

Dedication iii

Acknowledgement.. iv

Abstract v

Table of Contents vii

List of Tables..x

List of Figures xi

List of Abbreviations.. xiii

CHAPTER ONE: INTRODUCTION.. 1

1.1 Overview 1

1.2 Research Problem 3

1.3 Research Aim ……………………………………………………………………3

1.4 Challenges 4

1.5 Objectives...4

1.6 Related Work 8

1.7 Thesis Organization 4

CHAPTER TWO: THEORETICAL BACKGROUND 25

2.1 Overview 26

2.2 Machine learning...26

2.2.1 Supervised Learning...27

2.2.2: Deep Learning..29

2.3 Convolutional Neural Networks (CNNs)..30

2.3.1 Convolutional Layer..31

2.3.2 Pooling Layer..31

2.3.3 Fully Connected Layer..32

2.3.4 Activation Functions 33

2.3.5 Optimizer Selection...34

2.4 You Only Look Once (YOLO) 35

2.4.1 YOLO Architecture and Working Principle 36

2.4.2 YOLOv8n-pose 37
2.5 Facial Landmark Detection Techniques 38
2.5.1 MediaPipe for Landmark Detection 38
2.6 Speech Processing Systems 39
2.6.1 Whisper Model for Speech-to-Text (STT) 40
2.6.2 XTTS for Text-to-Speech (TTS) 41
2.7 Gemini AI Assistant 42
2.8 Raspberry Pi 4 43
2.8 Raspberry Pi 4 43
2.8.2 Applications and Use Cases 44
2.9 Telegram Bot API 45
2.10 Evaluation Metrics 46
2.10.1 Accuracy 46
2.10.2 Precision 46
2.10.3 Recall 47
2.10.4 F1 Score 47
2.10.5 Mean Average Precision (mAP) 47
2.10.6 Confusion Matrix 48
2.10.1 Accuracy 47
CHAPTER THREE: PROPOSED METHOD 49
3.1 Overview 50
3.2 The Architecture of the Proposed System 50
3.3 Image processing 51
3.4 Landmark Extraction 54
3.5 Data Splitting 56
3.6 Model training 57
3.7 Wake and Sleep Monitoring 58
3.8 Record the Movement 59
3.9 System Operating Capacity 60
3.9.1 System Readiness 60
3.9.2 Commands and Eyebrow Movement 60

3.9.3 Sync across Raspberry Pi 61
3.9.4 Voice Transcript and Text-to-Speech 61
3.9.5 System Deactivating...62
CHAPTER FOUR: RESULTS AND DISCUSSION 64
4.1 Overview 64
4.2 Hardware 64
4.3 Dataset and Preprocessing 65
4.3.1 Dataset Description 66
4.3.2 Data Augmentation 67
4.4 Landmark Extraction 68
4.5 YOLOv8n-Pose Model Results...73
4.5.1 Training Details...73
4.5.2 Face detection Results 74
4.5.2.1 F1 Score 74
4.5.2.2 Precision...76
4.5.2.3 Recall...77
4.5.2.4 mAP@0.5...78
4.5.2.5 Confusion matrix..79
4.5.2.6 Normalized Confusion Matrix 80
4.5.3 Pose Estimation Results 81
4.5.3.1 F1 Score 81
4.5.3.2 Precision...82
4.5.3.3 Recall...83
4.5.3.4 mAP@0.5...84
4.5.3.5 Training and Validation Metrics 85
4.5.3.6 Train Batch labels 85
4.5.3.7 Predictions and Validation Batch Labels 86
4.6 System Implementation and Integration 87
4.6.1 Real-time YOLOv8 Pose Estimation 88
4.6.2 Smart Home Control System Implementation 91
4.6.3 Integration of artificial intelligence assistants and speech processing..92

4.7 Comparison with Other Works 96
4.7.1 Accuracy Comparison..96
CHAPTER FIVE: CONCLUSION AND FUTURE WORKS 98
5.1 Conclusion..99
5.2 Future Works..101
REFERENCES...102

List of Tables

Table 1.1: Comparative Summary of Research 13
Table 3.1: Data augmentation parameters and values...42
Table 3.2: Split the dataset into two groups...45
Table 3.3: YOLOv8 pretrained Pose models 47
Table 3.3: Commands and Eyebrow Movement...50
Table 4.1: List of Led pins and corresponding devices 80
Table 4.2: List of voice commands..82
Table 4.3: Accuracy Comparison with Other Works...83

List of Figures

Figure 2.1: Supervised Learning Workflow..17

Figure 2.2: Deep Learning Neural Network Structure..19

Figure 2.3: Convolutional Layer Operation...20

Figure 2.4: Pooling Layer Operation. 21

Figure 2.5: Fully Connected Layer Operation 22

Figure 2.5: Activation Function..23

Figure 2.6: YOLOv8 Architecture 25

Figure 2.7: Non-Max Suppression in YOLOv3...26

Figure 2.8: YOLO-pose Architecture 27

Figure 2.9: MediaPipe landmarks numbers 28

Figure 2.10: Whisper Model Architecture. 29

Figure 2.11: XTTS training architecture overview..31

Figure 2.12: Raspberry Pi 4 Board Layout 33

Figure 2.13: Confusion Matrix Structure...37

Figure 3.1: Flowchart for Proposed System..40

Figure 3.2: Overview the model architecture of MediaPipe.....................................44

Figure 4.1: Hardware and Resource Utilization..54

Figure 4.2: Collection images from both datasets...55

Figure 4.3: The images after applying augmentation..56

Figure 4.4: MediaPipe facial landmarks 57

Figure 4.5: Highlighted points of landmarks selected from the 468..........................58

Figure 4.6: Landmarks categorized by facial regions 59

Figure 4.7: Examples of dataset images with applied landmarks 60

Figure 4.8: Text file format after normalization 61

Figure 4.9: BoxF1 Curve...64

Figure 4.10: Precision-Confidence Curve..65

Figure 4.11: Recall-Confidence Curve. 66

Figure 4.12: Precision-Recall Curve..67

Figure 4.13: Confusion Matrix...68

Figure 4.15: PoseF1 Curve....70
Figure 4.16: Precision-Confidence Curve of Pose estimation score....71
Figure 4.18: Precision-Recall Curve....73
Figure 4.19: Training and Validation measures....74
Figure 4.20: Sample of Training Batch Images. 75
Figure 4.21: Validation Batch Labels and Predictions....76
Figure 4.22: Real-time YOLOv8 Pose Estimation 78
Figure 4.23: After raising an eyebrow 79
Figure 4.24: After raising the eyebrow six times the LED 3 turned on. 79
figure 4.25 Raspberry pi 4 and others components....81

List of Abbreviations

Abbreviation	Description
AI	Artificial Intelligence
ANN	Artificial Neural Network
AP	Average Precision
CNN	Convolutional Neural Network
CV	Computer Vision
DL	Deep Learning
DNN	Deep Neural Network
FN	False Negative
FP	False Positive
FPS	Frames Per Second
IoT	Internet of Things
IOU	Intersection Over Union
ML	Machine Learning
mAP	Mean Average Precision
RPi	Raspberry Pi
ReLU	Rectified Linear Unit
STT	Speech-to-Text
SVM	Support Vector Machine
TTS	Text-to-Speech
VGG	Visual Geometry Group
YOLO	You Only Look Once

CHAPTER ONE

INTRODUCTION

1.1 Overview

The development and integration of machine learning and artificial intelligence technologies have fundamentally changed assistive technology's terrain for people with disabilities [1]. Among these innovations, face gesture detection systems have been demonstrated to be useful tools for enabling smart home control. These technologies enable users to use tools and engage more efficiently by way of a non-invasive, basic touch approach [2].

Face gesture recognition investigates motions and expressions by applying computer vision and machine learning methods. Since it promotes autonomy and quality of life, the ability to accurately identify and respond to facial expressions could be rather important for people with extreme physical limitations First preprogrammed responses are noted; real-time video input is gathered; facial landmarks are found; specific actions are looked at [3].

CNNs (Convolutional Neural Networks) are valuable for image classification and object recognition since they can automatically learn hierarchical features from raw image data [4]. Especially the YOLO (You Only Look Once) model, recent CNNs developments have greatly enhanced the capabilities of real-time face detection and posture evaluation, so suitable for applications needing significant accuracy and speed [5].

Apart from computer vision, assistive systems depend significantly on speech-processing technologies. Models like Whisper for speech-to-text conversion [6] and Coqui xtts-v2 (extended text-to-speech) [7] provide seamless voice interactions, adding another level of user access. These models alternate in providing aural answers and translate verbal commands into active instructions, therefore enabling a full interaction framework.

Including these technologies into a logical system for smart-home management requires diverse components and solutions. In interface with home appliances, for example, Raspberry Pi coupled with Relay Modules for device control generates a strong infrastructure capable of permitting real-time interactions. Such systems define their value since their ability to recognize and understand gestures and orders correctly helps to define consistent and responsive management of the home environment [8].

1.2 Research Problem

Difficulty communicating with paralyzed people and their inability to operate devices in the absence of family caregivers. Furthermore, current assistive technologies may not be flexible enough for people with severe physical disabilities, and thus a more accurate and user-friendly solution is needed.

1.3 Research aim

This research intends to develop a strong and efficient smart home control system based on eyebrow movements and speech recognition. The following objectives will guide this endeavor:

- Develop a machine learning model capable of precisely analyzing eyebrow motions in real-time, enhancing gesture recognition precision through the integration of facial landmark detection technologies.
- Create a seamless interface using Raspberry Pi and relay modules to connect smart home devices with detection systems.
- Integrate speech recognition technology to enable voice-based control and facilitate message delivery.
- Ensure the system's reliability under various real-world conditions, thereby guaranteeing accessibility for individuals with very limited physical abilities.

1.4 Research Challenges

Establishing a smart home control system based on eyebrow movements and speech recognition has many challenges, particularly in recognizing and interpreting facial movements. The primary challenges are:

1. Under various lighting conditions and several face expressions, exact identification and interpretation of small and delicate eyebrow movements can be difficult in gestural detection [9].

2. The system must respond instantly to human commands. This calls for optimizing the computing efficiency of the employed machine learning models [10].

3. A flawless interface between the facial gesture recognition system and various smart home devices will help to provide consistent communication and control [11].

4. Preserving great accuracy and dependability, the system needs to be adaptable enough to match several users with different face traits and movement patterns [12].

5. Correct gesture identification calls for the reduction of noise and interference in the video feed including background activities and unintended gestures [15].

6. Designing the system to be quite easily accessible and user-friendly for people with considerable physical limitations promises simplicity of use without requiring major setup or calibration [16].

1.5 Research Objectives

By using advanced machine learning models and approaches, this thesis significantly introduces the discipline of assistive technology and smart home control systems. The main objectives of this effort consist of:

1. Providing an easy and non-invasive interface for people with severe physical limitations by implementing a system that uses eyebrow movements for controlling different smart home devices.

2. Combining the strengths of MediaPipe for exact facial landmark detection with YOLOv8 for face detection helps to improve the accuracy and dependability of eyebrow movement awareness.

3. Optimizing the computational efficiency of the models, one might guarantee smooth interaction and instantaneous feedback for the consumers by achieving real-time processing and responsiveness.
4. Employing advanced speech-to-text and text-to-speech models (Whisper and Coqui xtts-v2).

5. Providing more extensive testing and validation of the system in numerous real-world settings allowed for an increase in model efficiency over several environments and user conditions.

6. Designing a user-friendly interface with easily accessible ensures that persons with physical restrictions can run the system with the least effort and best efficiency.

1.6 Related Work

This section provides a comprehensive analysis of previous research on smart home control systems.

Khattar, S., et al. [17] With an eye toward turning houses into Smart Homes, the thesis presents a Virtual Assistant called "Olivia" designed for household use. The device helps to simplify many house chores and presents the convenience of voice-activated interactions. Using a Raspberry Pi, a small and moderately priced computer, Olivia links cameras, speakers, microphones, and sensors, therefore transforming any house into a smart one. The research focuses on employing face recognition to differentiate between guests and homeowners, so enabling Olivia to be included into a smart door lock system and provide suitable responses. Olivia provides an SMS picture of missing properties together with a brief visitor caution. Furthermore, the system can be extended to include other smart devices like lighting and air conditioners, therefore improving their performance. For individuals with visual impairments, especially with its voice-activated features including weather updates, stock searches, math, jokes, and music playback, Olivia is extremely useful.

Fadillah, N. and A. Ihsan. [18] introduce a Smart Bed system that aims to assist paralyzed patients by utilizing voice recognition technology. For those with physical limitations, such the elderly or those with paralysis, this approach streamlines daily tasks. This system consists of speech recognition module, an Arduino unit, a relay circuit, and a changeable bed mechanism. The gadget first listens from a microphone for speech commands. The speech recognition module deals first with these spoken orders. To understand this module, it must go through a training phase learning to distinguish specific voice patterns. The Arduino unit turns on after a command is suitably detected. Then the Arduino gadget uses the relay circuit to start operations changing the height of the bed, thereby providing three various height alternatives for maximum comfort.

Raj, V., A. Chandran, and A.P. RS. [19] investigate a "IoT discusses art Home" system aimed to improve modern life by automation. The main drivers behind this approach are technical advancements and cellphones' general usage. Using the Internet of Things (IoT), the study intends to build an automated system linking multiple objects such as sensors, tablets, and cellphones. Once connected, these gadgets allow smart systems and services to satisfy minimum needs using intelligence. Google Assistant helps the research to provide a clever concept for creating a smart house employing many language voice commands. This system aims to control home appliances by including a smart door-unlocking mechanism.

Klaib, A.F., et al. [20] address an IoT-based smart home system incorporating voice interfaces and eye tracking to support elderly and special needs persons in controlling house equipment. Using Video Oculography based on Tobii technology and Azure Cloud for voice engagements, the device tracks eyeballs. By means of reflected infrared light patterns, the eye-tracking gadget detects the user's gaze and generates precise gaze estimations, therefore enabling interaction with domestic equipment.

Aqel, M.O., A. Alashqar, and A. Badra [21] work on a smart home automation system using eye tracking for paraplegic users. The idea is to provide people with disabilities—more specifically, those with quadriplegia—a means of using their eye motions to run household appliances. The system makes this contact using hardware components including webcams, Arduino controllers, and relay switchboards as well as computer application tools. Python programming language and OpenCV packages help to build algorithms that detect eye movements and blinking commands; a simple user interface is meant to be ease of use.

The research of Wassan, Z.A., et al. [22] focuses on developing an IoT-based smart home system intended for paraplegic people so they may control house equipment with eye blinks. With an eye toward improving the quality of life for those with physical constraints, the system combines several technologies and approaches to enable this interaction. Strategies and Instruments: Devices the Internet of Things, or IoT, is the system allowing wireless communication to remotely link and control house appliances. TCRT 5000 infrared sensors identify deliberate eye blinks, intended as orders to control appliances. Micro-controllers fit for Arduino behave depending on pre-defined patterns and process signals from the eye blink sensors.

Table 1.1 shows a concise synopsis of every publication together with their techniques and accuracy.

Table 1.1: Comparative Summary of Research

Ref	Author	Year	Techniques	Accuracy
[17]	**Khattar, S., et al.**	2019	Voice Recognition, IoT, AI	-
[18]	**Fadillah, N. and A. Ihsan.**	2020	Voice Recognition, Smart Bed	75%
[19]	**Raj, V., A. Chandran, and A.P. RS.**	2019	Voice Recognition, IoT	-
[20]	**Klaib, A.F., et al.**	2019	Eye Tracking, Voice Interfaces	93.2%
[21]	**Aqel, M.O., A. Alashqar, and A. Badra.**	2020	Eye Tracking, Python	-
[22]	**Wassan, Z.A., et al.**	2021	Eye Blink, IoT	80%

1.6 Thesis Organization

This thesis consists of five chapters structured as follows:

- Chapter 2 provides an overview of the IoT system background and important ideas in machine learning, supervised learning, deep learning, and the technologies used in this research. This chapter introduces a brief IoT.

- Chapter 3 focuses on developing an IoT system a smart home device as a case study to help disabled people. It addresses in depth the architecture of the system and the integration of many components including whisper, Coqui MediaPipe, Gemini, and YOLOv8 to reach real-time facial movement recognition.

- Chapter 4 shows the experimental setup and results. It presents the performance evaluation of the developed system using evaluation metrics with careful analysis and discussion of the findings.

- Chapter 5 provides a summary and the findings of the research presented in this thesis. This chapter also discusses the ideas for future work.

CHAPTER TWO

THEORETICAL BACKGROUND

2.1 Overview of IoT system and its components

To understand the IoT system components, this chapter provides a comprehensive review of the theoretical background and related work about IoT applications essential to understanding the techniques and methodologies used in this research. The goal of this thesis is to develop a smart home device system that is capable of helping disabled people by using eyebrow movement and speech recognition. The following sections will cover the key techniques and tools used for eyebrow movements, face detection and pose estimation, speech processing, and smart home integration. Each section will delve into the specific technologies and their roles in this research.

2.2 Overview of smart home devices system classification

Classification techniques play an important role in the field of machine learning (ML) aims to create algorithms adept at learning from and generating predictions on data. Unlike traditional programming, in which clear instructions are given to do tasks, ML algorithms learn from data to detect patterns and make decisions with the least human intervention. Therefore, it is very important to extract useful features from the face image and then use those features for the classification [23].

The basic idea underlying machine learning is the construction of models able to extend from provided data to unforeseen circumstances. Usually spanning many significant steps, this process comprises data gathering, data preprocessing, model selection, training, evaluation, and application [24].

Data collection involves gathering relevant information from multiple sources to provide raw materials for models of training machine learning. Data preparation comes next, when the acquired data is cleaned and processed to remove noise and disparities, therefore ensuring that it is in a suitable state for research [25].

Selection of models is the process of choosing the appropriate technique in view of the kind of problem and data characteristics. Currently there are numerous methods with benefits and drawbacks depending on the specific task at hand [26]. Following model selection, the training phase begins wherein the model learns underlying patterns and relationships in a dataset.

Many assessment criteria—accuracy, precision, recall, F1 score—are used to assess the model after it is trained. These standards direct model performance and help to determine if it is suitable for application. Always learning and changing as more data becomes available, the model is finally used to provide projections or decisions on new, unknown data [27].

2.2.1 Supervised Learning

In machine learning, supervised learning is a fundamental technique wherein a labeled dataset helps a model to learn a mapping from inputs to outputs. Applications of this kind of learning abound in tasks like classification and regression [28]. for example, in image classification, a model is trained to recognize objects inside pictures using labeled samples [29].

Training and test sets separate apart a dataset in a supervised learning system. Usually applying linear regression, decision trees, support vector machines (SVM), and neural networks [30], the model trained on the training data by reducing the error between the predicted and actual outputs. Test set evaluations of performance help the model to be generalized to fresh data [31].

