# INNOVATIVE IOT SOLUTION FOR CONTACTLESS ATTENDANCE WITH TEMPERATURE DETECTION

**A PROJECT REPORT**

***Submitted by***

| | |
|---|---|
| **ADITHYA S S** | **(910619104004)** |
| **HARSHAN R S** | **(910619104027)** |
| **JEYA PRATHAP P** | **(910619104035)** |

***in partial fulfillment for the award of the degree***
***of***

**BACHELOR OF ENGINEERING**

**in**

**COMPUTER SCIENCE AND ENGINEERING**

**K.L.N. COLLEGE OF ENGINEERING, POTTAPALAYAM**
(An Autonomous Institution, Affiliated to Anna University, Chennai)

**ANNA UNIVERSITY: CHENNAI 600 025**

**APRIL 2023**

## ANNA UNIVERSITY: CHENNAI 600 025

## BONAFIDE CERTIFICATE

Certified that this project report **"INNOVATIVE IOT SOLUTION FOR CONTACTLESS ATTENDANCE WITH TEMPERATURE DETECTION"** is the bonafide work of "**ADITHYA S S (910619104004), HARSHAN R S (910619104027), JEYA PRATHAP P (910619104035)"** who carried out the project work under my supervision.

| | |
|---|---|
| **SIGNATURE** | **SIGNATURE** |
| Dr.S.MIRUNA JOE AMALI | Dr.R.KARTHICK |
| **HEAD OF THE DEPARTMENT** | **SUPERVISOR** |
| | ASSOCIATE PROFESSOR |
| Computer Science and Engineering | Computer Science and Engineering |
| K.L.N. College of Engineering, | K.L.N. College of Engineering, |
| Pottapalayam, | Pottapalayam, |
| Sivagangai-630 612. | Sivagangai-630 612. |

Submitted for the project viva-voce conducted on ______________.

**INTERNAL EXAMINER** **EXTERNAL EXAMINER**

# ACKNOWLEDGEMENT

The satisfaction and euphoria that accompany the successful completion of any task would be but incomplete without the mention of the people made it possible, whose constant guidance and encouragement crowned our efforts with success.

We extend our gratitude to the Founder, **Late. Thiru. K.L.N. KRISHNAN, K.L.N. College of Engineering and Management Members** for making us march towards the glory of success. We express our sincere thanks to our respected Principal **Dr .A.V. RAM PRASAD**, **M.E, Ph.D., MISTE,FIE,** for all the facilities offered.

We would like to express our deep gratitude and heartfelt thanks to **Dr.S.MIRUNA JOE AMALI, M.E, Ph.D.,** Professor and Head of the Department of Computer Science and Engineering, **K.L.N. College of Engineering,** who motivated and encouraged us to do this out rival project for this academic year.

Our profound, delightful and sincere thanks to our Project guide **Dr.R.KARTHICK, M.E.,Ph.D.,** and our respected Project Coordinator **Mrs.V.KARPAGAM, M.E.,** whose support was inevitable during the entire period of our work.

We thank our teaching staffs for sharing their knowledge and view to enhance our project. We also thank our non-teaching staff for extending their technical support to us. I thank my parents for giving me such a wonderful life and my friends for their friendly encouragement throughout the project.

Finally, we thank the Almighty for giving the full health to finish the Project successfully.

## ABSTRACT

Face recognition technology has become more widely used recently, particularly in attendance systems. Nevertheless, the conventional approach to taking attendance takes a long time and is prone to mistakes. In this research, an IoT Arduino-based facial recognition attendance system with temperature detection is proposed. The system consists of a camera, an IoT Arduino microcontroller, and a temperature sensor. When people enter the building, the camera catches their faces, which the face recognition algorithm then analyses. To identify the person, the system matches the captured face with the database. Using a temperature sensor, the device also determines the user's body temperature. The technology warns the relevant authorities if the temperature rises beyond the threshold value but does not record attendance. To automate the method of recording attendance, which cuts down on the time and work needed for human attendance. The adoption of IoT Arduino allows for remote system monitoring and management. To give additional functionality, the system can be coupled with other IoT devices such as automatic door locking and unlocking. To create an attendance system that can be used in a variety of companies and is dependable, effective, and secure. The facial recognition-based attendance system with temperature detection based on IoT Arduino has various benefits, including enhanced accuracy, decreased errors, and real-time monitoring. In addition, the technology handles the present COVID-19 problem by detecting body temperature of people entering the building and halting the transmission of the virus.

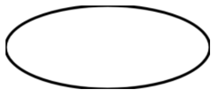
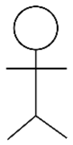
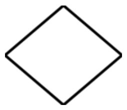
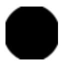
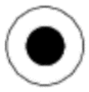
# TABLE OF CONTENTS

# LIST OF FIGURES

# LIST OF ABBREVIATIONS

| ABBREVIATIONS | EXPANSION |
|---|---|
| NLP | Natural Language Processing |
| CNN | Convolutional Neural Network |
| LBPH | Local Binary Patterns Histograms |
| IR | Infra-Red |
| HOG | Histogram of Oriented Gradients |
| COVID -19 | Corona Virus Disease - 2019 |
| CSV | Comma - Separated Values |
| ADC | Analog - to - Digital Converter |
| UML | Unified Modeling Language |

## LIST OF SYMBOLS

| SYMBOL | SYMBOL NAME |
|---|---|
|  | Use Case |
|  | Actor |
|  | Decision |
|  | Start |
|  | Stop |

# CHAPTER 1
# INTRODUCTION

## 1.1 Machine Learning

Machine learning is a type of artificial intelligence that allows computer systems to automatically improve their performance on a task through experience. It involves the use of algorithms and statistical models to analyze and identify patterns in data, and then use those patterns to make predictions or decisions without being explicitly programmed. Machine learning can be categorized into three main types, supervised learning, unsupervised learning, and reinforcement learning.

Supervised learning involves training a machine learning model on labeled data, where the model learns to predict an output variable based on a set of input variables. Unsupervised learning, on the other hand, involves training a model on unlabeled data, where the model must identify patterns or structure within the data without any specific guidance. Reinforcement learning involves training a model through a trial-and-error process, where the model receives feedback in the form of rewards or penalties for each decision it makes.

Machine learning has numerous applications across many fields, including computer vision, natural language processing, speech recognition, fraud detection, recommendation systems, and autonomous vehicles, among others.

## 1.2 Deep Learning

Deep learning is a subset of machine learning that involves the use of neural networks with multiple layers. These networks are designed to simulate the way the human brain works, by using layers of interconnected nodes or "neurons" to process and analyze data.

In a deep neural network, each layer extracts features or patterns from the data and passes that information to the next layer, which then extracts even more complex features. The output of the final layer can be used to make predictions or decisions based on the input data.

Deep learning has been highly successful in a variety of applications, including image and speech recognition, natural language processing, and computer vision. One of the key advantages of deep learning is that it can automatically learn and extract relevant features from raw data, without the need for manual feature engineering, making it highly effective for processing unstructured data such as images and text.

## 1.3 Convolutional Neural Network

A CNN or Convolutional Neural Network, is a type of deep neural network commonly used in image and video processing applications. It is designed to automatically learn and extract features from images through a process called convolution.

The key feature of a CNN is its use of convolutional layers, which consist of a set offilters or "kernels" that slide across an input image, performing mathematical operations ateach position to produce a feature map. The resulting feature map highlights patterns or features of the image that are relevant to the given task, such as edges, shapes, or textures.
In addition to convolutional layers, a typical CNN also includes pooling layers, which reduce the spatial dimensions of the feature map by down sampling or subsampling the data, and fully connected layers, which perform the final classification or regression based on the extracted features.

CNNs have achieved state-of-the-art performance in a variety of computer vision tasks, including image classification, object detection, and segmentation, and are widely used in industry and academia for a wide range of applications such as autonomous vehicles, facial recognition, and medical imaging.

### 1.3.1 Object Detection

Object detection is an important aspect of face recognition technology as it helps in identifying and locating objects of interest within an image or video frame. In the context of face recognition, object detection is used to identify and locate faces within an image or video stream.

Objection detection for face recognition involves the use of various algorithms and techniques to detect and locate faces in an image or video stream. These algorithms use machine learning and deep learning techniques to analyse the image or video frame and identify the location of the face. The most commonly used algorithms for object detection in face recognition include Haar cascades, Histogram of Oriented Gradients (HOG), and Convolutional Neural Networks (CNN).

Haar cascades use a series of simple classifiers to detect the presence of a face within an image or video frame. HOG algorithms use gradient orientation information to detect the edges of objects within an image, including faces. CNNs use a deep neural network to learnand classify the features of faces, allowing for more accurate and reliable detection.

Overall, objection detection is a crucial component of face recognition technology as it allows for accurate and reliable identification and authentication of individuals in a variety of settings, including security and surveillance, identity verification, and biometric authentication systems.

### 1.3.2 Face Recognition

Face recognition is a computer vision technology that involves identifying individuals from digital images or videos. Face recognition technology has numerous applications, including security systems, surveillance, and attendance tracking. In Python, face recognition can be implemented using the face recognition library, which is built on top of deep learning models.

The face recognition library in Python utilizes convolutional neural networks(CNNs) to extract facial features from images and videos. The CNNs are trained on large datasets of faces to learn to identify unique features that differentiate one person from another. These unique features are then used to recognize individuals in new images and videos.

The face recognition library in Python supports different types of face recognition algorithms, including the Eigen faces algorithm, Fisher faces algorithm, and Local Binary Patterns Histograms (LBPH) algorithm is used in attendance system. These algorithms differ in their approach to feature extraction and matching, but they all rely on deep learningmodels to identify individuals accurately.

To use the face recognition library in Python, one needs to first install the library and its dependencies, including OpenCV, Dlib library and NumPy. Once installed, the library can be used to load pre-trained models, detect faces in images and videos, and recognize individuals. The library also supports training custom models on custom datasets, which can be used for specific applications. The dlib library in python used to provide tools for face detection, landmark detection, face recognition, face shape prediction and it can able to detect faces under low light and partial occlusion with combination of ML algorithms like HOG algorithm. Attendance system will identify the faces of person who works in Office or Laboratory will  announce the workspace name and name of the employee.