2.3 Structure of a smart home devices system classification

in this thesis, YOLO models and speech recognition models have been used to achieve the objectives of this study.

2.3.1 YOLO Architecture and Working Principle

You Only Look Once (YOLO) a modern, real-time object recognition technology. Unlike traditional object recognition systems that apply a classifier to an image at various locations and scales, YOLO frames object identification as a single regression problem, straight from picture pixels to bounding box coordinates and class probabilities [32].

The algorithm of YOLO divides the input image into a grid of cells, and each cell forecasts a certain number of bounding boxes along with confidence scores for them. These confidence scores capture the conviction of the model on the correctness of the box and its conviction that it comprises of an item. Simultaneously for every one of the multiple enclosing boxes, the system expects class probability [33]. This consistent approach greatly speeds up detection and streamlines the procedure.

YOLO's architecture is built on one neural network that concurrently forecasts many bounding boxes and class probabilities for those boxes. Running the complete image in one forward pass, the network generates surprisingly fast detection times.

In the fundamental architecture, there are many convolutional layers then completely coupled layers [34].

YOLO's architecture consists of three primary components: the detection network, the post-processing phase, and the feature extraction network [35]. Usually, ImageNet pre-trained, CNN builds the convolutionally based feature extractor network. This network generates from the input picture whole high-level characteristics [36].

These properties let the detection network project border boxes and class probabilities. One of the most often used variations of YOLOv8 uses a sequence of neural layers to anticipate bounding boxes at three distinct levels, hence extending its capacity to distinguish objects of diverse sizes [37]. Figure 2.1 depicts YOLOv8's construction.

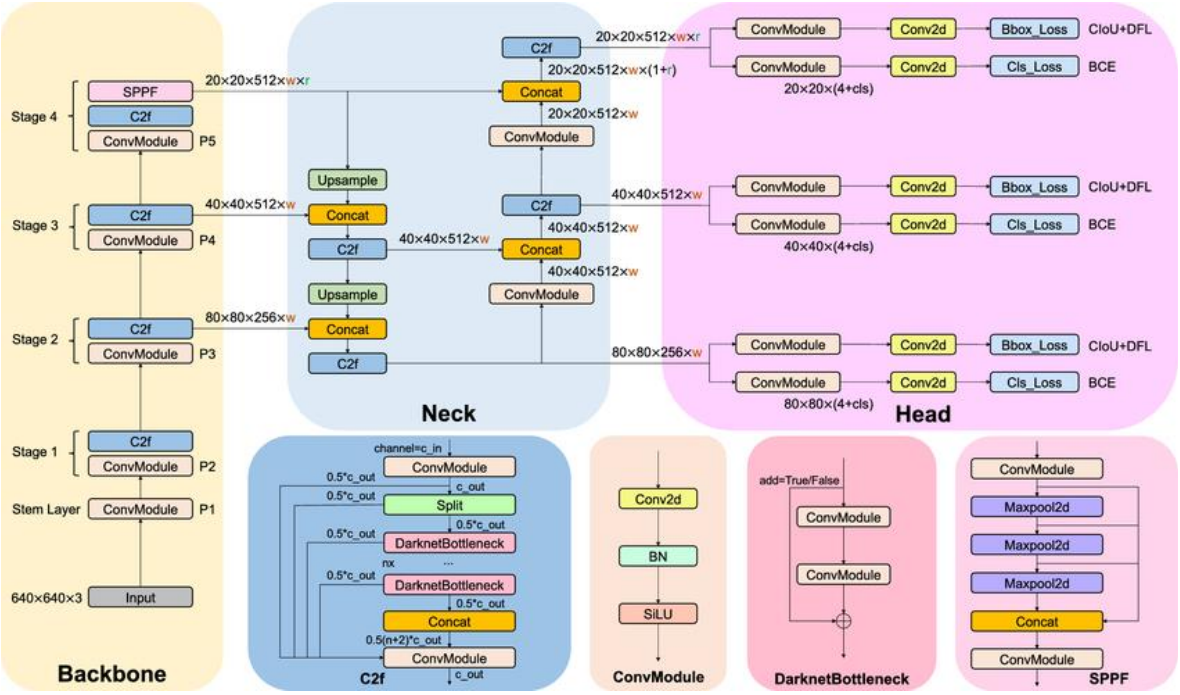


Figure 2.1: YOLOv8 Architecture [38].

Every item is only once found because post-processing uses non-max suppression to remove too many bounding boxes [39]. This pretty clearly shown in Figure 2.2: this extremely critical interval provides very strong detection accuracy.

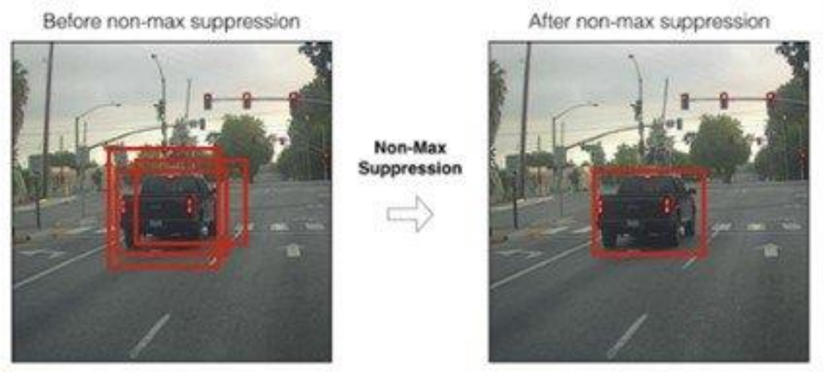


Figure 2.2: Non-Max Suppression in YOLOv3 [10].

The end-to-differentiable nature of YOLO makes fast optimization easy as well. Its remarkable balance of speed and accuracy equips it for real-time applications like robotics, security monitoring, and autonomous driving [41].

YOLOv8n-pose is a version of YOLOv8 particularly meant for positions requiring pose estimation. It extends YOLOv8's great object recognition capabilities to recognize and assess human poses. Applications in human-computer interaction, sports analysis, healthcare as well as others profit tremendously from this idea [42].

2.3.2 Facial Landmark Detection Techniques

Many computer vision systems are based on facial landmark detection, which enables perfect identification of significant face areas like eyes, nose and mouth. Many often-used techniques and frameworks for facial landmark identification have benefits and specific applications [43]. In this structure, the MediaPipe library has been used for landmark detection.

2.3.2.1 MediaPipe for Landmark Detection

Designed by Google, MediaPipe is a flexible architecture providing cross-platform, real-time machine-learning solutions for live and broadcast media [44]. It is very effective for facial landmark identification, MediaPipe face recognition, and face mesh generation [45] MediaPipe benefits especially on mobile devices as its speed and efficiency enable real-time processing even there [46]. Its cross-platform compatibility also makes flawless integration into numerous applications feasible. Figure 2.3 shows the facial landmark detection capability of MediaPipe.

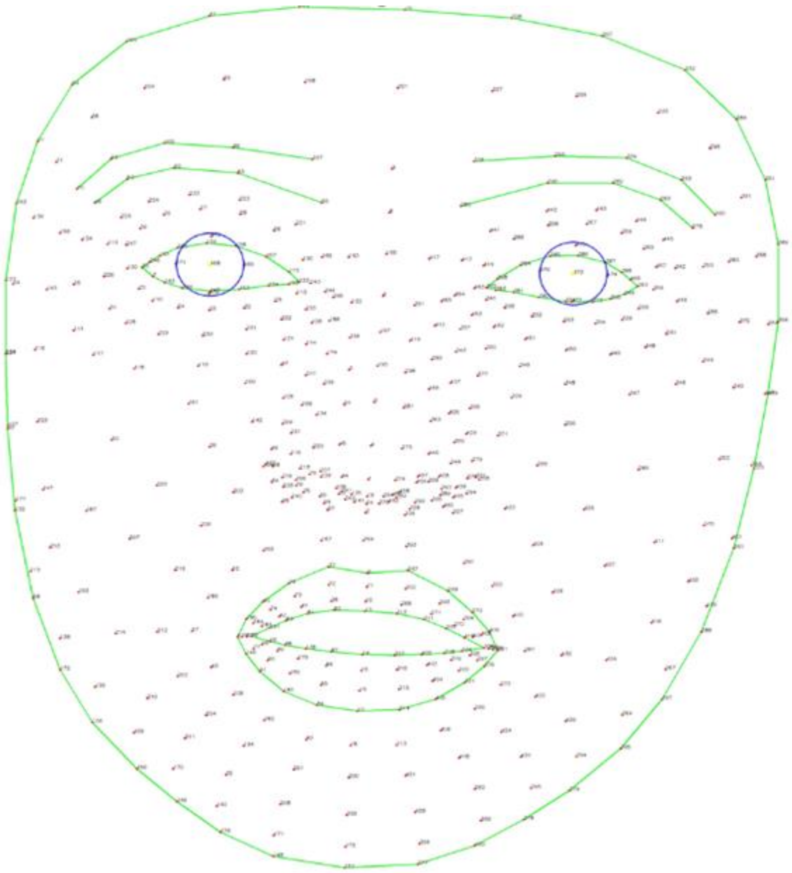


Figure 2.3: MediaPipe landmarks numbers [47].

2.3.3 Speech Processing Systems

Speech processing systems are fundamental in many diverse applications including voice assistants, transcription services, and accessibility assistance for persons with disabilities [48]. Two basic components of speech processing systems are Text-to-Speech (TTS) and Speech-to-Text (STT) [49]. This thesis employs the Whisper model for converting speech-to-text and the Coqui-XTTS model for converting text-to-speech to recognize voice commands for disabled people.

2.3.3.1 Whisper Model for Speech-to-Text (STT)

The whisper model designed by OpenAI is a powerful Speech-to-Text (STT) system that very precisely converts whispered speech into written text. Whisper employs deep learning methods on a diverse dataset spanning several languages and dialects, so it is robust and versatile for different application scenarios. When audio input is received, it undergoes several preprocessing steps, including noise reduction and normalization, to enhance clarity. The model then transforms the audio waveform into a series of features that represent the speech content. Using attention mechanisms, the model identifies patterns in the data, allowing it to focus on relevant portions of the audio while disregarding irrelevant noise. This enables the Whisper model to distinguish between different speech forms and accurately transcribe spoken words, even in challenging acoustic environments. The extensive training on a diverse set of multilingual data allows the model to adapt to various accents and dialects, making it an effective tool for a wide range of speech recognition applications [50].

Among Whisper's key advantages are her ability to correctly control many speaking forms and noisy surroundings. Long-range connections in speech are captured by its transformer-based construction, hence enhancing transcription accuracy [51]. Figure 2.4 shows the architecture of the Whisper model.

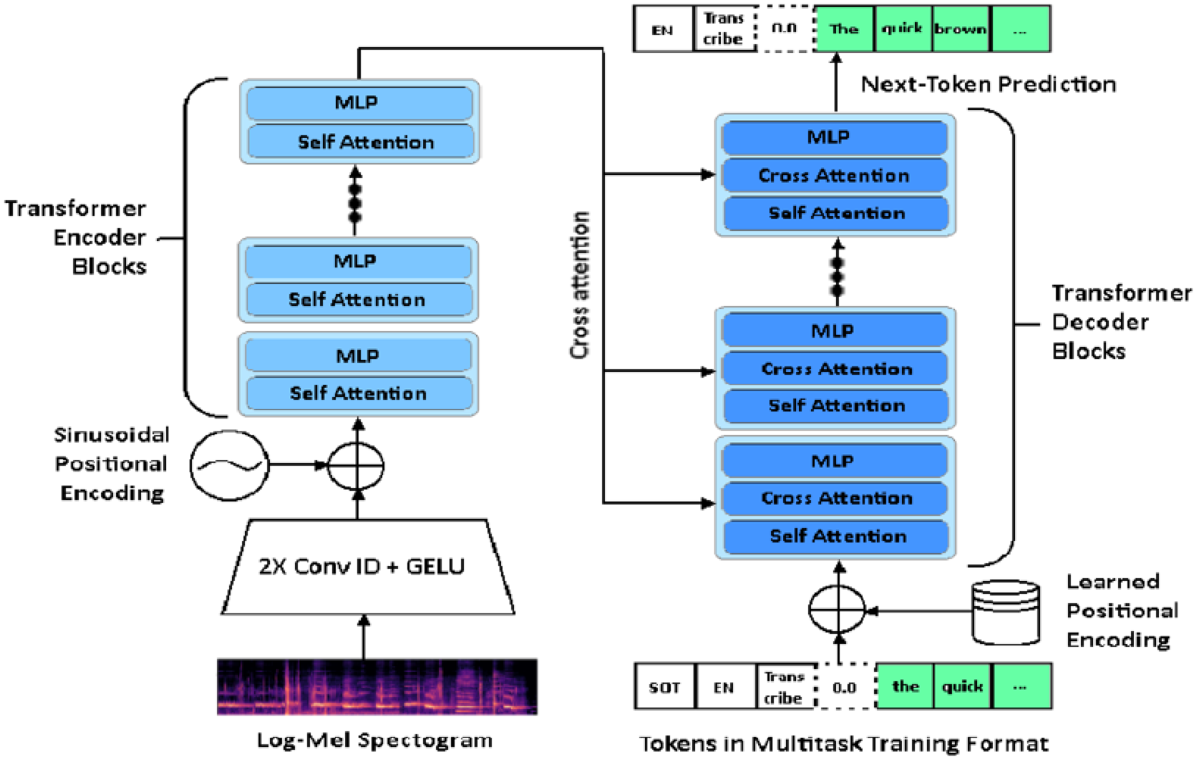
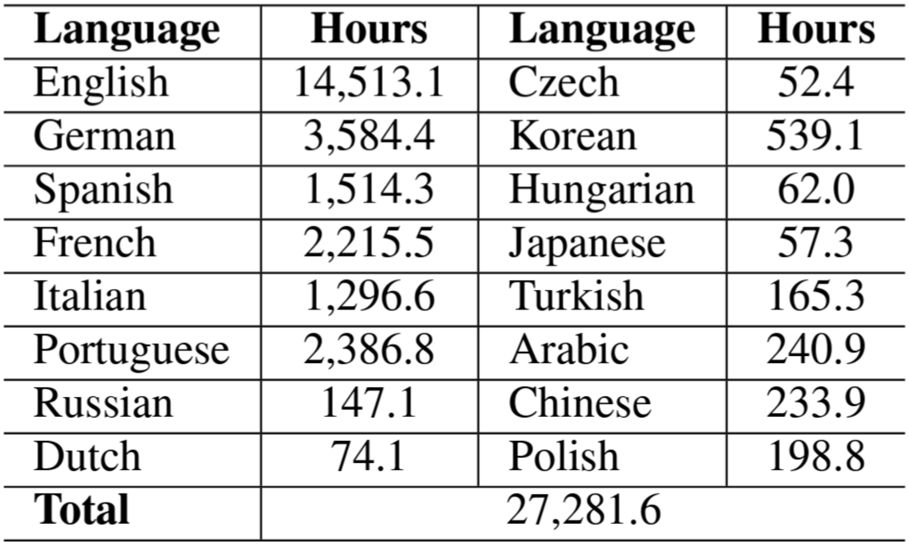


Figure 2.4: Whisper Model Architecture [52].

2.3.3.2 XTTs for Text-to-Speech (TTS)

Perfect in zero-shot multi-speaker TTS and multilingual training, XTTS is a state-of-the-art Text-to-Speech technology. Based on the Tortoise model, this one adds some fresh changes to allow high accuracy handling of multilingual workloads [7]. XTTS demonstrated superior performance in the majority of cases, surpassing previous benchmarks, following training in sixteen different languages. Table 2.1 presents the amount of time needed for each language.

Table 2.1: the XTTS dataset provides the Number of hours of language data collected for training [7].



Language	Hours	Language	Hours
English	14,513.1	Czech	52.4
German	3,584.4	Korean	539.1
Spanish	1,514.3	Hungarian	62.0
French	2,215.5	Japanese	57.3
Italian	1,296.6	Turkish	165.3
Portuguese	2,386.8	Arabic	240.9
Russian	147.1	Chinese	233.9
Dutch	74.1	Polish	198.8
Total	27,281.6		

XTTS is flexible and fit for many uses as it can enhance voice cloning and allow quicker training and inference, thus improving its core strengths. Figure 2.5 displays XTTS training architecture overview details.

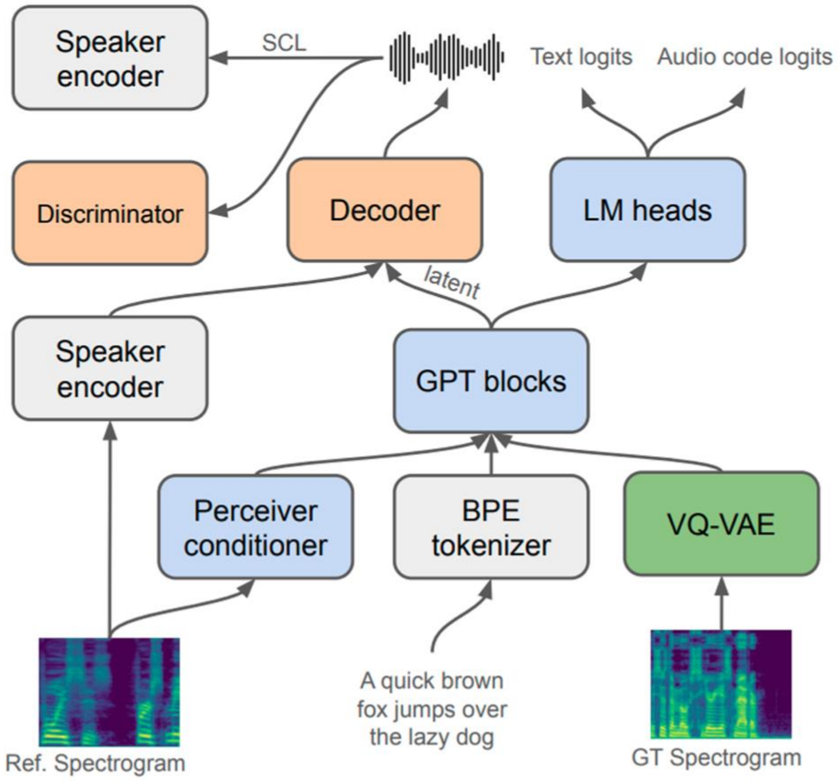


Figure 2.5: XTTS training architecture overview [7].

2.3.4 Gemini AI Assistant

Gemini is a powerful artificial intelligence assistant aimed to support efficient and natural human-computer interaction. Gemini provides a good user experience by integrating proactive support, contextual awareness, and natural language understanding [53] and is meant to meet numerous needs.