### 1.3.3 Temperature Detection

MLX90614 temperature sensor are used to measure the body temperature of individuals as they enter a building or room, allowing for the detection of fever, which is oneof the common symptoms of COVID-19.

In a facial recognition attendance system that includes temperature sensors, the system first detects the individual's face using facial recognition technology, and then uses the temperature sensor to measure their body temperature. If the temperature reading is within the acceptable range, the individual is granted access to the building or room, and their attendance is marked.

The temperature sensor used in these systems is typically a non-contact infrared sensor that measures the infrared radiation emitted by the body. This type of sensor is ideal for use in attendance systems because it does not require physical contact with the individual, reducing the risk of disease transmission.

### 1.3.4 Liveliness Detection

Liveliness detection is an important aspect of face recognition systems as it helps to ensure that the system is not being tricked by fake or manipulated images. Liveliness detection involves the use of various techniques to determine whether the face being presented to thesystem is from a live person or a still image or a video.

The technique used for Liveliness detection in face recognition:

Facial movement detection: Live faces exhibit natural movements that still images or videos cannot replicate. The system looks for slight changes in the facial features such as subtle head movements, facial expressions, and changes in skin texture or color, which indicate that the face is live.

### 1.3.5 Result Analysis

The Results of attendance system are announced via voice engine and store data about Registered candidate or employee details. The Three results are provided as output are

Authentication of Employee**:**

By Identifying Employee identity by using Facial Recognition and dataset and broadcast by means of voice welcome message.

- **Liveliness Detection**

During Face Recognition, the system looks for slight changes in the facial features such as subtle head movements, facial expressions, and changes in skin texture or color, which indicate that the face is live for authentication purpose.

- **Thermal Analysis**

After Authentication of Employee, Temperature are inspected by sensor and stated by voice and if the temperature is higher, then it will notify to System admin.

- **Database**

At last, The Records are stored in format of Comma-Separated Values or CSVfile for future purpose.

# CHAPTER 2
# SYSTEM ANALYSIS

## 2.1 Literature Survey

### 2.1.1 Khushbu Gupta, Aakanksha S,” A Review on Face Detection based Attendance System with Temperature Monitoring”, 20 December 2020.

The face is the identity of a person. The method to exploit this physical feature have seen a great change since the advent of image processing techniques. The attendance is taken in every school, colleges and library. Traditional approach for attendance is professor calls student name & record attendance. Now a day, Machine Learning has been highly explored for computer vision applications. So, we use the concept of machine learning in Face – recognition for automatic attendance systems. In this project, we perform the face recognition and face detection algorithms, to provide the computer systems the ability of finding and recognizing human faces fast and precisely in images or videos so that the systems can used in giving attendance. Along with the face detection temperature measurement is also done using mlx90614 infrared temperature sensor.

### 2.1.2 Busra Kocacinar, Bilal Tas, Fatma Patlar Akbulut, (Member, Ieee), Cagatay Catal , And Deepti Mishra, (Senior Member, Ieee) , “A Real-Time CNN-Based Lightweight Mobile Masked Face Recognition System”, June 13,2022.

Due to the global spread of the Covid-19 virus and its variants, new needs and problems have emerged during the pandemic that deeply affects our lives. Wearing masks as the most effective measure to prevent the spread and transmission of the virushas brought various security vulnerabilities. Today we are going through times when wearing a mask is part of our lives, thus, it is very important to identify individuals who violate this rule. Besides, this pandemic makes the traditional biometric authentication systems less effective in many cases such as facial security checks, gated community access control, and facial attendance. So ar, in the area of masked face recognition, a small number of contributions have been accomplished. It is definitely imperative to enhance the recognition performance of the traditional face recognition methods on masked faces. Existing masked face recognition approaches are mostly performed based on deep learning models that require plenty of samples. Nevertheless, there are not enough image datasets containing a masked face. As such, the main objective of this study is to identify individuals who do not use masks or use them incorrectly and to verify their identity by building a masked face dataset. On this basis, a novel real-time masked detection service and face recognition mobile application was developed based on an ensemble of fine-tuned lightweight deep Convolutional Neural Networks (CNN). The proposed model achieves 90.40% validation accuracy using 12 individuals' 1849 face samples. Experiments on the five datasets built in this research demonstrate that the proposed system notably enhances the performance of masked face recognition compared to the other state-of-the-art approaches.

### 2.1.3 Chaoyou Fu, Xiang Wu, Yibo Hu, Huaibo Huang, and Ran He*, Senior Member, IEEE, DVG-Face: "Dual Variational Generation for Heterogeneous Face Recognition", July, 2020.