Gemini uses advanced natural language processing (NLP) and machine learning (ML) techniques to respond to consumer inquiries. It can handle a wide range of tasks from answering factual questions and generating reminders to controlling smart home gadgets and providing custom recommendations. Highly interactive, the AI assistant reacts based on user preferences.

2.3.5 Raspberry Pi 4

The Raspberry Pi 4 is a development of the well-known single-board computer series started by the Raspberry Pi Foundation. Performance, connectivity, and capacity-wise it displays a clear increase over its forebears. With a speed of 1.5 GHz, the quad-core ARM Cortex-A72 CPU included in the Raspberry Pi 4 provides quite a high processing capacity for a variety of applications. It comes in 2GB, 4GB, and 8GB of RAM among other memory configurations to satisfy different performance needs [54]. This flexibility helps it to properly handle numerous circumstances and ever more difficult tasks.

One of its best characteristics is the many connection choices of the Raspberry Pi 4. High-speed data transmission and peripheral connection are made possible by dual-band 802.11ac wireless networking, Bluetooth 5.0, Gigabit Ethernet, and four USB ports—two USB 3.0 and two USB 2.0. Particularly helpful for multimedia applications and complicated visual outputs, the device now has twin 4K HDMI outputs, therefore allowing the connecting of two high-resolution displays [55].

While the official Raspberry Pi OS (previously Raspbian) is the most often used operating system, the Raspberry Pi 4 provides a range of alternatives. This running system offers a decent development environment and a basic user interface. Moreover, the General-Purpose Input/Output (GPIO) pins of the board enable direct control of motors, sensors, and other peripherals, therefore allowing hardware projects [56].

2.3.5.1 Applications and Use Cases

The versatility of the Raspberry Pi 4 qualifies it for a broad spectrum of uses. Programming and electronics are taught mostly in schools. For enthusiasts and students, its cheap cost and simplicity of use make it the perfect instrument. Furthermore, used in many IoT (Internet of Things) initiatives, the Raspberry Pi 4 serves as a central hub for gathering and analyzing sensor data [57]. Figure 2.6 shows the board of Raspberry Pi 4.

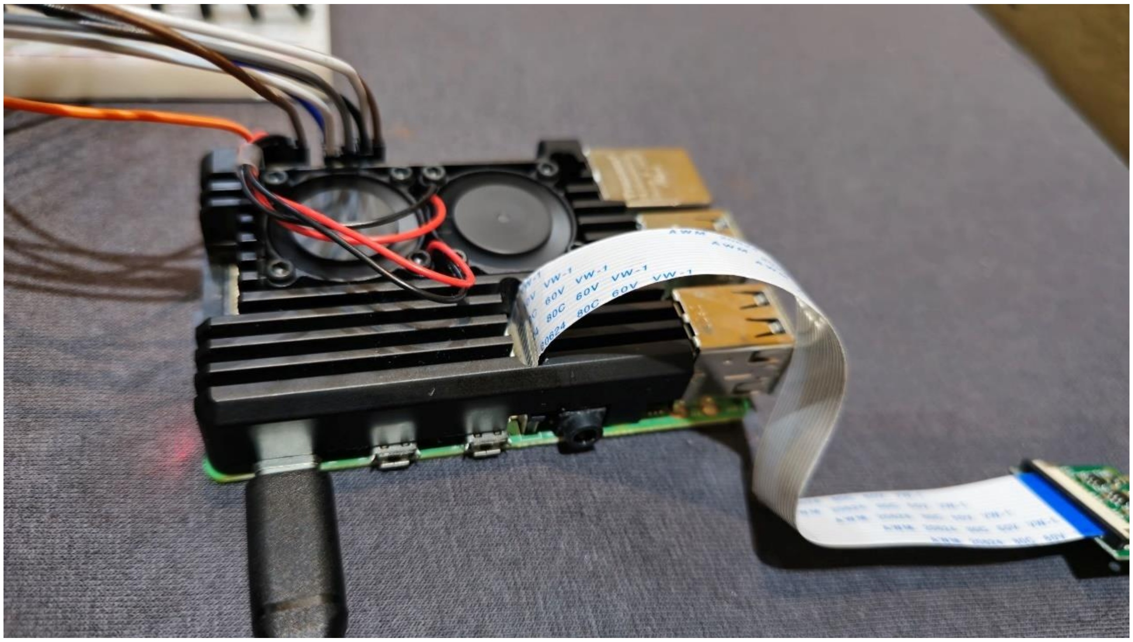


Figure 2.6: Raspberry Pi 4 Board Layout with Case.

The Raspberry Pi 4 has many restrictions despite its several advantages. It has powerful software features, but it could not be as good as premium desktop PCs. Furthermore, running resource-intensive programs might call for appropriate cooling mechanisms to avoid overheating [54].

2.3.6 Telegram Bot API

With Telegram's strong UX and Telegram Bot API, which allows developers create bots able of interacting with users on the Telegram chat network, These bots may handle a wide range of chores from messaging and alerting to command processing and service integration.

One may register a bot account using the Telegram Bot API by way of the BotFather, a particular bot provided by Telegram for bot management. Once the bot is constructed, developers have an API token allowing them to programmatically access its capabilities. The API's support of many message kinds—text, audio, video, and documents [58] helps to create rich interaction options.

The Telegram Bot API helps significantly in creating interactive applications demanding real-time communication. Human inputs allow bots to be trained to respond to user requests, provide automated updates, and handle specific tasks. The Telegram Bot API is thus a wonderful tool for customer help, automated notifications, and interactive services [59]. In this study, the Telegram Bot API was used to enable real-time communication, allowing paralyzed patients to send a voice message that is converted to text and sent to a specific person whose chat ID was previously taken or an automated message if the patient is non-spoken, providing seamless integration of message-based interactions with the smart home system.

2.4 Evaluation Metrics

Assessment criteria mostly define the performance of machine learning models. These steps expose areas requiring improvement and provide numerical figures that help to understand a model's performance. Accuracy, precision, recall, F1 score, mean average precision (mAP), and the confusion matrix rank among the most regularly used evaluation tools have been discussed in chapter 4.

CHAPTER THREE

PROPOSED METHOD

3.1 Overview

This chapter presents the proposed method for creating a smart home control system using eyebrow movements or voice commands to assist paralyzed individuals. The procedure starts with preparing and augmenting data for paralyzed individuals for those who cannot speak, then moves on to splitting the data and training the YOLOv8n-pose model. The trained model is then used to identify and analyze eyebrow movements, allowing control of home appliances. For those who can speak, voice commands are used to operate the home or to request assistance, which will be discussed later.

3.2 The Architecture of the Proposed System

The proposed smart home management system architecture is specifically designed to provide powerful and effective control through eyebrow movements and voice commands. The system starts by preparing the dataset, which includes feature extraction and image analysis. It then progresses to using these actions, which provide a unified and optimized dataset that enhances the creation of effective models. The training is followed by data separation for the YOLOv8n-pose model. The trained model starts the system, which then continuously monitors the user’s state by checking for wake-up/sleeping. Real-time detection of eyebrow movements triggers several actions: the smart home is controlled by eyebrow movements, movements 1 to 6 turn on the electrical appliances connected to the Raspberry Pi, the seventh movement sends an automatic message requesting help from the family that cares for the user via the Telegram-bot, the eighth movement is used when the user wants to go to sleep, so the system goes to the first step when checking whether the user is asleep or awake. If the user can speak, there is no need to use eyebrow movements and shorten them by applying voice commands to operate home appliances or send a message using voice and it reaches a family member via

Telegram in the form of a text message or even when he wants to voice chat with the AI assistant, the system allows the user to do so. With an emphasis on the seamless integration of data processing, model training, and system operations, the flow chart in Figure 3.1 illustrates the many steps involved. The design of this system emphasizes its ability to provide autonomous control to people with paralysis, thereby improving their quality of life using contemporary technology.

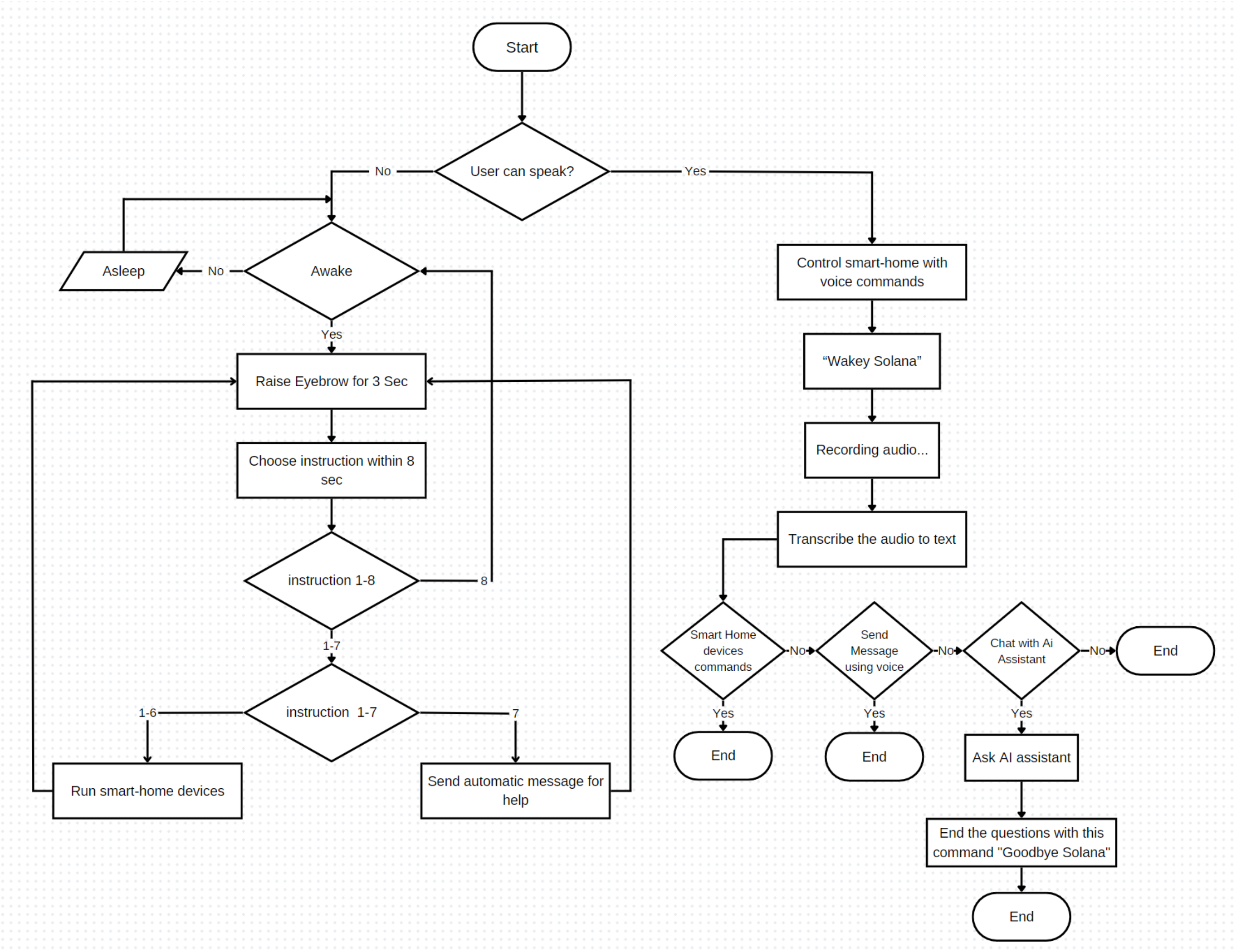


Figure 3.1: Flowchart for Proposed System.

3.3 Image processing

Image processing must be done first for the YOLOv8n-pose model dataset. Standardizing the pictures and the other improvement techniques used in improving the dataset helps one to further increase the generalizability and robustness of the model.

Resizing photos from Wider Face Dataset and the Celebrity Face Image Dataset to a standard size of 400x400 pixels has been done. Effective learning and processing from the dataset are allowed by this reduction producing homogeneity in the dataset.

Furthermore, improving computing speed during model training is an image size standardizing operation. To feed Yolov8 with more data, the proposed system will augment the data by using some of the data augmentation techniques to increase the size and volatility of the data set. Enhancement techniques include horizontal flipping, rotation, brightness adjustment, translation, cropping, adding noise, blurring, color jitter, and random scanning. These methods are essential to make the model robust against many types of noise and variations that may occur in real-world settings. All boosting techniques and their associated parameters and values are shown in Table 3.1:

Table 3.1: Data augmentation parameters and values.

Augmentation Technique	Parameter(s)	Value	Applying
Horizontal Flipping	**-**	**-**	
Rotation	**Angle**	**-30 to 30 degrees**	
Brightness Adjustment	**brightness**	**-50 to 50**	

Translation	**tx, ty**	**-50 to 50 pixels**	
Noise Addition	**Noise**	**mean=0, stddev=25**	
Shear	**shear_factor**	**-0.2 to 0.2**	
Blur	**-**	**kernel size=5x5**	

Color Jitter	**Brightness, Contrast**	**-20 to 20**	
Random Erasing	**Probability (p)**	**0.8**	

3.4 Landmark Extraction

The proposed smart home management system depends on extracting landmarks, which calls for the identification and tracking of many facial characteristics necessary for eyebrow movement detection. This part looks at the whole landmark extraction procedure, its importance, and the instruments used to get precise findings.

Landmark extraction focuses on the vital parts of a face—including the eyes, eyebrows, nose, cheekbones, and surrounding areas—which are required for facial identification and research. The major objective of our system is to detect eyebrow movements, which function as control signals. The proposed system can distinguish between numerous instructions based on the minute facial expression changes of the user by correctly locating and monitoring these landmarks.

One of the most powerful and fastest libraries for facial detection is Google's MediaPipe library. This library is particularly well-suited for real-time applications due to its efficacy and compatibility with a variety of platforms, including mobile devices.

MediaPipe identifies 468 specific landmarks within the detected face area. The mesh model is shown in Figure 3.2. The proposed system used MediaPipe as a raw material for Yolov8. So basically, the extraction of landmarks is used to create datasets from any images that contain faces and train them after that using the Yolov8 model.

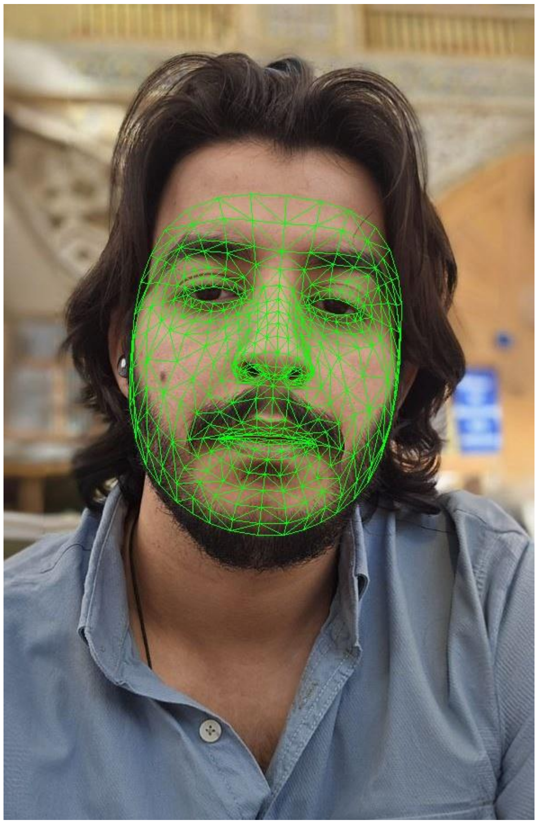


Figure 3.2: Overview the model architecture of MediaPipe.

The proposed system chooses 102 landmarks only. X and Y coordinates of each of the 102 landmarks can be obtained from any image. The precise mapping of the face characteristics, which these coordinates provide, is essential for tracking movements.

The obtained coordinates are normalized to a conventional scale to ensure uniformity across multiple photographs. Typically, this normalization scales the coordinates to a defined range, thereby maintaining consistency regardless of the original picture size, which is typically [0, 1]. Additionally, a five-pixel margin is included around areas of approximately facial landmarks to establish a bounding box around the face.

A structured dataset is created from the obtained and standardized landmarks. This dataset consists of the X and Y coordinates of the landmarks as text files for future use. The consistent and trustworthy landmark information given by the structured dataset helps one to efficiently train the YOLOv8n-pose model.

3.5 Data Splitting

It is required while preparing the dataset for the YOLOv8n-pose model. This ensures that the model is tested properly and can generalize well about new data. The dataset consists of three sets: training, validation, and testing. The initial division is shown in the following Table 3.2:

Table 3.2: Split the dataset into two groups instead of three group

Dataset Split	Number of Images and Labels	Percentage
Training Set	51,846	82.4%
Validation and Testing Set	11,092	17.6%
Total	62,938	100%

The validation and testing sets are combined into a single validation set. Combining them makes the training and evaluation speed up the process by providing a larger validation set, resulting in more consistent performance metrics throughout model development. This strategy ensures that the model is regularly evaluated throughout the training phase and reduces the effort of retaining separate validation and testing sets.

3.6 Model training

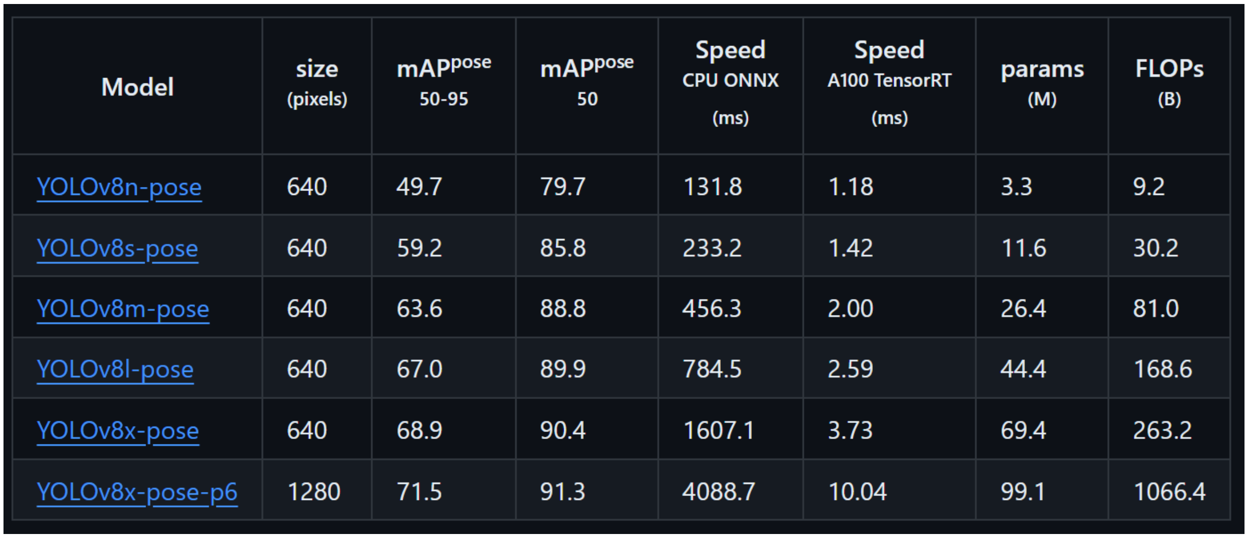
Because of its great speed, the proposed system trains using the YOLOv8n-pose model. For smart home systems, it is perfect; it runs with medium hardware devices and do not need strong hardware. With the YOLOv8 series, many versions are perfect for different performance and processing needs.