Heterogeneous Face Recognition (HFR) refers to matching cross-domain faces and plays a crucial role in public security. Nevertheless, HFR is confronted with challenges from large domain discrepancy and insufficient heterogeneous data. In this paper, we formulate HFR as a dual generation problem, and tackle it via a novel Dual Variational Generation (DVG-Face) framework. Specifically, a dual variational generator is elaborately designed to learn the joint distribution of paired heterogeneous images. However, the small-scale paired heterogeneous training data may limit the identity diversity of sampling. In order to break through the limitation, we propose to integrate abundant identity information of large-scale visible data into the joint distribution. Furthermore, a pair wise identity preserving loss is imposed on the generated paired heterogeneous images to ensure their identity consistency. As a consequence, massive new diverse paired heterogeneous images with the same identity can be generated from noises. The identity consistency and identity diversity properties allow us to employ these generated images to train the HFR network via a contrastive learning mechanism, yielding both domain-invariant and discriminative embedding features. Concretely, the generated paired heterogeneous images are regarded as positive pairs, and the images obtained from different samplings are considered as negative pairs. Our method achieves superior performances over state-of-the-art methods on seven challenging databases belonging to five HFR tasks, including NIR- VIS, Sketch-Photo, Profile-FrontalPhoto, Thermal-VIS, and ID-Camera.

### 2.1.4 A Arjun Raj, Mahammed Shoheb, K Arvind, K S Chethan,” A Face Recognition Based Smart Attendance System”, June 19, 2020.

Education institutes today are concerned about the consistency of students ' performance. One cause of this decrease in student performance is the inadequate attendance. There are several ways to mark your attendance, the most common ways tosign or call the students. It took longer and was problematic. From now on, a computer-based student attendance checking system is required that supports the faculty to keep records of attendance. An intelligent attendance system based on face recognition in this project. To implement a "Smart Attendance System for Face Recognition" through this large application are incorporated. The present implementation includes facial identification that is time saving and eradicates the possibilities of proxy attendance due to the facial authorization. This system can now be used in an area in which participation plays an important role. Raspberry Pi, Open CV and Dlib using python are the basic requirements for this system. The system implemented uses LBPH face recognizer to identify the face of the person in real time. Eigen faces and Fisher faces are affected both by light and we cannot ensure perfect light conditions in real life. An improvement in the LBPH faces recognizer to overcome this problem. This system compares the image of the test and the training image and determines who is and is not present. The attendance data is stored in an excel sheet that is automatically updated in the system. If a student is absent a message will be automatically sent to their parent's phone number using GSM. Students can check their attendance using an Android application that we have developed using MIT app Inventor.

### 2.1.5 Dr. V Suresh, Srinivasa Chakravarthi Dumpa, Chiranjeevi Deepak Vankayala, HaneeshaAduri, Jayasree Rapa, "Facial Recognition Attendance System Using Python and OpenCv", March 5, 2020.

The main purpose of this project is to build a face recognition-based attendance monitoring system for educational institution to enhance and upgrade the current attendance system into more efficient and effective as compared to before. The current old system has a lot of ambiguity that caused inaccurate and inefficient of attendance taking. Many problems arise when the authority is unable to enforce the regulation that exist in the old system. The technology working behind will be the face recognition system. The human face is one of the natural traits that can uniquely identify an individual. Therefore, it is used to trace identity as the possibilities for a face to deviate or being duplicated is low. The face databases will be created to pump data into the recognizer algorithm. Then, during the attendance taking session, faces will be compared against the database to seek for identity. When an individual is identified, its attendance will be taken down automatically saving necessary information into a excel sheet. At the end of the day, the excel sheet containing attendance information regarding all individuals are mailed to the respective faculty.

## 2.2 Existing System

The existing system is based on Face Recognition Attendance Management System which integrates images of person’s face authentication into the process of attendance management for both staff and student. It is made up of two processes namely; enrolment andauthentication. Attendance management is the act of managing attendance or presence in a work setting to minimize loss due to employee downtime. Attendance control has traditionally been approached using time clocks and timesheets, but attendance management goes beyond this to provide a working environment which maximizes and motivates employee attendance.

**Disadvantages**

- Traditional projects did not use Real time Liveliness of people’s image.
- People must submit the picture in order to detect face so it will take more time.
- Accuracy will be low.

## 2.3 Proposed System

The Goal of the proposed system is to realize an attendance system that can be managed through face recognition. To be a good attendance system, it must have several characteristics, namely a high recognition rate, fast recognition, and the ability to cope with many different faces. This work uses two-stage networks to reach the goal of face detection and recognition. First, the system will continuously detect the input screen. When a face is detected on the screen, the system will immediately crop the face. Then, the cropped face image will be sent to the face recognition model to generate a facial feature vector, which will be compared with the facial features previously built into the database to confirm the identity of the person. Finally, the recognition results will be announced by voiceover and body temperature will be measured by using temperature sensor. After this process will be done, it will store a user information on CSV file format. This method can be used to record the temperature of people while assisting them to punch in and out. In today’s rampant COVID-19 environment, it is important to measure and record the temperature of people entering and exiting.

**Advantages**

- For security purpose the face recognition will validate the Liveliness of people.
- High reliability and accuracy.
- Time consumption is less.
- Real-time streaming makes the system more convenient to detect the face recognition.

# CHAPTER 3
# SYSTEM REQUIREMENTS

## 3.1 Hardware Specification

- Webcam
- Arduino Uno
- MLX90614 Temperature Sensor
- Laptop
- Breadboard
- Jumper Pins

## 3.2 Hardware Description

### 3.2.1 Webcam

The system uses inbuilt USB 2.0 webcam for capturing and performing social distanceoperations. A webcam is a video camera that feeds or streams an image or video in real timeto or through a computer to a computer network, such as the Internet. Webcams are typicallysmall cameras that sit on a desk, attach to a user's monitor, or are built into the hardware. Webcams can be used during a video chat session involving two or more people, with conversations that include live audio and video. Webcam software enables users to record a video or stream the video on the Internet. As video streaming over the Internet requires much bandwidth, such streams usually use compressed formats. The maximum resolution of a webcam is also lower than most handheld video cameras, as higher resolutions would be reduced during transmission. The lower resolution enables webcams to be relatively inexpensive compared to most video cameras, but the effect is adequate for video chat sessions.

The term "webcam" (a clipped compound) may also be used in its original sense of a video camera connected to the Web continuously for an indefinite time, rather than for a particular session, generally supplying a view for anyone who visits its web page over the Internet. Some of them, for example, those used as online traffic cameras, are expensive, rugged professional video cameras.

### 3.2.2 Arduino Uno

The Arduino Uno is an open-source microcontroller board based on the ATmega328P microprocessor. It contains 14 digital input/output pins, 6 analogue inputs, a 16 MHz quartz crystal, a USB connection, a power connector, and an ICSP header. It may be programmed using the Arduino IDE (Integrated Development Environment) and used to build a variety of electrical applications like as robots, home automation systems, and sensors. The digital pins may be utilized for both input and output and can deliver up to 40 mA of current to each pin. The analogue inputs may detect voltages ranging from 0 to 5 volts, and the board includes an analog-to-digital converter (ADC) that converts the analogue input values into digital signals that can be read by a computer.

### 3.2.3 MLX90164 Temperature Sensor

The MLX90614 is an infrared (IR) temperature sensor that can measure the temperature of an object without making physical contact with it. The sensor uses IR radiation to detect the temperature of an object, and converts the radiation into an electrical signal that can be read by a microcontroller or other digital device. The MLX90614 has a measurement range of -70°C to +380°C, and has an accuracy of ±0.5°C between -20°C and +85°C. The sensor has a small form factor and can be easily integrated into electronic projects. The MLX90614 communicates with other devices using the I2C protocol, and can be powered using a voltage between 3.3V and 5V. The sensor also includes a 17-bit ADC (analog-to-digital converter) for high-resolution temperature readings.

## 3.3 Software Specification

- Python 3.7
- Pycharm
- Arduino IDE

## 3.4 Software Description

### 3.4.1 Python

Python is an interpreted, high-level, general-purpose programming language Created by Guido van Rossum and first released in 1991, Python's design philosophy emphasizes code readability with its notable use of significant whitespace. Its language constructs and object-oriented approach aim to help programmers write clear, logical code for small and large-scale projects.

Python is a high-level programming language designed to be easy to read and simpleto implement. It is open source, which means it is free to use, even for commercial applications. Python can run on Mac, Windows, and Unix systems and has also been ported to Java and .NET virtual machines.

Python is dynamically typed and garbage-collected. It supports multiple programming paradigms, including procedural, object-oriented, and functional programming. Python is often described as a "batteries included" language due to its comprehensive standard library.

### 3.4.2 PyCharm

PyCharm is an integrated development environment (IDE) for Python programming language. It is developed by JetBrains and is available in both free and paid versions. PyCharm provides a comprehensive set of tools for developing and debugging Python code, including code completion, syntax highlighting, refactoring, debugging, testing, and version control integration. It also has support for web development frameworks such as Django, Flask, and Pyramid. Some of the key features of PyCharm include a customizable user interface, integration with popular version control systems such as Git and Mercurial, a powerful debugger with remote debugging capabilities, code analysis and inspection tools, and support for virtual environments. PyCharm is widely used by developers and is considered one of the most popular Python IDEs available.