YOLOv8 posture models are meant for perfect and quick identification of key areas. This is necessary for understanding face landmarks including eyebrow motions. Applications requiring real-time interaction find YOLOv8 models perfect as they can control high-dimensional inputs and make quick inferences. Among the YOLOv8 models, YolOv8n-pose was selected because of the best trade-off between performance and computational economy.

With an inference speed of 1.18 ms per picture, YOLOv8n-pose is well fit for real-time processing. With an average accuracy (mAPose 50-95) of 49.7% and mAPose 50 of 79.7%, it guarantees consistent detection of face landmarks. Although the model is lightweight enough to operate on devices with little computational capability, it has 3.3 million parameters and 9.2 billion FLOP operations, therefore delivering strong performance.

Selecting YOLOv8n-pose guarantees that the system can run effectively in real-time, hence offering precise and quick reactions to user inputs. Emphasizing the balance between speed, accuracy, and computational demands, Table 3.3 presents a comparison of many YOLOv8 mode models, therefore identifying YOLOv8n mode as the optimum option for this usage.

Table 3.3: YOLOv8 pretrained Pose models



Model	size (pixels)	mAP$^{\text{pose}}$ 50-95	mAP$^{\text{pose}}$ 50	Speed CPU ONNX (ms)	Speed A100 TensorRT (ms)	params (M)	FLOPs (B)
YOLOv8n-pose	640	49.7	79.7	131.8	1.18	3.3	9.2
YOLOv8s-pose	640	59.2	85.8	233.2	1.42	11.6	30.2
YOLOv8m-pose	640	63.6	88.8	456.3	2.00	26.4	81.0
YOLOv8l-pose	640	67.0	89.9	784.5	2.59	44.4	168.6
YOLOv8x-pose	640	68.9	90.4	1607.1	3.73	69.4	263.2
YOLOv8x-pose-p6	1280	71.5	91.3	4088.7	10.04	99.1	1066.4

3.7 Wake and Sleep Monitoring

A fundamental part of the smart home management system, wake/sleep monitoring assures the system responds only when the user is awake. This capacity enhances system efficiency and helps to avoid unintentional actions during sleep.

The proposed system uses the Gemini API to determine if the user is asleep or awake. That's done by capturing a facial image of the user every five seconds using a camera. The Gemini API has been designed to provide fast results based on photo processing, thereby enabling simple counting: "Awake" or "Asleep". Gemini is supposed especially to provide these quick responses.

If the eyes are regularly observed closed throughout a five-minute, the algorithm decides that the person is asleep. Conversely, the system senses the user's awareness if, in most frames during this time, the eyes are open. Should the user be found to be asleep, the system might enter low-power mode to save energy.

Depending on eyebrow movements, the system gets ready to run if the user is awake. Constant monitoring assures that the system can respond swiftly when the user wakes up, therefore providing a quick and simple approach for smart home control.

3.8 Record the Movement

The detecting procedure starts with the user in a neutral posture and awake recording a reference frame. Later comparisons start from this first frame as a baseline. In every collected picture, the algorithm constantly gauges the eyebrow-to-eye distance. Distance Should the surpass the baseline by more than two pixels, the algorithm notes this as a movement.

This technique of motion detection based on pixel distance guarantees sensitivity to even minute eyebrow motions, therefore enabling exact and consistent control. The technology offers an easy and responsive interface for managing smart home appliances as it is designed to identify certain patterns of these motions to carry out related actions.

3.9 System Operating Capacity

Combining many functions, the proposed smart home management system provides paralyzed individuals with a responsive and simple interface for surrounding environment control. The system operations are supposed to provide successful interaction by means of real-time feedback, seamless interconnection with various devices and services, and eyebrow movements—which ensure effective engagement.

3.9.1 System Readiness

When it detects that the user is awake, the system enters the standby state. The user can raise his eyebrows for three seconds to indicate that the system is ready. This intentional gesture ensures intentional activation and prevents accidental stimuli.

The user has an eight-second window once the system is ready to especially wiggle their eyebrows to run various smart home appliances and send automatic messages. The system immediately detects and analyzes these movements throughout this time thereby allowing the user to do the required actions. After the period ends, the system goes to standby.

3.9.2 Commands and Eyebrow Movement

The system detects seven different gestures, each matching a certain instruction. The detecting mechanism consists of measuring the distance between the eyebrows and the eyes. The technique marks movement should the distance exceed the baseline by more than two pixels. The following Table 3.4 shows the movements and their commands:

Table 3.3: Commands and Eyebrow Movement

Command	LED Pin	Device
1	17	LED 1
2	18	LED 2
3	27	Ceiling Fan
4	22	TV
5	23	Air Conditioner
6	24	LED 3
7	-	Send Automatic message for help

3.9.3 Sync across Raspberry Pi

Coupled via a Wi-Fi connection to a Raspberry Pi, the system may run several household appliances. Reaching signals from the laptop of charge of the smart home management system, the Raspberry Pi controls the devices via relay modules. This design guarantees responsive and continuous management of the surrounding living environment.

3.9.4 System Deactivating

Movement eight allows the user to switch off the system, which then resumes observing the user's wake/sleep condition. This feature is very useful when the user needs to sleep as it ensures that the system saves energy and avoids unintentional activity during this moment.

3.9.5 Voice Command Processing

For individuals with paralysis who retain the ability to communicate verbally, the system incorporates advanced speech recognition technologies to facilitate greater independence and enhance communication capabilities. Utilizing OpenAI's Whisper model, the system converts spoken commands into written text with high accuracy, enabling users to control smart home devices, send messages, or engage with the AI assistant for interaction and entertainment.

The Whisper model processes audio inputs by capturing voice commands and transcribing them into text, which is then linked to specific actions within the system. For example, users can turn on lights, control various appliances, or initiate communication by simply stating their commands. When a command is recognized, the system processes the instruction and performs the corresponding action, such as activating home devices or sending a message via Telegram. This seamless audio-to-text translation enables effective device control and message delivery for users, enhancing autonomy and access to essential services.

Additionally, the Whisper model’s robustness allows it to function effectively across diverse acoustic environments, maintaining command accuracy even in varied noise levels and speaking styles. Once transcribed, text-based commands are either directed to the smart home controller for device activation or forwarded through Telegram for messaging, further demonstrating Whisper’s role in making the system accessible and responsive to the unique needs of individuals with limited physical mobility. This enables users to fully engage with their environment, enhancing overall usability and independence.

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Overview

This chapter presents the results and analysis of a smart home control system based on eyebrow movements and speech recognition for paralyzed individuals who cannot speak and voice commands for paralyzed individuals who can say. The system was evaluated using the YOLOv8n-pose model for real-time keypoint detection. Performance metrics include F1 Score, Precision, Recall, and mAP@0.5 for face detection, pose estimation, and voice command. The results demonstrate the system's effectiveness in accurately detecting and responding to eyebrow movements, enabling paralyzed individuals to control their environment.

The evaluation was conducted on a personal laptop using JupyterLab within an Anaconda environment. This setup ensured a controlled and reproducible environment for testing and validating the system's performance.

4.2 Hardware requirement

To sufficiently train and validate the YOLOv8n-pose model, the smart home management system was trained and validated on a high-performance computer employing the necessary processing capacity and memory. Maintaining extraordinary performance and precision, the hardware layout ensured that the training process ended in the appropriate time range.

Figure 4.1 presents the hardware and resource consumption requirements, therefore providing a clear image of the capabilities of the system as well as the resources consumed in the phases of training and validation. This includes the overall period spent as well as the time taken for every epoch—which will be detailed in detail later.

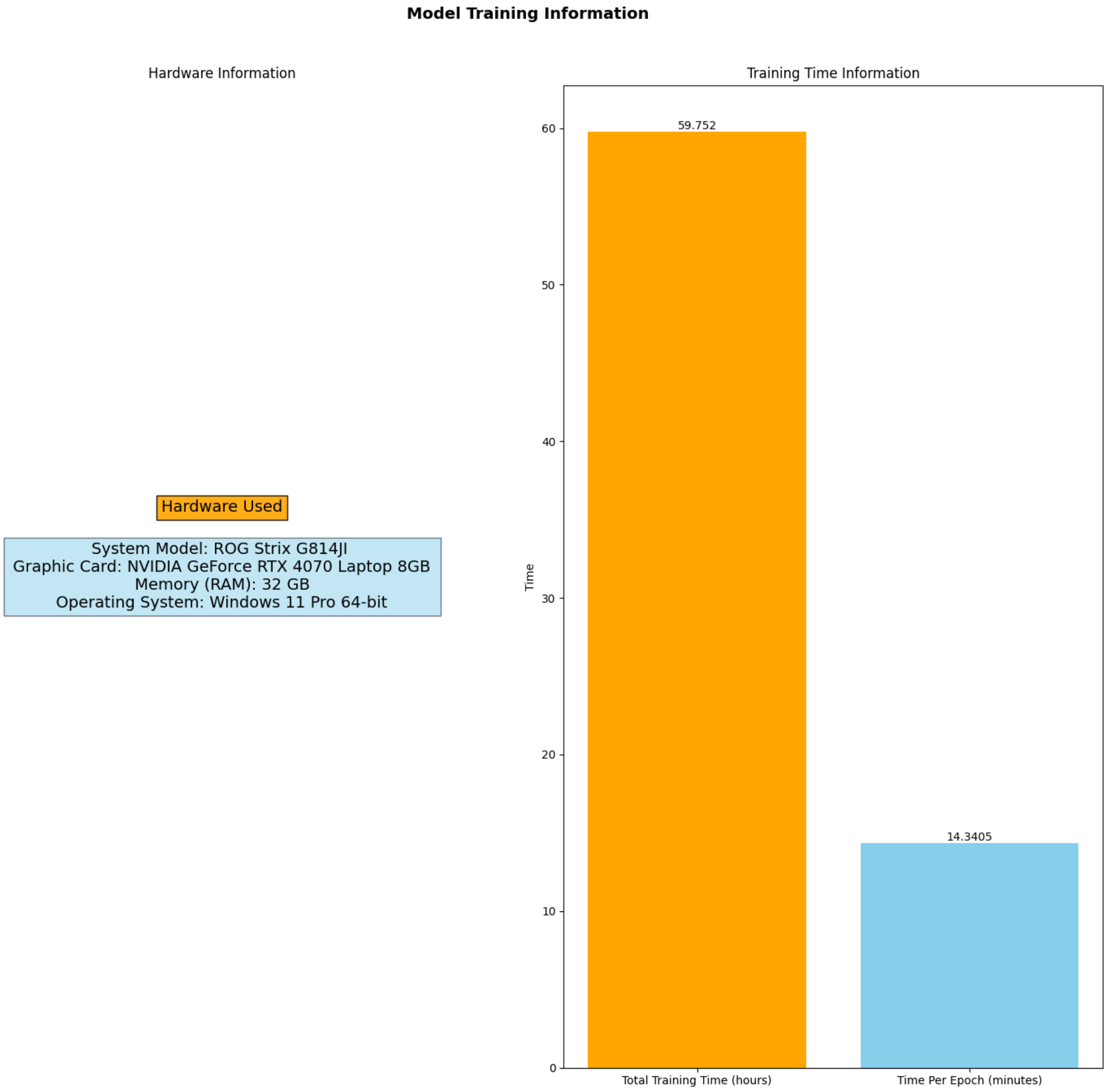


Figure 4.1: Training time with hardware requirement

4.3 Dataset and Preprocessing

This part presents a complete knowledge of the data preparation process as it includes the pretreatment stages as well as the training and validation datasets.

4.3.1 Dataset Description

Two main datasets—the Wider Face Dataset and the Celebrity Face Image Dataset—both obtained from Kaggle helped to train the system. Combining these datasets created a complete training set. These datasets were chosen to show the faces suitable from an acceptable distance. After augmentation, there were a total of 62,938 images after data cleaning. To guarantee consistency across the collection, every picture was cropped to 400x400 pixels. Figure 4.2 shows training raw dataset image instances.

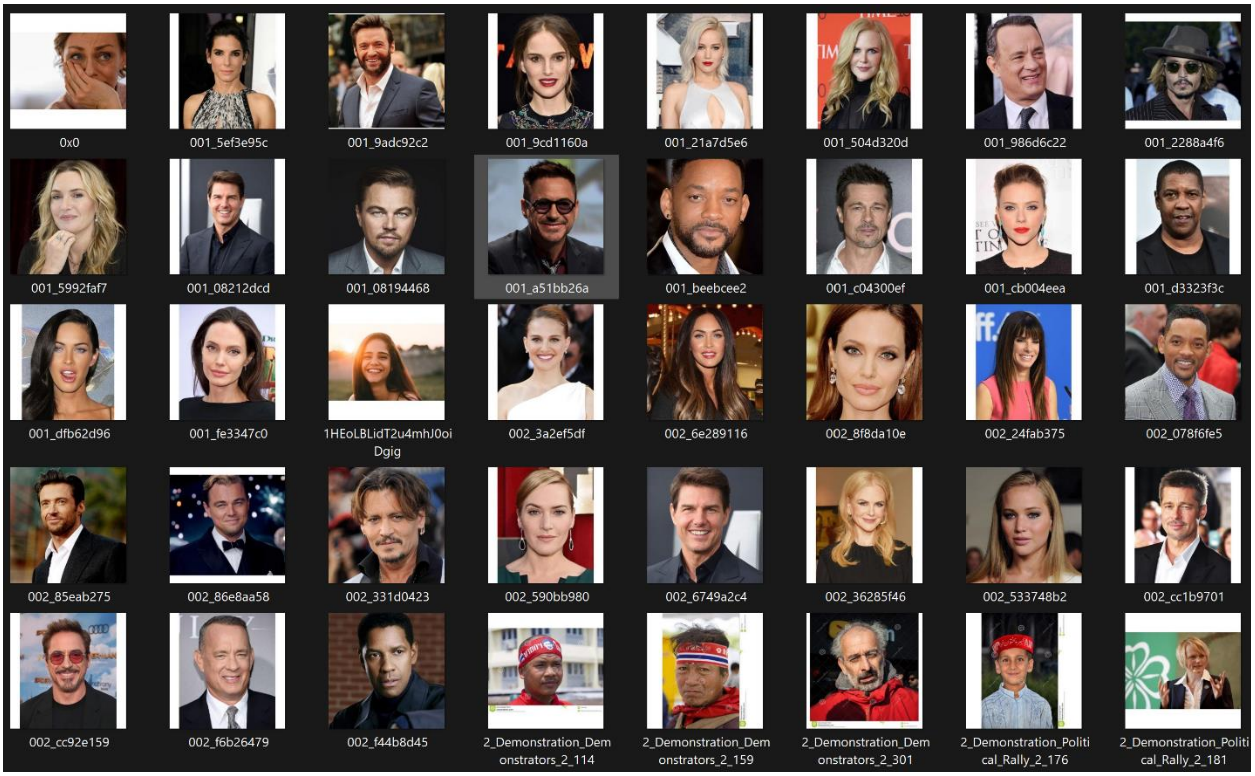


Figure 4.2: Collection images from both datasets

4.3.2 Data Augmentation

Many data augmentation methods were applied to the original images to strengthen the dataset and enhance the model's robustness. These techniques simulate a wide range of scenarios the model may encounter in practical applications, allowing it to generalize better across different contexts.

The applied augmentation methods included horizontal flipping, rotation, brightness adjustment, translation, noise addition, shear, blur, color jitter, and random erasing. Starting with 9,333 images, these augmentations expanded the dataset to 83,516 images before data cleaning. This diversity exposed the model to various situations during training, significantly improving its performance on unseen data. Figure 4.3 displays post-augmentation samples, illustrating the extensive variety these methods provide.

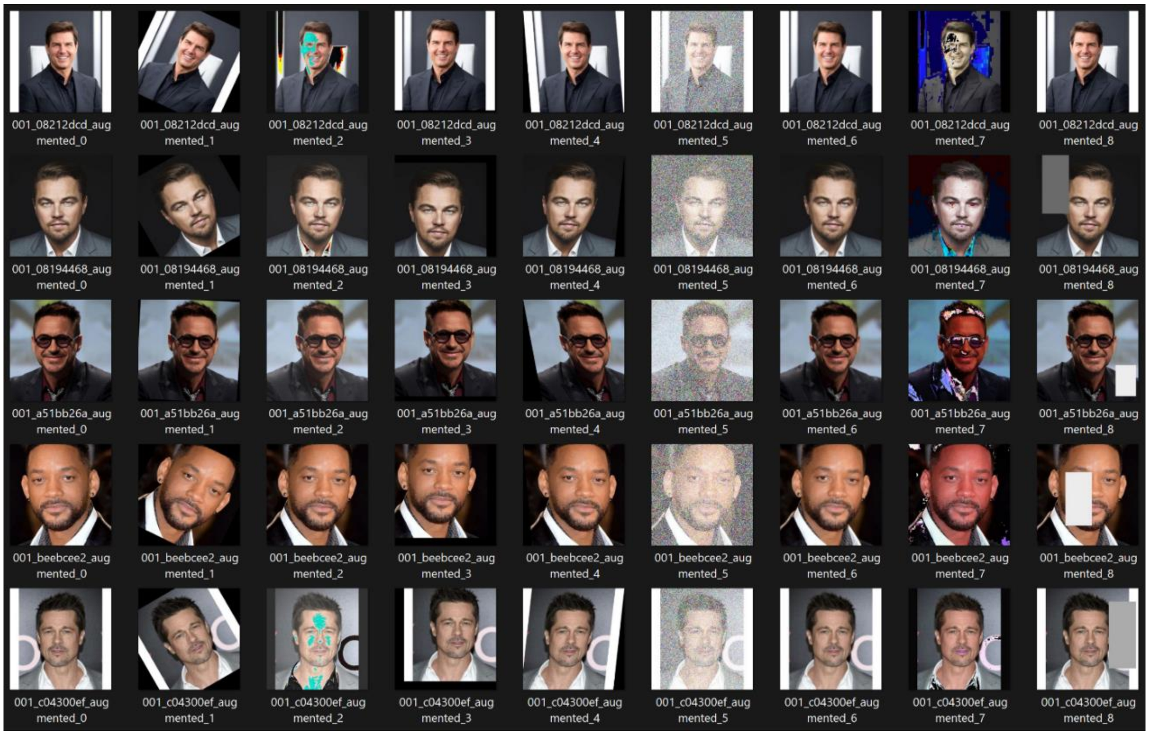


Figure 4.3: The images after applying pre-processing methods

4.4 Landmark Extraction

Extensive identification and monitoring of important facial features like the eyes, eyebrows, lips, and the area around the face depends on the essential preprocessing phase of facial landmark extraction. These landmarks are essential for the system to identify and analyze face movements, therefore allowing precise control of smart home appliances depending on eyebrow motions.

From every shot, the x and y coordinates of 102 sites were derived using a previously mentioned library. Originally the library identified 468 places on the face; for this system simply the 102 most important spots were chosen. Among these chosen sites are the eyes, eyebrows, mouth, and the region surrounding the face. Accordingly, these will help one to recognize and examine eyebrow motions and generate a bounding box around the face. Figure 4.4 shows each 468 landmarks found on the face.

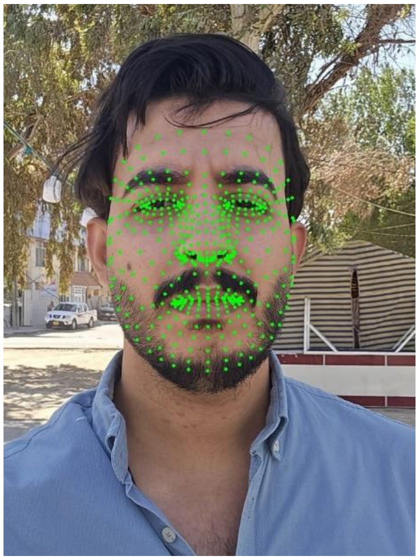


Figure 4.4: MediaPipe facial landmarks

From the complete set of 468 landmarks, a selection process was conducted to identify the 102 most important points. These points especially cover areas surrounding the eyes, eyebrows, lips, and other facial features needed for movement detection.

Finding facial landmarks and then giving exact coordinates for every critical spot form part of the extraction procedure. These coordinates are scaled proportionally to the picture size to normalize them once retrieved so that they match across several photos. A five-pixel margin is inserted around the identified landmarks to create rectangles for face identification, therefore allowing tiny differences in landmark placements and hence improving accuracy. Figure 4.5 lists the particular sites used for this investigation.

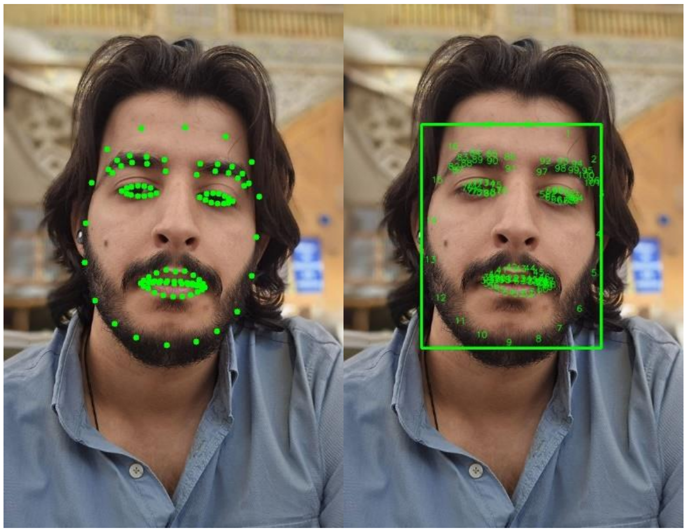
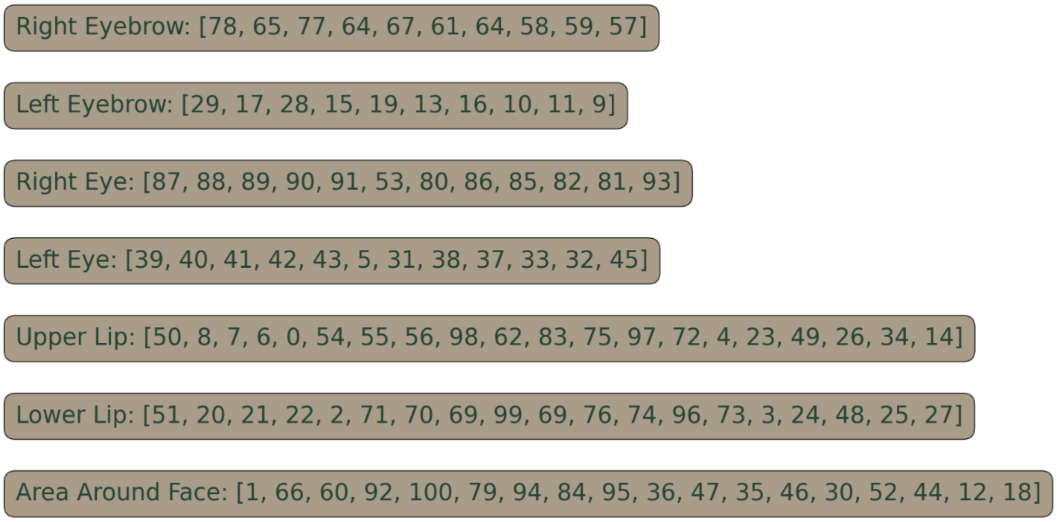


Figure 4.5: Highlighted points of landmarks selected from the 468.

Figure 4.6 shows the landmarks used for this research arranged by facial areas including the eyes, eyebrows, lips, and the area of the face.



Right Eyebrow: [78, 65, 77, 64, 67, 61, 64, 58, 59, 57]

Left Eyebrow: [29, 17, 28, 15, 19, 13, 16, 10, 11, 9]

Right Eye: [87, 88, 89, 90, 91, 53, 80, 86, 85, 82, 81, 93]

Left Eye: [39, 40, 41, 42, 43, 5, 31, 38, 37, 33, 32, 45]

Upper Lip: [50, 8, 7, 6, 0, 54, 55, 56, 98, 62, 83, 75, 97, 72, 4, 23, 49, 26, 34, 14]

Lower Lip: [51, 20, 21, 22, 2, 71, 70, 69, 99, 69, 76, 74, 96, 73, 3, 24, 48, 25, 27]

Area Around Face: [1, 66, 60, 92, 100, 79, 94, 84, 95, 36, 47, 35, 46, 30, 52, 44, 12, 18]

Figure 4.6: Landmarks categorized by facial regions

Before applying augmentation techniques, the dataset was thoroughly checked to ensure all images were processed correctly by the MediaPipe library. This stage was essential to find and fix any photos in which landmarks were absent or poorly sensed. Figures 4.7 illustrate any variations by showing instances of the dataset photos with landmarks added.

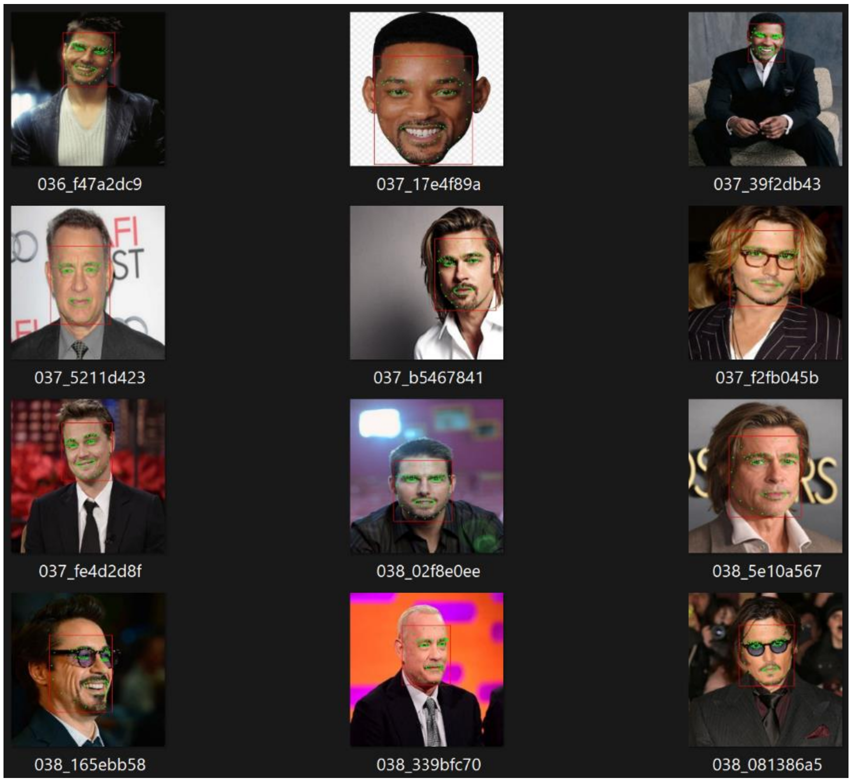
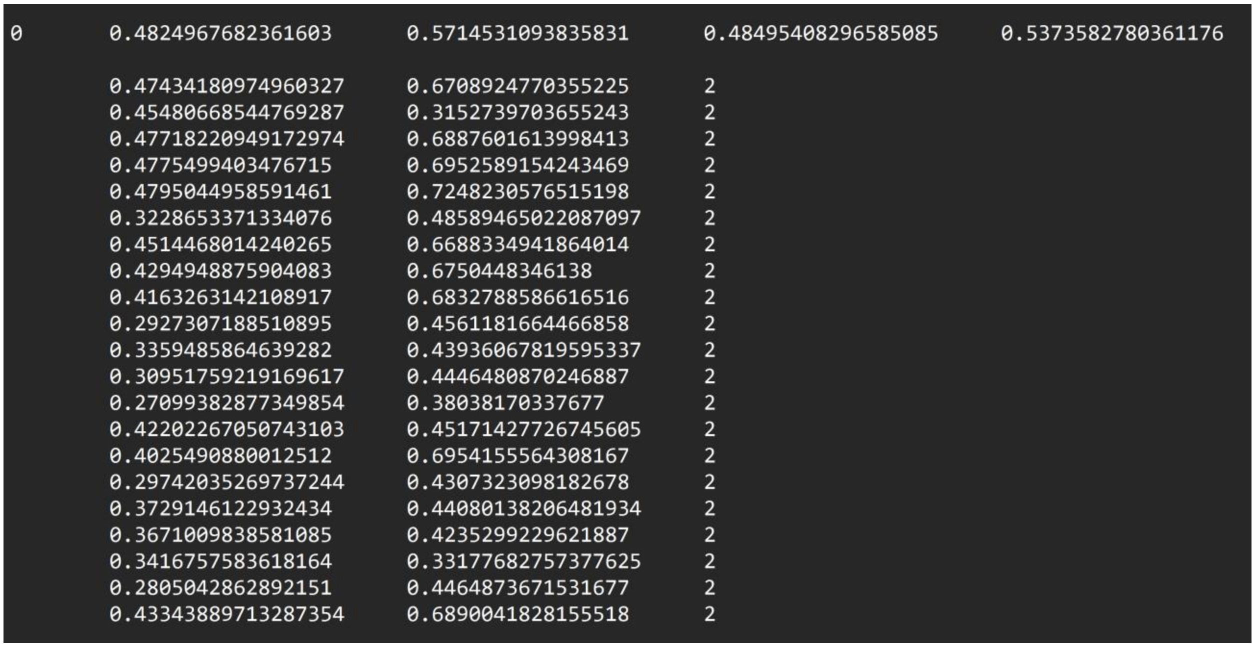


Figure 4.7: Examples of dataset images with applied landmarks before augmentation

Adopted by the YOLOv8n-pose model, the output text file format has various elements:

- The class number comes first.
- The next four values are the coordinates for the bounding box (center_x, center_y, width, height).
- After pairs of coordinates for every landmark, the remaining data consists of a visibility score (x1, y1, 2, x2, y2, 2, ...).

Figure 4.8 shows an instance of this structured text file after normalizing. This arrangement guarantees correct recording of the landmarks and enables the model to utilize them efficiently to identify face motions.



```
0       0.4824967682361603      0.5714531093835831      0.48495408296585085     0.5373582780361176

        0.47434180974960327     0.6708924770355225      2
        0.45480668544769287     0.3152739703655243      2
        0.47718220949172974     0.688760161399841З      2
        0.477549940347671５      0.6952589154243469      2
        0.4795044958591461      0.7248230576515198      2
        0.3228653371334076      0.48589465022087097     2
        0.451446801424026５      0.6688334941864014      2
        0.4294948875904083      0.6750448346138         2
        0.4163263142108917      0.6832788586616516      2
        0.2927307188510895      0.4561181664466858      2
        0.3359485864639282      0.43936067819595337     2
        0.30951759219169617     0.4446480870246887      2
        0.27099382877349854     0.38038170337677        2
        0.42202267050743103     0.45171427726745605     2
        0.4025490880012512      0.6954155564308167      2
        0.29742035269737244     0.4307323098182678      2
        0.3729146122932434      0.44080138206481934     2
        0.3671009838581085      0.4235299229621887      2
        0.3416757583618164      0.33177682757377625     2
        0.2805042862892151      0.4464873671531677      2
        0.43343889713287354     0.6890041828155518      2
```

Figure 4.8: Text file format after normalization

Driven by the requirement for both speed and accuracy, MediaPipe was chosen for first landmark recognition and then input these landmarks into the YOLOv8n-pose model. MediaPipe is faster than YOLO in detecting landmarks but is less accurate. Achieving the best possible accuracy was essential given the sensitivity needed to identify eyebrow movements. Therefore, an ideal solution came from combining the fast detection powers of MediaPipe with the great precision of YOLOv8n-pose

The system's capacity to identify and interpret eyebrow movements—used to operate many smart home appliances— depends on this thorough landmark data.

4.5 YOLOv8n-Pose Model Results

Landmark and face detection performance of the YOLOv8n-pose model under reference to the smart home management system was evaluated. Among the important performance metrics are mAP@0.5, recall, precision, and F1 Score. These tests reveal the actual performance of the model in fields like eyebrow movement detection. Presenting the results in terms of object recognition and posture estimate, we concentrate on the general model performance in all respects.

4.5.1 Training Details

The training process utilized 8 workers to efficiently handle data loading and augmentation. From an overall Eight GB, the GPU memory used during training was around 3.27 GB, therefore ensuring that the model training was both effective and efficient. Having an average of 14.34 minutes per epoch, the 250-epoch total training duration amounted to around 59.252 hours.

Comprising an 82.4% training and 17.6% validation ratio, training and validation sets generated 11,092 photographs and labels for validation; and training, it generated 51,846 images and labels. As stated in Chapter Three, the data augmentation produced an aggregate accumulation of 83,516 images. Following image removal of improperly processed photographs and picture cleansing, the final dataset included 62,938 images along with corresponding labels.

The training ground required the development of a YAML file with class information and dataset locations. Included in the YAML file were details such as the class count, keypoint form, and locations of the validation and training datasets.

4.5.2 Face detection Results

Face detection of the YOLOv8n-pose model showed good performance. The following steps together provide a whole picture of its effectiveness:

4.5.2.1 F1 Score

The F1 Score tests the model by balancing accuracy with recall. Especially for imbalanced datasets, it is very useful as it provides a single statistic spanning erroneous positives and false negatives. With a 0.639 confidence level, the face detection F1 Score came out in this evaluation at 98%. This high F1 Score indicates the model's exceptionally exact object identification performance. Figure 4.9 illustrates the BoxF1 Curve, showcasing the F1 Score across different confidence levels.

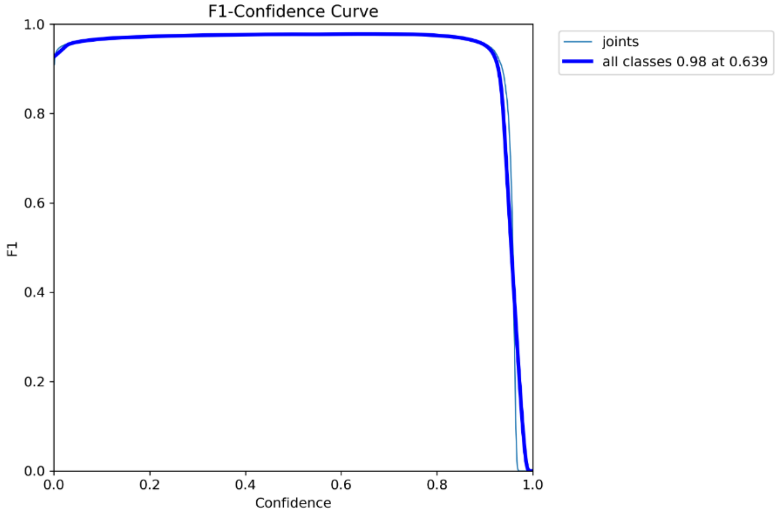


Figure 4.9: BoxF1 Curve

4.5.2.2 Precision

Precision is the proportion of real positive detections among all positive detections. Applications where false alarms might be problematic rely on the model with low false positive rate, Accordingly, high precision was suggested. With a confidence level of 0.988, the face detection accuracy score was 100% indicating the ability of the model to properly identify objects free from erroneous detections. Figure 4.10 presents the Precision-Confidence Curve, highlighting the precision of the model at various confidence thresholds.

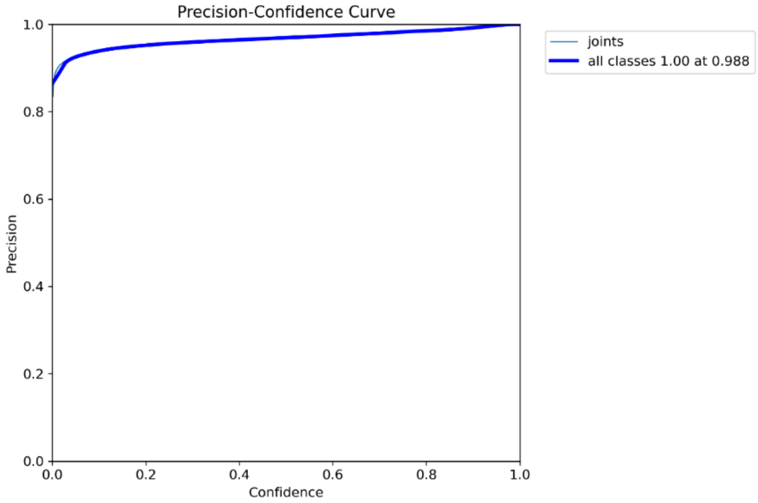


Figure 4.10: Precision-Confidence Curve

4.5.2.3 Recall

Recall measures the proportion of true positive detections among all actual positives. High recall guarantees that most items are found by means of a low false negative rate of the model. At a confidence level of 0.000, the object identification recall score was 100% suggesting that the model effectively identifies all pertinent things inside the pictures. Figure 4.5.2.3 depicts the recall-confidence Curve, therefore displaying the recall performance at several confidence levels.