### 3.4.3 Arduino IDE

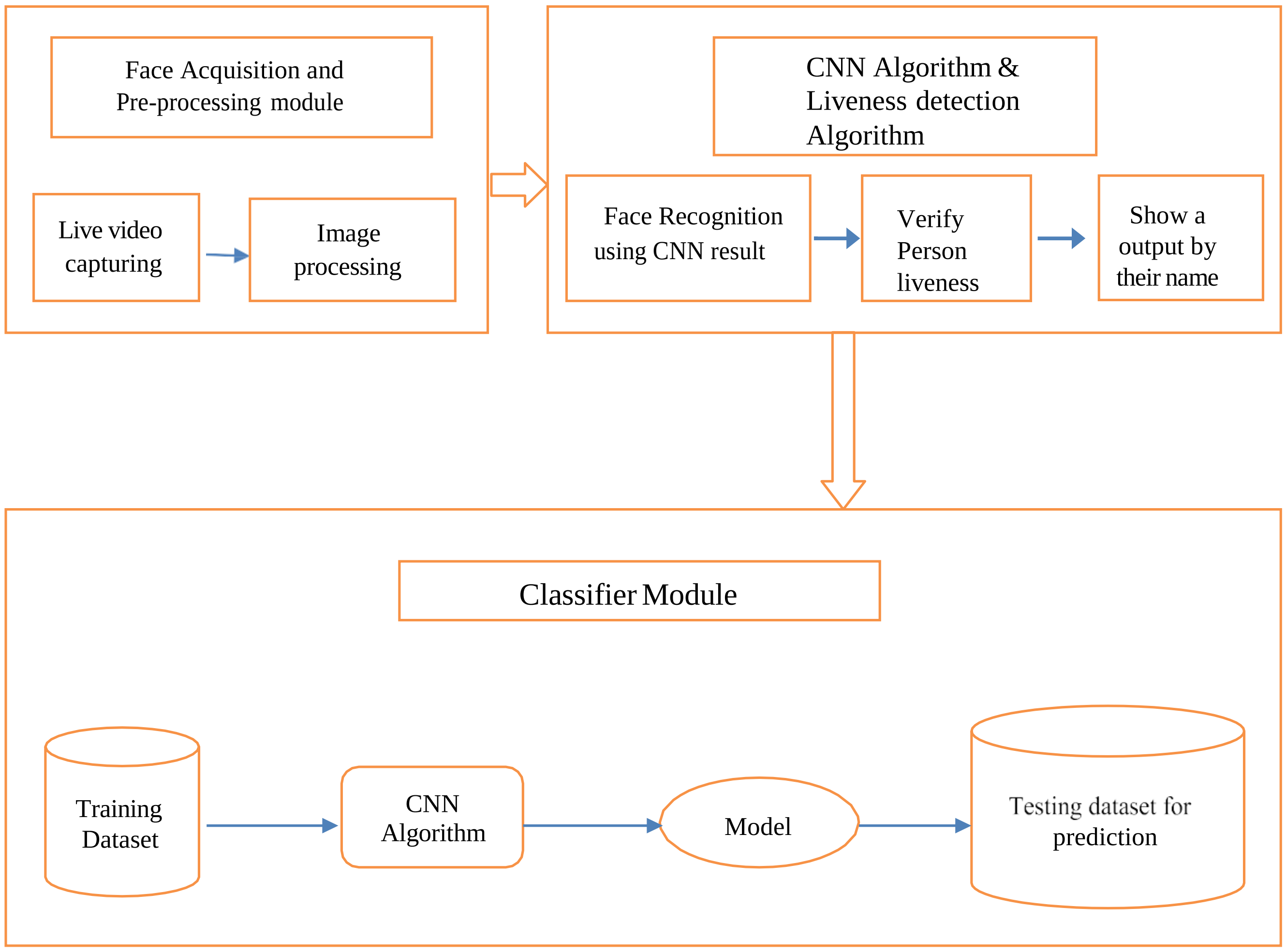
The Arduino IDE (Integrated Development Environment) is a free and open- source software tool for writing and uploading code to Arduino boards. Arduino is a free and open-source electronics platform with simple hardware and software. It is made comprisedof a microcontroller board and a software development environment that is used to write, compile, and upload code to the board. The Arduino IDE is intended to be user-friendly, evenfor those with little or no programming knowledge. It comes with a code editor that providessyntax highlighting, code completion, and error checking. A serial monitor is also includedin the IDE for debugging and testing the code on the board. The Arduino IDE supports a variety of programming languages, including C and C++, making it simple to build code for Arduino boards.

# CHAPTER 4

# SYSTEM DESIGN

## 4.1 Architecture Diagram

An architecture diagram is a graphical representation of a set of concepts, that are part of an architecture, including their principles, elements and components



Face Acquisition and Pre-processing module
Live video capturing
Image processing
CNN Algorithm & Liveness detection Algorithm
Face Recognition using CNN result
Verify Person liveness
Show a output by their name
Classifier Module
Training Dataset
CNN Algorithm
Model
Testing dataset for prediction

**Fig. 4.1 Architecture Diagram**

## 4.2 Use Case Diagram

A use case diagram in the Unified Modelling Language (UML) is a type of behavioral diagram defined by and created from a Use-case analysis. Its purpose is to represent a graphicaloverview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases.

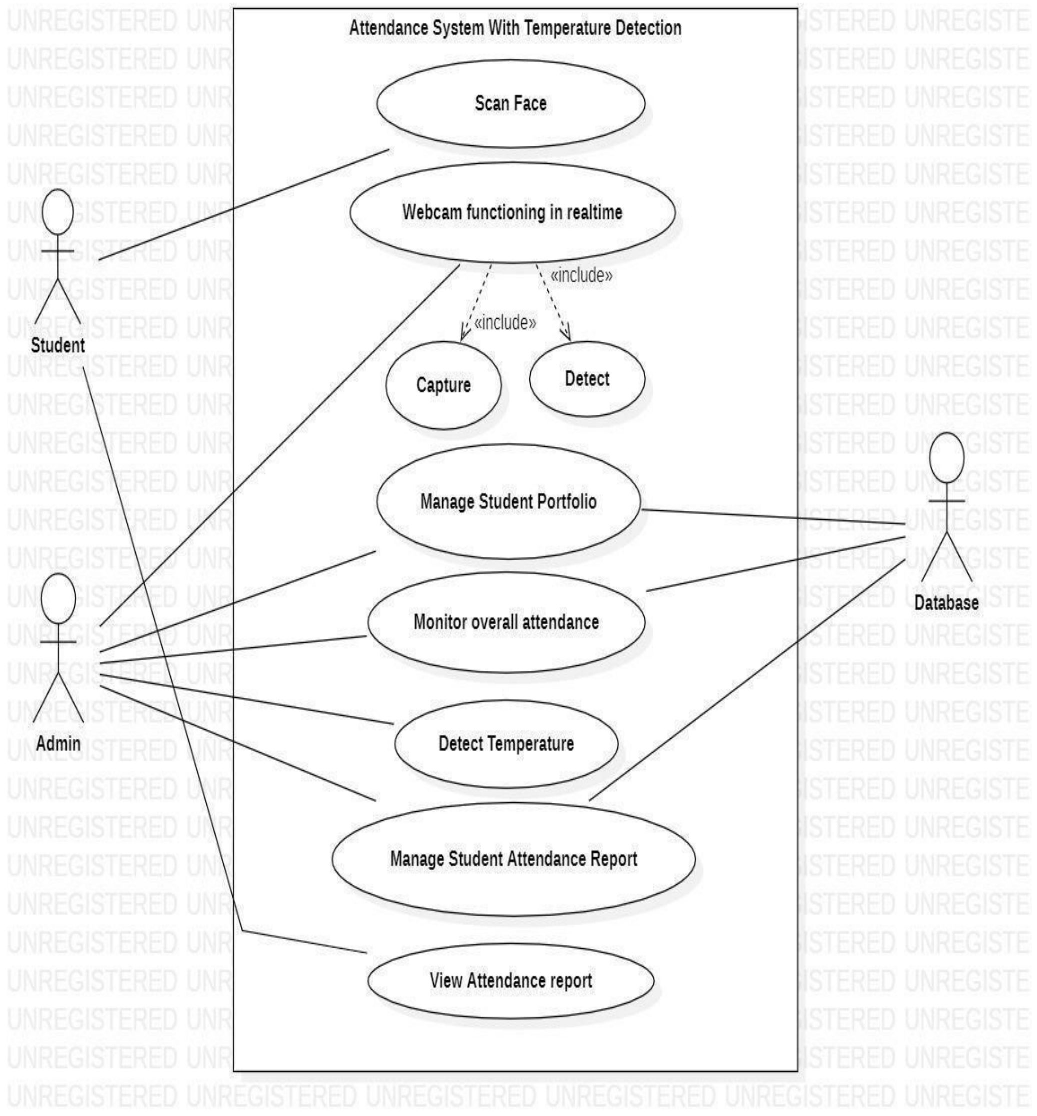

Attendance System With Temperature Detection
Scan Face
Webcam functioning in realtime
«include»
«include»
Capture
Detect
Manage Student Portfolio
Monitor overall attendance
Detect Temperature
Manage Student Attendance Report
View Attendance report
Student
Admin
Database

**Fig. 4.2 Use Case Diagram**

## 4.3 Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modelling Language, activity diagrams can be used to describe the business and operational step-by-step workflows of components in a system. An activity diagram shows the overall flow of control.