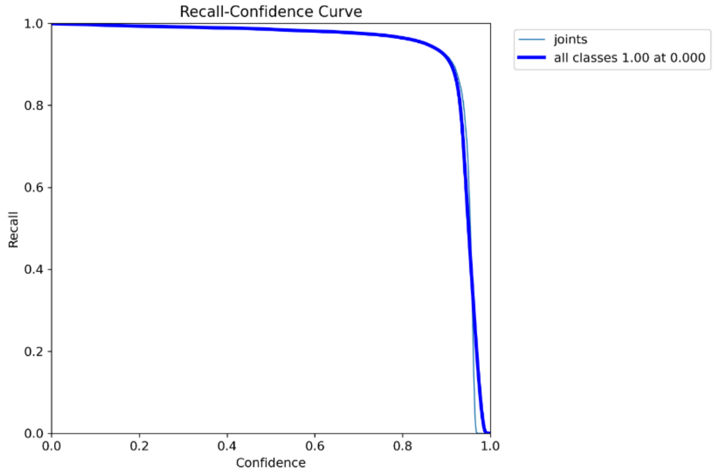


Figure 4.11: Recall-Confidence Curve

4.5.2.4 mAP@0.5

Mean Average accuracy at a 0.5 Intersection over Union (IoU) threshold, mAP@0.5 is a complete measure including both accuracy and recall across many confidence levels. face detection's mAP@0.5 was 99.2%. Balancing accuracy and recall, this high mAP value indicates that the model works very well throughout a variety of detecting situations. Figure 4.12 displays the Precision-Recall Curve, therefore illustrating the trade-off between precision and recall.

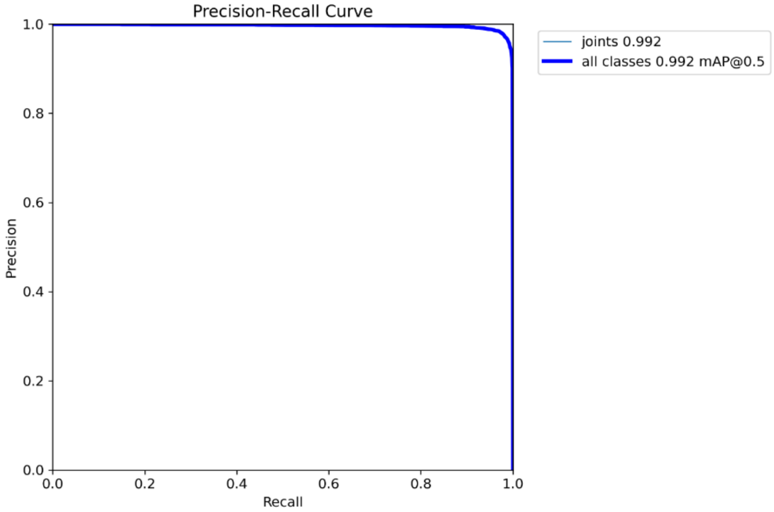


Figure 4.12: Precision-Recall Curve

4.5.2.5 Confusion matrix

The confusion matrix shows the true positives, false positives, true negatives, and false negatives, thereby offering a whole performance of the model analysis. This representation enables one to better grasp the strengths and weaknesses of the model in categorization as well as to identify certain areas where the model needs refinement. The confusion matrix is shown in Figure 4.13.

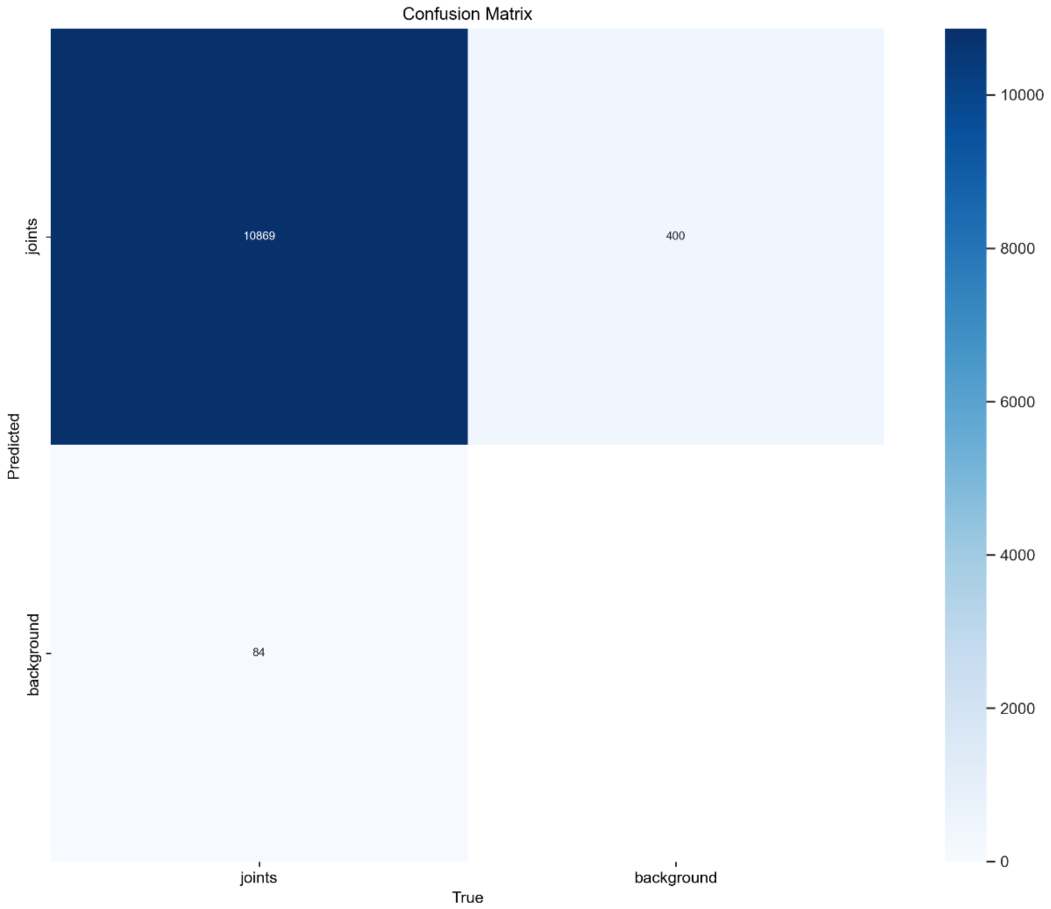


Figure 4.13: Confusion Matrix

4.5.2.6 Normalized Confusion Matrix

It is simpler to understand the relative performance across many classes by use of the normalized confusion matrix, which scales the values in the confusion matrix to a range of 0 to 1. More precisely, this representation reveals the accuracy and misclassification rates of the model. The Normalized Confusion Matrix shown in Figure 4.14 helps to enable a comparison of performance across multiple classes.

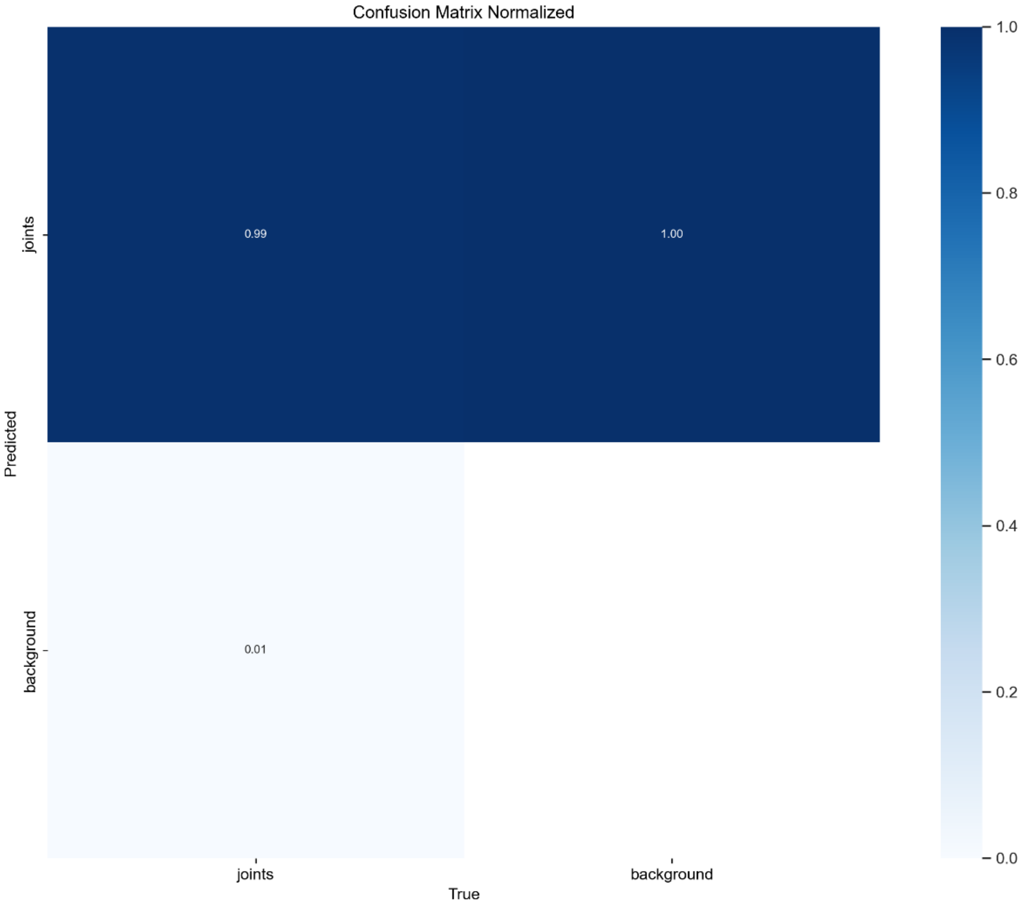


Figure 4.14: Confusion Matrix after Normalization

4.5.3 Pose Estimation Results

YOLOv8n-position model for pose estimation shows remarkable accuracy and reliability in identifying important facial landmarks. These measurements provide a comprehensive evaluation of its performance:

4.5.3.1 F1 Score

s The F1 Score for pose estimation achieved 94% at a confidence level of 0.771. This high F1 Score highlights the capacity of the model to precisely identify and localize facial landmarks. Figure 4.15 shows the PoseF1 Curve.

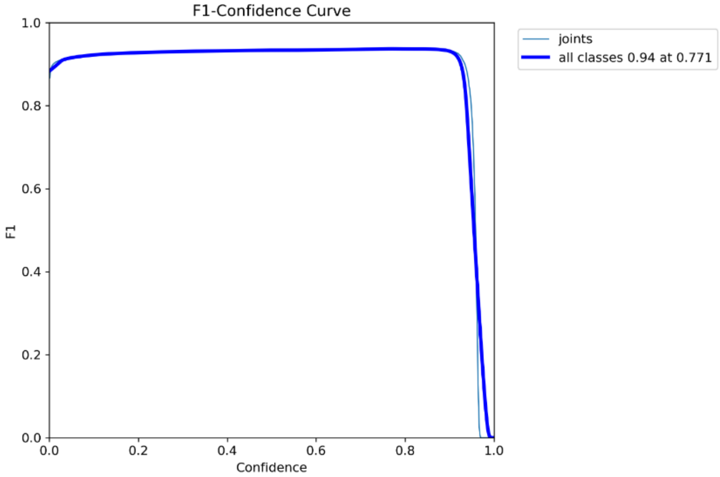


Figure 4.15: PoseF1 Curve

4.5.3.2 Precision

The model attained a precision score of 100% at a confidence level of 0.988, highlighting its effectiveness in accurately identifying landmarks without making incorrect detections. Figure 4.16 presents the Precision-Confidence Curve.

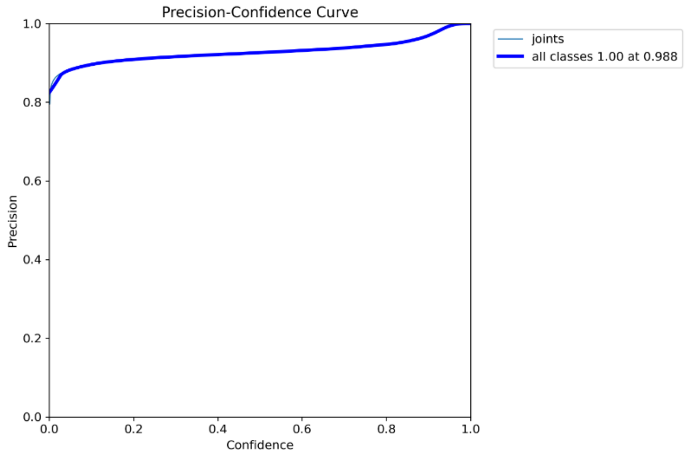


Figure 4.16: Precision-Confidence Curve of Pose estimation score

4.5.3.3 Recall

The recall score for pose estimation was 95% at a confidence level of 0.000, indicating the model's success in detecting all relevant landmarks within the images. Figure 4.17 illustrates the Recall-Confidence Curve.

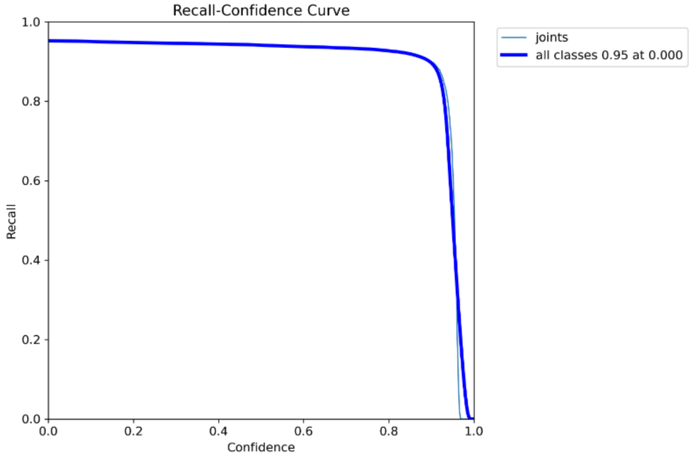


Figure 4.17: Recall-Confidence Curve of Pose estimation score

4.5.3.4 mAP@0.5

The mAP@0.5 for pose estimation was 96.4%. This high mAP value illustrates the remarkable performance of the model in identifying and localizing face landmarks across various settings, so balancing precision and recall. Figure 4.18 shows the Precision-Recall Curve, so illustrating the interaction between precision and recall.

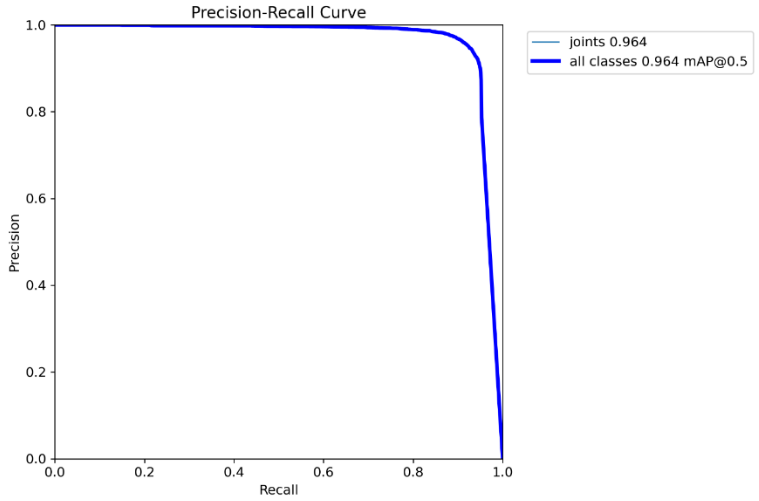


Figure 4.18: Precision-Recall Curve

4.5.3.5 Training and Validation Metrics

Figure 4.19 summarizes across 250 epochs the outcomes of the training and validation measures. This covers precision and mAP50 measures as well as the box loss, pose loss, keypoint object loss, classification loss, and distribution focal loss for both training and validation datasets.

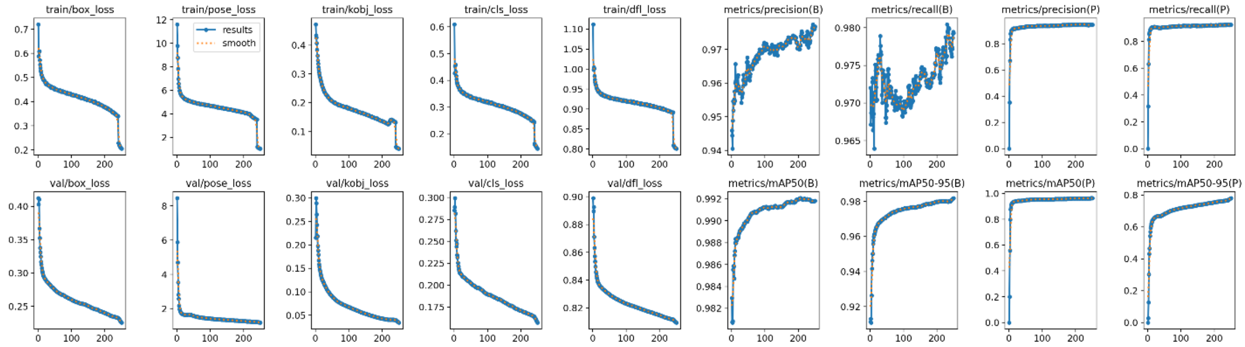


Figure 4.19: Training and Validation measures

4.5.3.6 Train Batch labels

Figure 4.20 shows the training batch images with the detected landmarks. This visual aid clarifies the degree of face feature learning of the model throughout the training period.

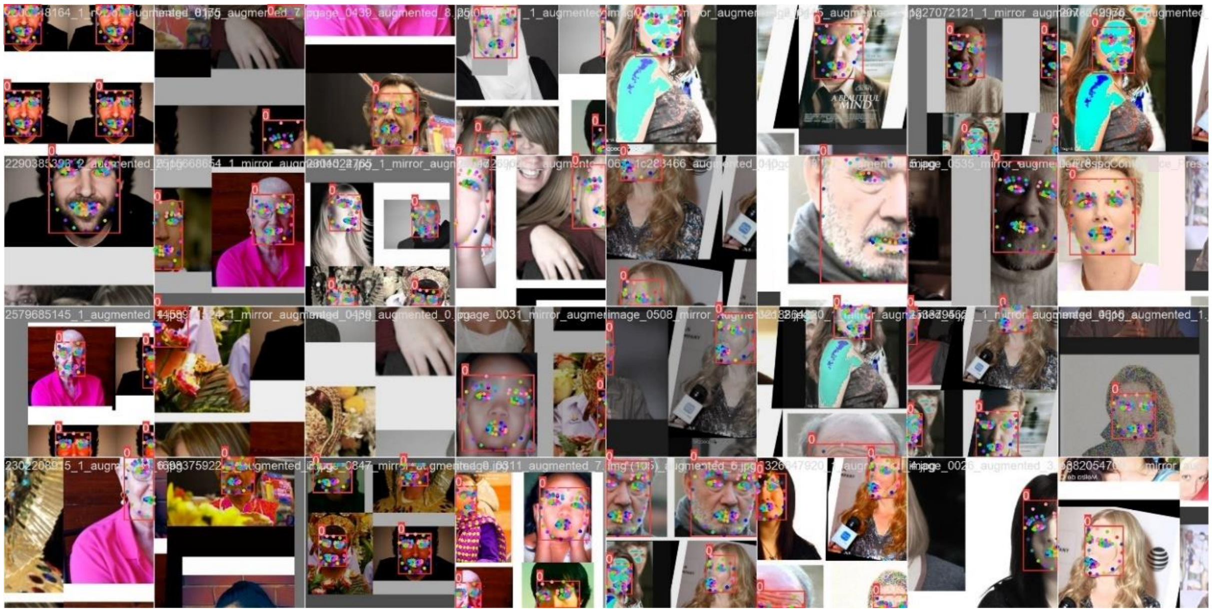


Figure 4.20: Sample of Training Batch Images.

4.5.3.7 Predictions and Validation Batch Labels

Figure 4.21 presents the validation batch images along with the predicted landmarks. This visualization emphasizes the correctness and consistency of the found keypoints, thereby helping one to grasp the performance of the model on the validation set.

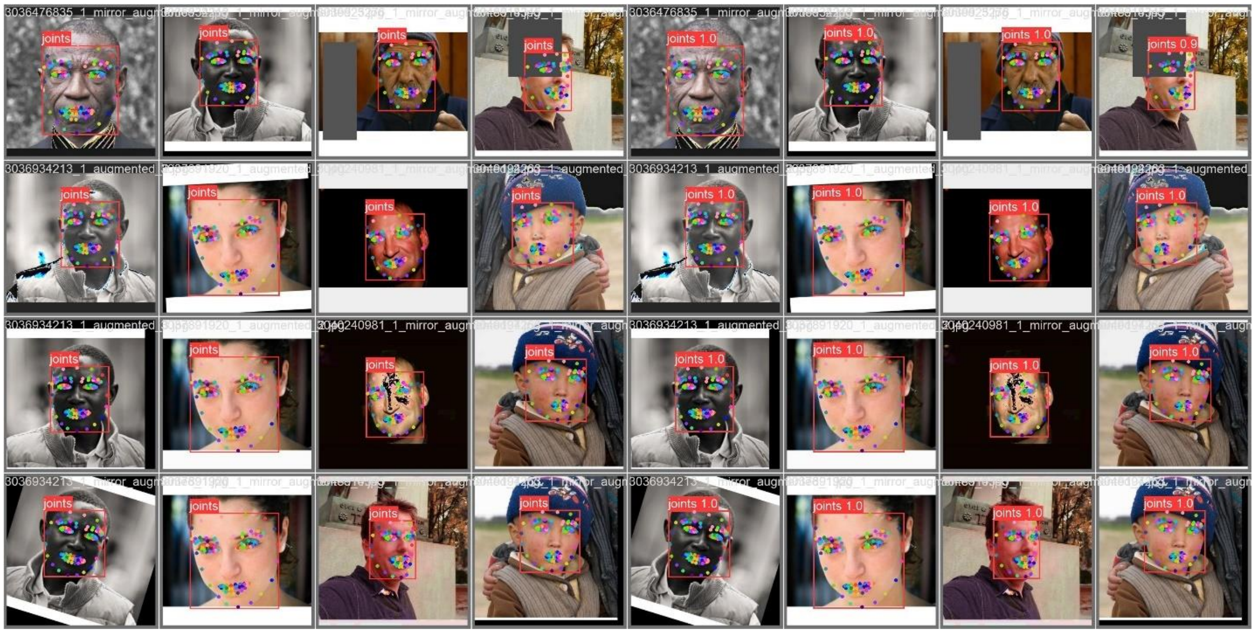


Figure 4.21: Validation Batch Labels and Predictions.

4.6 System Implementation and Integration

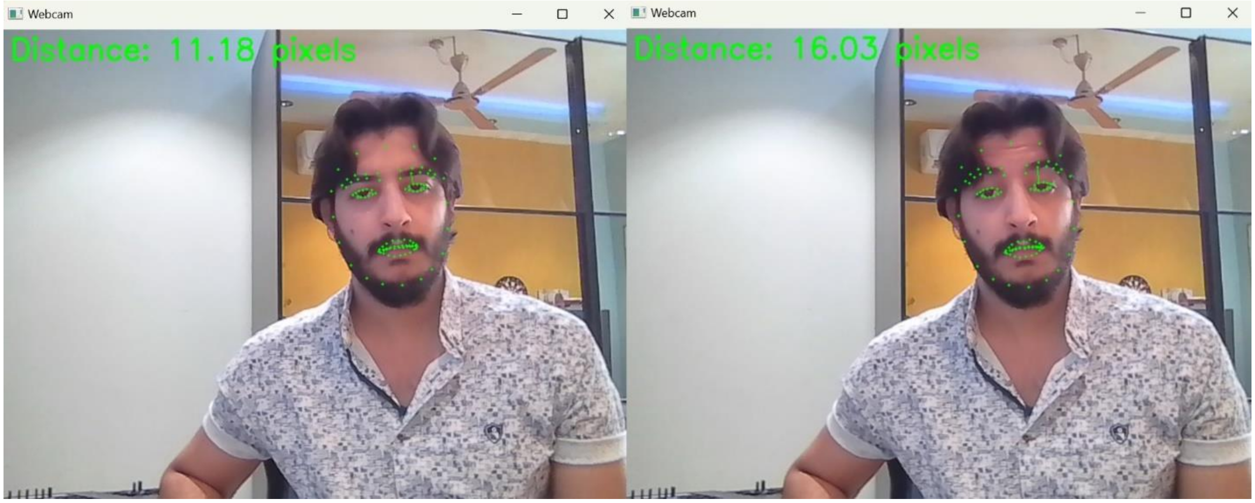
This section describes the complete deployment of the smart home control system including integrating YOLOv8 for real-time situation estimation, using Raspberry Pi for device control, and implementing voice and speech recognition transcription and smart assistant functions for both speaking and non-speaking paralytics.

4.6.1 Real-time YOLOv8 Pose Estimation

The smart home control system uses the YOLOv8-pose model to instantly identify and quantify eyebrow movements. With a camera, the system records live video feed and analyzes every frame to locate face landmarks—more especially, emphasizes the eyes and eyebrows here. To find motions, one measures the distance between a point on the eyebrow and an accompanying point on the eye.

The first shot in Figure 4.23 catches the user still in terms of eyebrows. The technique indicates the important facial features and shows the distance between the chosen eyebrow and eye positions. Here, the distance is noted as 11.18 pixels.

The distance between the dots moves to 16.03 pixels when the user lifts their eyebrow as seen in the second section of Figure 4.22 The algorithm tracks these distances constantly and finds movement when the change above a predetermined threshold—in this example, two pixels. This approach guarantees exact and deliberate eyebrow movement recognition, which is then used to regulate many smart home appliances.



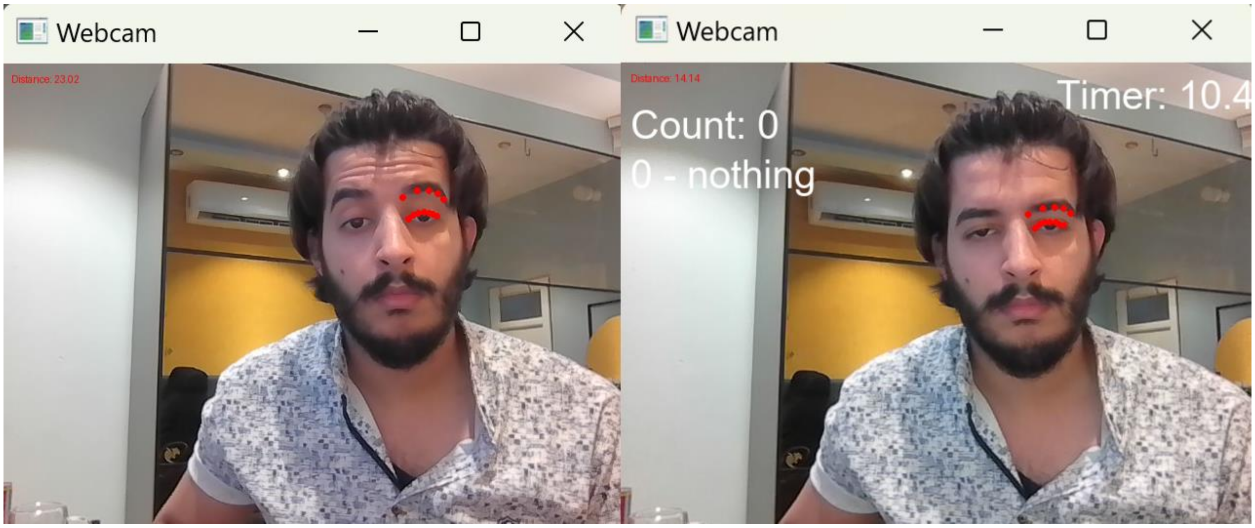
Distance: 11.18 pixels

Distance: 16.03 pixels

Figure 4.22: Real-time YOLOv8 Pose Estimation before and after raising the eyebrow.

One should be aware that the pixel measurements depend on the distance separating the camera from the user. This mechanism kept the user's and the camera's distance at about 90 cm constant. This distance was used to guarantee precise capturing of the found motions within the 2-pixel threshold. Reducing the distance between the camera and the user would need pixel threshold recalibration to maintain detection accuracy.

The system detects a three-second eyebrow raise and moves into a state of listening for certain orders. The system shows that it is ready with a count of "0" and a timer as shown in Figure 4.23.

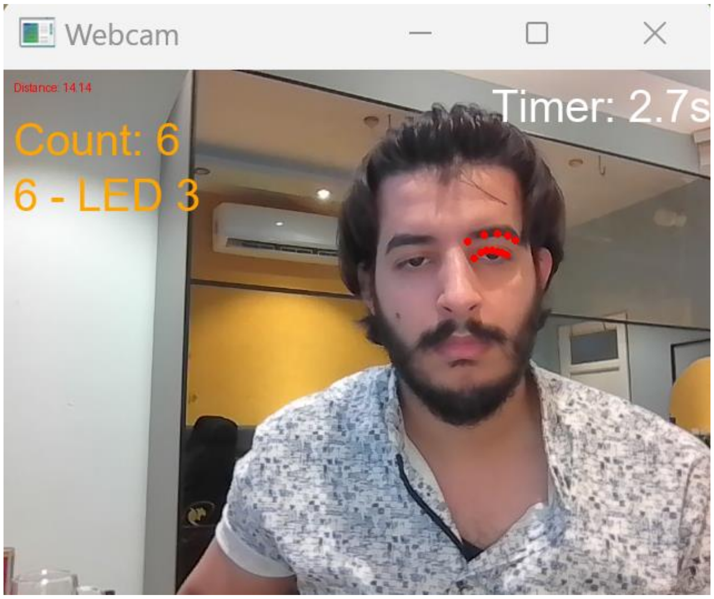


Distance: 23.02

Distance: 14.14
Count: 0
0 - nothing
Timer: 10.4

Figure 4.23: After raising an eyebrow for three seconds listening mode window will pop up.

The user may choose a command within eight seconds by once again raising their eyebrow. On a Raspberry Pi 4, the instructions are coupled to a Relay Module that regulates many smart home devices. The user chooses instruction "six," which translates in Figure 4.24 into turning on LED 3.



Distance: 14.14
Count: 6
6 - LED 3
Timer: 2.7s

Figure 4.24: After raising the eyebrow six times the LED 3 turned on.

4.6.2 Smart Home Control System Implementation

The YOLOv8-Pose model detects eyebrow movements in real-time, therefore enabling control via the smart home management system. The system then instructs a Raspberry Pi 4, which uses a Relay Module to run several devices. The LED pins of the Raspberry Pi and their matching devices are detailed in the following Table 4.1:

Table 4.1: List of Led pins and corresponding devices

Command	LED Pin	Device
1	17	LED 1
2	18	LED 2
3	27	Ceiling Fan
4	22	TV
5	23	Air Conditioner
6	24	LED 3

Figure 4.25 shows the Raspberry Pi 4's configuration, and the Relay Module used to run device control. This integration guarantees the exact execution of the instructions found by the system on the corresponding devices.

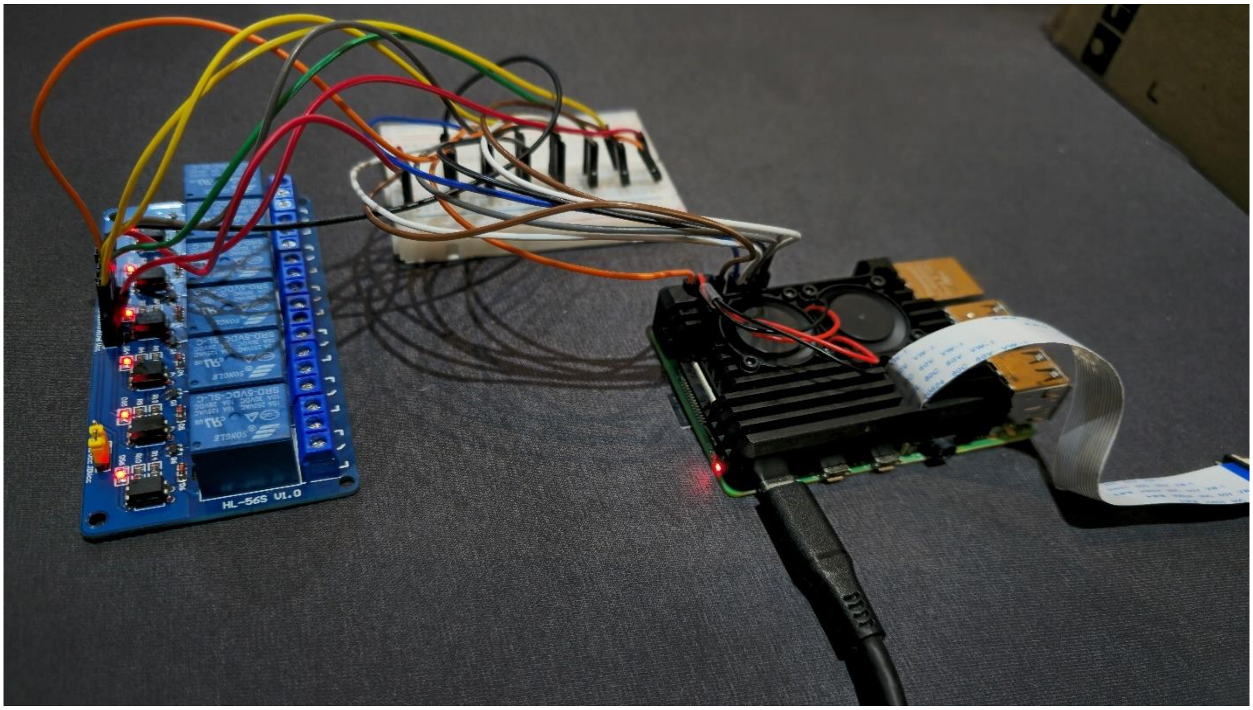


Figure 4.25: Raspberry Pi 4, Relay Module, Breadboard with Wiring, and GPIO Pins Connections are all connected to the house devices.

4.6.3 Integration of artificial intelligence assistants and speech processing

The smart assistant integration and speech processing features help the smart home management system become quite accessible. By combining the Telegram bot with voice-to-text transcription and voice command management, this section shows how the system delivers messages and controls the home through voice commands.

The system converts the user's voice commands into text using the Whisper model of speech-to-text (STT) functionality. This allows for the spoken words to be accurately transcribed, thus enabling effective communication with the Telegram bot and AI assistant. Of the many model sizes offered by Whisper – Small, Basic, Small, Medium, Large, V2, and Large V3 – each has its own characteristics and uses.

The medium model was chosen based on the hardware capability of the system. Real-time applications will find the medium model suitable because it offers a fair balance between speed and accuracy. In particular, the medium model can capture 15 seconds of audio in about 3 seconds, which is a suitable delay for the intended use case. Although the small model is also a decent option, the medium size was chosen to ensure higher transcription accuracy. 100 voice commands were tested and 95 of them were correctly responded to, with a 95% accuracy rate.

When the user says, "Smart Solana" the device uses a microphone to capture the user's voice over a five-second period to determine what is being played from smart home devices, send an automatic message via Telegram, or talk to the smart assistant. Table 4.2 shows the voice commands used.

Table 4.2: List of voice commands

Command	Explain
Turn the right light (on or off).	Control first light device in the room
Turn the left light (on or off).	Control second light device in the room
Turn the Ceiling Fan (on or off).	Control the ceiling fan device in the room
Turn the TV (on or off).	Control the TV device in the room
Turn the air conditioner (on or off).	Control the Air Conditioner device in the room
Turn the center light (on or off).	Control third light device in the room
Send message to (the user).	Send message via telegram to specific user
Hello Solana.	Start chatting with AI assistant voice-to-voice
Bye-bye.	To exit the AI assistant chat voice-to-voice

4.7 Comparison with Other Works

The performance and approaches of the created smart home control system employing eyebrow motions are evaluated in this part in relation to other already published systems. The comparison mostly addresses application areas, technology employed, and accuracy. The aim is to place the present work in the larger field and underline its advances over earlier methods.

4.7.1 Accuracy Comparison

Evaluating the success of different systems depends much on their degree of precision. Table 4.3 lists the stated system accuracy, therefore stressing the performance of the created system.

Table 4.3: Accuracy Comparison with Other Works

Papers	Author(s)	Year	Technology Used	Accuracy
[12]	Khattar, S., et al.	2019	Voice Recognition, IoT, AI	-
[13]	Fadillah, N. and A. Ihsan	2020	Voice Recognition, Smart Bed	75%
[14]	Raj, V., A. Chandran, and A.P. RS	2019	Voice Recognition, IoT	-
[15]	Klaib, A.F., et al.	2019	Eye Tracking, Voice Interfaces	93.2%
[16]	Aqel, M.O., A. Alashqar, and A. Badra	2020	Eye Tracking, Python	-
[17]	Wassan, Z.A., et al.	2021	Eye Blink, IoT	80%
	This Work	2024	Eyebrow Movement, YOLOv8, MediaPipe, Whisper, Coqui	98% (Face Detection), 94% (Pose Estimation), 95% (Voice Command)

The developed system achieves a high accuracy of 98% for face detection, 94% for pose estimation, 95% for voice command surpassing several present approaches in the literature.

CHAPTER FIVE

CONCLUSION AND FUTURE WORK

5.1 Conclusion

By means of strong machine learning models, this work built a smart home control system based on eyebrow movement and speech recognition. high accuracy in system responsiveness and face gesture detection was obtained by several techniques. Based on the results and system performance, the following conclusions have been deducing:

1. Developing an IoT system using YOLOv8 and MediaPipe for paralyzed people allowed the system to observe eyebrow movement and speech recognition.

2. Integrating effective machine learning models guaranteed real-time responsiveness, therefore allowing the system to be suitable for pragmatic purposes in smart home environments.

3. Increasing the generalizing power of the model, data augmentation techniques helped to increase performance on hitherto unmet data.

4. Adding Whisper for speech-to-text and Coqui xtts-v2 provides consistent and powerful voice interaction features, therefore adding a useful layer of accessibility.

5. Using Raspberry Pi and relay modules, the system essentially interfaced with numerous smart home devices, therefore the system was designed to be quite user-friendly and easily accessible so that those with major physical restrictions may run it with little effort.

6. Providing extensive testing and validation verifying the dependability and efficiency of the system in many real-world scenarios thereby attesting its practical relevance.

7. To create a smart home appliance system that is suitable for the needs of multiple users and with additional capabilities, the modular structure of the system must be expandable and flexible.
8. Combining MediaPipe for facial landmark detection with innovative machine learning models such as YOLOv8 for Face detection yielded a better approach for gesture recognition.
9. By employing basic eyebrow movements makes the technology enables persons with physical impairments more successful communication.

10. Comparing the accuracy and performance with to other similar works, the proposed system shows its possible leading solution in the field of assistive technology.

5.2 Future Works

- Installing the system entirely on embedded devices like the Raspberry Pi is one of the primary future objectives. The Raspberry Pi 4 lacks the processing capability required to manage current system requirements. Future models like the Raspberry Pi 5 or more recent models could enable this, therefore offering a more compact and effective option.

- Increasing the utility of the system will help it to be more compatible with a greater spectrum of smart home gadgets. This implies creating interfaces and communication systems to run several house appliances on their own will.

- Better system usability resulting from integration of powerful natural language processing models will increase voice interaction opportunities. Usually, more nuanced and complicated voice commands made possible by this improve user experience.

- Developing tools allowing users to adapt and personalize the system depending on their particular needs and preferences would help to raise user satisfaction. This addresses various gesture recognition sensitivity levels and customized spoken answers.

REFERENCES

[1] K. Zdravkova, "The potential of artificial intelligence for assistive technology in education," in *Handbook on Intelligent Techniques in the Educational Process: Vol 1 Recent Advances and Case Studies*: Springer, 2022, pp. 61-85.

[2] M. Oudah, A. Al-Naji, and J. Chahl, "Hand gesture recognition based on computer vision: a review of techniques," *journal of Imaging,* vol. 6, no. 8, p. 73, 2020.

[3] F. Noroozi, C. A. Corneanu, D. Kamińska, T. Sapiński, S. Escalera, and G. Anbarjafari, "Survey on emotional body gesture recognition," *IEEE transactions on affective computing,* vol. 12, no. 2, pp. 505-523, 2018.

[4] A. Dhillon and G. K. Verma, "Convolutional neural network: a review of models, methodologies and applications to object detection," *Progress in Artificial Intelligence,* vol. 9, no. 2, pp. 85-112, 2020.

[5] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using YOLO: Challenges, architectural successors, datasets and applications," *multimedia Tools and Applications,* vol. 82, no. 6, pp. 9243-9275, 2023.

[6] A. Goldstein *et al.*, "Deep speech-to-text models capture the neural basis of spontaneous speech in everyday conversations," *bioRxiv,* p. 2023.06. 26.546557, 2023.

[7] E. Casanova *et al.*, "XTTS: a Massively Multilingual Zero-Shot Text-to-Speech Model," *arXiv preprint arXiv:2406.04904,* 2024.

[8] I. Ashraf *et al.*, "Home automation using general purpose household electric appliances with Raspberry Pi and commercial smartphone," *Plos one,* vol. 15, no. 9, p. e0238480, 2020.

[9] K. Masai, "Facial Expression Classification Using Photo-reflective Sensors on Smart Eyewear," PhD Thesis, 2018.

[10] S. Bian *et al.*, "Machine learning-based real-time monitoring system for smart connected worker to improve energy efficiency," *Journal of Manufacturing Systems,* vol. 61, pp. 66-76, 2021.

[11] T. K. Ghazali and N. H. Zakaria, "Security, comfort, healthcare, and energy saving: A review on biometric factors for smart home environment," *Journal of Computers,* vol. 29, no. 1, pp. 189-208, 2018.

[12] P. Shi, X. Zhang, W. Li, and H. Yu, "Improving the robustness and adaptability of sEMG-based pattern recognition using deep domain adaptation," *IEEE journal of biomedical and health informatics,* vol. 26, no. 11, pp. 5450-5460, 2022.

[13] R. H. Venkatnarayan and M. Shahzad, "Gesture recognition using ambient light," *Proceedings of the ACM on Interactive, Mobile, Wearable and Ubiquitous Technologies,* vol. 2, no. 1, pp. 1-28, 2018.

[14] A. Kuznetsova, "Multilingual Multimodal Detection of Humour in Stand-Up Comedy," 2023.

[15] L. Jiang, M. Xia, X. Liu, and F. Bai, "Givs: Fine-grained gesture control for mobile devices in driving environments," *IEEE Access,* vol. 8, pp. 49229-49243, 2020.

[16] I. A. Doush *et al.*, "A survey on accessible context-aware systems," *Technological trends in improved mobility of the visually impaired,* pp. 29-63, 2020.

[17] S. Khattar, A. Sachdeva, R. Kumar, and R. Gupta, "Smart home with virtual assistant using raspberry pi," in *2019 9th International conference on cloud computing, data science & engineering (Confluence)*, 2019: IEEE, pp. 576-579.

[18] N. Fadillah and A. Ihsan, "Smart bed using voice recognition for paralyzed patient," in *IOP Conference Series: Materials Science and Engineering*, 2020, vol. 854, no. 1: IOP Publishing, p. 012045.

[19] V. Raj, A. Chandran, and A. P. RS, "Iot based smart home using multiple language voice commands," in *2019 2nd International Conference on Intelligent Computing, Instrumentation and Control Technologies (ICICICT)*, 2019, vol. 1: IEEE, pp. 1595-1599.

[20] A. F. Klaib, N. O. Alsrehin, W. Y. Melhem, and H. O. Bashtawi, "IoT Smart Home Using Eye Tracking and Voice Interfaces for Elderly and Special Needs People," *J. Commun.,* vol. 14, no. 7, pp. 614-621, 2019.

[21] M. O. Aqel, A. Alashqar, and A. Badra, "Smart home automation system based on eye tracking for quadriplegic users," in *2020 International Conference on Assistive and Rehabilitation Technologies (iCareTech)*, 2020: IEEE, pp. 76-81.

[22] Z. A. Wassan *et al.*, "IoT Based Smart Home for Paralyzed Patients through Eye Blink," *International Journal of Advanced Trends in Computer Science and Engineering, Sindh, Pakistan,* 2021.

[23] S. Panat and R. Kumar, "Introduction to Artificial Intelligence & ML," in *A Guide to Applied Machine Learning for Biologists*: Springer, 2023, pp. 127-146.

[24] B. Lantz, *Machine learning with R: expert techniques for predictive modeling*. Packt publishing ltd, 2019.

[25] Y. Roh, G. Heo, and S. E. Whang, "A survey on data collection for machine learning: a big data-ai integration perspective," *IEEE Transactions on Knowledge and Data Engineering,* vol. 33, no. 4, pp. 1328-1347, 2019.

[26] R.-C. Chen, C. Dewi, S.-W. Huang, and R. E. Caraka, "Selecting critical features for data classification based on machine learning methods," *Journal of Big Data,* vol. 7, no. 1, p. 52, 2020.

[27] R. Yacouby and D. Axman, "Probabilistic extension of precision, recall, and f1 score for more thorough evaluation of classification models," in *Proceedings of the first workshop on evaluation and comparison of NLP systems,* 2020, pp. 79-91.

[28] H. H. Rashidi, N. K. Tran, E. V. Betts, L. P. Howell, and R. Green, "Artificial intelligence and machine learning in pathology: the present landscape of supervised methods," *Academic pathology,* vol. 6, p. 2374289519873088, 2019.

[29] A. Kuznetsova *et al.*, "The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale," *International journal of computer vision,* vol. 128, no. 7, pp. 1956-1981, 2020.

[30] M. Bansal, A. Goyal, and A. Choudhary, "A comparative analysis of K-nearest neighbor, genetic, support vector machine, decision tree, and long short term memory algorithms in machine learning," *Decision Analytics Journal,* vol. 3, p. 100071, 2022.

[31] C. Zhang, S. Bengio, M. Hardt, B. Recht, and O. Vinyals, "Understanding deep learning (still) requires rethinking generalization," *Communications of the ACM,* vol. 64, no. 3, pp. 107-115, 2021.

[32] Z. Zou, K. Chen, Z. Shi, Y. Guo, and J. Ye, "Object detection in 20 years: A survey," *Proceedings of the IEEE,* vol. 111, no. 3, pp. 257-276, 2023.

[33] T. Diwan, G. Anirudh, and J. V. Tembhurne, "Object detection using YOLO: Challenges, architectural successors, datasets and applications," *multimedia Tools and Applications,* vol. 82, no. 6, pp. 9243-9275, 2023.

[34] Y.-H. Hsu, C.-C. Yu, and H.-Y. Cheng, "Enhancing Badminton Game Analysis: An Approach to Shot Refinement via a Fusion of Shuttlecock Tracking and Hit Detection from Monocular Camera," *Sensors,* vol. 24, no. 13, p. 4372, 2024.

[35] C. Santos, M. Aguiar, D. Welfer, and B. Belloni, "A new approach for detecting fundus lesions using image processing and deep neural network architecture based on YOLO model," *Sensors,* vol. 22, no. 17, p. 6441, 2022.

[36] S. Teng, Z. Liu, G. Chen, and L. Cheng, "Concrete crack detection based on well-known feature extractor model and the YOLO_v2 network," *Applied Sciences,* vol. 11, no. 2, p. 813, 2021.

[37] M. Hussain, "YOLO-v1 to YOLO-v8, the rise of YOLO and its complementary nature toward digital manufacturing and industrial defect detection," *Machines,* vol. 11, no. 7, p. 677, 2023.

[38] R.-Y. Ju and W. Cai, "Fracture detection in pediatric wrist trauma X-ray images using YOLOv8 algorithm," *Scientific Reports,* vol. 13, no. 1, p. 20077, 2023.

[39] S. N. Tesema and E.-B. Bourennane, "Resource-and Power-Efficient High-Performance Object Detection Inference Acceleration Using FPGA," *Electronics,* vol. 11, no. 12, p. 1827, 2022.

[40] C. Kumar and R. Punitha, "Yolov3 and yolov4: Multiple object detection for surveillance applications," in *2020 Third international conference on smart systems and inventive technology (ICSSIT)*, 2020: IEEE, pp. 1316-1321.

[41] S. Liang *et al.*, "Edge YOLO: Real-time intelligent object detection system based on edge-cloud cooperation in autonomous vehicles," *IEEE Transactions on Intelligent Transportation Systems,* vol. 23, no. 12, pp. 25345-25360, 2022.

[42] A. Kuzdeuov, D. Taratynova, A. Tleuliyev, and H. A. Varol, "OpenThermalPose: An Open-Source Annotated Thermal Human Pose Dataset and Initial YOLOv8-Pose Baselines," *Authorea Preprints,* 2024.

[43] K. Khabarlak and L. Koriashkina, "Fast facial landmark detection and applications: A survey," *arXiv preprint arXiv:2101.10808,* 2021.

[44] M. Kavitha, S. Saranya, E. Pragatheeswari, S. Kaviyarasu, N. Ragunath, and P. Rahul, "A Real-Time Hand-Gesture Recognition Using Deep Learning Techniques," in *International Conference on Artificial Intelligence and Smart Energy*, 2024: Springer, pp. 489-500.

[45] M. Mukhiddinov, O. Djuraev, F. Akhmedov, A. Mukhamadiyev, and J. Cho, "Masked face emotion recognition based on facial landmarks and deep learning approaches for visually impaired people," *Sensors,* vol. 23, no. 3, p. 1080, 2023.

[46] J. Bora, S. Dehingia, A. Boruah, A. A. Chetia, and D. Gogoi, "Real-time assamese sign language recognition using mediapipe and deep learning," *Procedia Computer Science,* vol. 218, pp. 1384-1393, 2023.

[47] F. Alanazi, G. Ushaw, and G. Morgan, "Improving Detection of DeepFakes through Facial Region Analysis in Images," *Electronics,* vol. 13, no. 1, p. 126, 2023.

[48] M. Jefferson, "Usability of automatic speech recognition systems for individuals with speech disorders: Past, present, future, and a proposed model," 2019.

[49] S. Nagdewani and A. Jain, "A review on methods for speech-to-text and text-to-speech conversion," *International Research Journal of Engineering and Technology (IRJET),* vol. 7, no. 05, 2020.

[50] J. A. Thuestad and Ø. Grutle, "Speech-to-text models to transcribe emergency calls," The University of Bergen, 2023.

[51] C. Graham and N. Roll, "Evaluating OpenAI's Whisper ASR: Performance analysis across diverse accents and speaker traits," *JASA Express Letters,* vol. 4, no. 2, 2024.

[52] Z. Kozhirbayev, "Kazakh Speech Recognition: Wav2vec2. 0 vs. Whisper," *Journal of Advances in Information Technology,* vol. 14, no. 6, pp. 1382-1389, 2023.

[53] Z. B. Akhtar, "From bard to Gemini: An investigative exploration journey through Google’s evolution in conversational AI and generative AI," *Computing and Artificial Intelligence,* vol. 2, no. 1, pp. 1378-1378, 2024.

[54] S. Karthikeyan, R. A. Raj, M. V. Cruz, L. Chen, J. A. Vishal, and V. Rohith, "A systematic analysis on raspberry pi prototyping: Uses, challenges, benefits, and drawbacks," *IEEE Internet of Things Journal,* vol. 10, no. 16, pp. 14397-14417, 2023.

[55] K. Kitawat, P. Kumari, P. Sapparapu, and S. N. Shoudha, "Raspberry Pi based IoT Device and User Management Architecture."

[56] S. Monk, *Raspberry pi cookbook*. " O'Reilly Media, Inc.", 2022.

[57] M. Maksimović, V. Vujović, N. Davidović, V. Milošević, and B. Perišić, "Raspberry Pi as Internet of things hardware: performances and constraints," *design issues,* vol. 3, no. 8, pp. 1-6, 2014.

[58] G. F. Avisyah, I. J. Putra, and S. S. Hidayat, "Open Artificial Intelligence Analysis using ChatGPT Integrated with Telegram Bot," *Jurnal ELTIKOM,* vol. 7, no. 1, pp. 60-66, 2023.

[59] V. Kobets and S. Savchenko, "Using telegram bots for personalized financial advice for staff of manufacturing engineering enterprises," in *Design, Simulation, Manufacturing: The Innovation Exchange*: Springer, 2022, pp. 561-571.

الخـــلاصـــة

تقدم هذه الأطروحة نظامًا متقدمًا قائمًا على إنترنت الأشياء مصممًا للتحكم في أجهزة المنزل الذكية، تم تطويره خصيصًا لتلبية احتياجات الأشخاص المصابين بالشلل أو الإعاقة. ويتناول الدور الحاسم لإنترنت الأشياء في تسهيل الحياة المستقلة للأفراد ذوي الإعاقات الجسدية من خلال تمكين تطبيقات المنزل الذكي التي تستجيب لمدخلات المستخدم المختلفة. يتم النظر في نوعين من المستخدمين المشلولين: أولئك الذين لا يستطيعون الكلام، باستخدام حركات الحاجب للتحكم في المنزل الذكي، وأولئك الذين يمكنهم التحدث، باستخدام الأوامر الصوتية المحولة إلى نص للتحكم في الأجهزة أو إرسال الرسائل.

يشتمل النظام على نموذجYOLO ، الذي تم تدريبه خصيصًا لاكتشاف معالم الوجه بدقة عالية من خلال معالجة ٩٣٣٣ صورة من مجموعتي بيانات الصور، Wider Face وCelebrity Face. بعد تطبيق التكبير وتنظيف البيانات، تم تحويلها إلى ٦٢٩٣٨ صورة. باستخدام مكتبةMediaPipe ، تم استخراج ١٠٢ معلم وجه، تغطي مناطق مثل الحاجبين والعينين والشفتين العلوية والسفلية ومعالم الوجه الأخرى. تم وضع علامة على كل صورة من مجموعة البيانات واستخدامها لتدريب نموذج.YOLOv8n-pose

تتضمن الميزات الرئيسية للنظام القدرة على التحكم في الأجهزة المنزلية - مثل الأضواء وأجهزة التلفزيون والمراوح ومكيفات الهواء - من خلال حركات الحاجب البسيطة. بالنسبة للمستخدمين الذين يمكنهم التحدث، تم استخدام مكتبة Whisper لتحويل الصوت إلى نص، مما يتيح التحكم في الجهاز عبر الأوامر الصوتية. يخدم نموذج واجهة برمجة تطبيقات Gemini وظيفتين: اكتشاف ما إذا كان المستخدم مستيقظًا أم نائمًا والانخراط في محادثة مع مساعد الذكاء الاصطناعي المخصص. يساعد الأول المستخدمين المتحدثين، بينما تم تصميم الأخير للمستخدمين غير المتحدثين.

يوضح النموذج مقاييس أداء استثنائية، حيث حقق دقة اكتشاف الوجه بنسبة ٩٨٪، وتقدير الوضع لـ ١٠٢ معلم بدقة ٩٤٪، ودقة الأوامر الصوتية بنسبة ٩٥٪.

جامعة كربلاء
كلية علوم الحاسب وتقنية المعلومات
قسم علوم الحاسوب

تطوير نظام إنترنت الأشياء للمرضى المصابين بالشلل باستخدام تقنيات تقييم حركة الحاجب والصوت

رسالة ماجستير

مقدمة إلى مجلس كلية علوم الحاسوب وتقنية المعلومات / جامعة كربلاء استكمالاً لمتطلبات درجة الماجستير في علوم الحاسوب

كتبت بواسطة

علي ثابت جبار

بأشراف

أ.د. بهيجة خضير شكر

أ.د. نضال خضير العبادي

١٤٤٦هـ ٢٠٢٤م