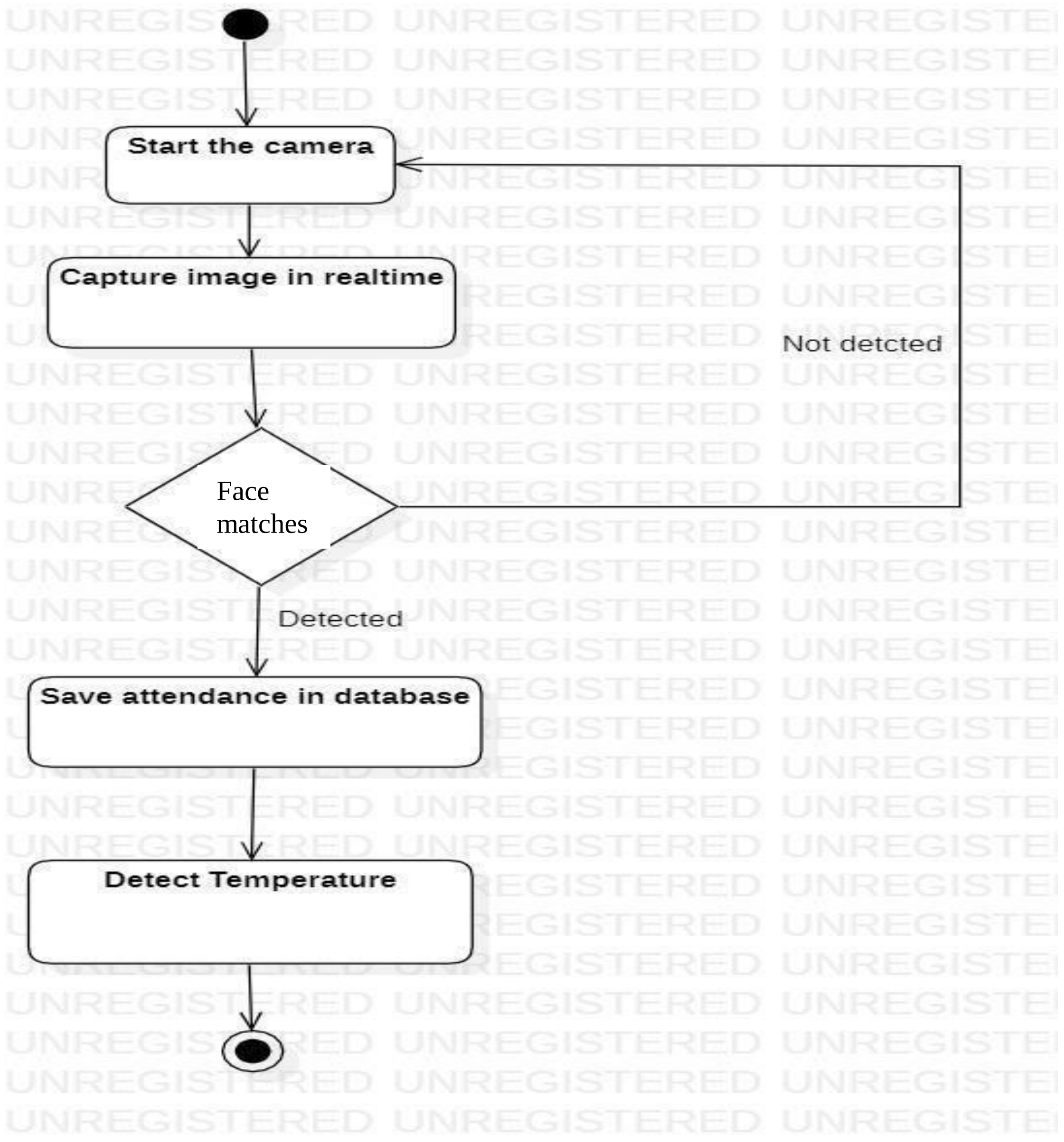


**Fig. 4.3 Activity Diagram**

# CHAPTER 5
# SYSTEM IMPLEMENTATION

## 5.1 List of Modules

- Image Processing
- Facial Recognition
- Temperature Detection
- Liveliness detection
- Real Time Tracking

## 5.2 Module Description

### 5.2.1 Image Processing

The alteration of digital photographs using mathematical methods and computer technology is referred to as image processing. Image processing aims to improve, edit, or analyze digital pictures in order to extract usable information from them.

Image processing methods are roughly classified into two categories:

Analog Image Processing is the manipulation of physical pictures via the use of optical and mechanical processes. Analog image processing techniques include the use of filters, lenses, and mirrors to alter pictures.

Digital image processing is the manipulation of digital pictures via the use of computer algorithms. Image enhancement, restoration, compression, and analysis are all uses of digital image processing.

### 5.2.2 Facial Recognition

Face recognition is a computer vision technology that involves identifyingindividuals from digital images or videos. Face recognition technology has numerous applications, including security systems, surveillance, and attendance tracking. In Python, face recognition can be implemented using the face recognition library, which is built on top of deep learning models.

The face recognition library in Python utilizes convolutional neural networks(CNNs) to extract facial features from images and videos. The CNNs are trained on large datasets of faces to learn to identify unique features that differentiate one person from another. Theseunique features are then used to recognize individuals in new images and videos.

The face recognition library in Python supports different types of face recognition algorithms, including the Eigen faces algorithm, Fisher faces algorithm, and Local Binary Patterns Histograms (LBPH) algorithm is used in attendance system. These algorithms differ in their approach to feature extraction and matching, but they all rely on deep learningmodels to identify individuals accurately.

To use the face recognition library in Python, one needs to first install the library and its dependencies, including OpenCV, Dlib library and NumPy. Once installed, the library can be used to load pre-trained models, detect faces in images and videos, and recognize individuals. The library also supports training custom models on custom datasets, which can be used for specific applications. The dlib library in python used to provide tools for face detection, landmark detection, face recognition, face shape prediction and it can able to detect faces under low light and partial occlusion with combination of ML algorithms like HOG algorithm. Attendance system will identify the faces of person who works in Office or Laboratory will  announce the workspace name and name of the employee.

### 5.2.3 Temperature Detection

The MLX90614 sensor measures temperature without being physically touched, unlike the majority of temperature sensors. By measuring the amount of IR energy radiated by an object, the MLX90614 sensor determines its temperature. This is often made possible with a law called Stefan-Boltzmann Law, which states that each one objects and living beings emit IR Energy and therefore the intensity of this emitted IR energy is going to be directly proportional to the temperature of that object or living being. The MLX90164 include a measurement range of -40°C to 125°C, high accuracy of ±0.5°C (at 25°C), low power consumption, and a small package size. The sensor is also factory calibrated and canprovide a temperature reading in either Celsius or Fahrenheit.

The MLX90164 temperature sensor works based on the principles of infrared radiation. The sensor consists of an infrared thermopile detector that can detect the infrared radiation emitted by an object and convert it into a temperature reading. When the sensor is aimed at an object, it receives the infrared radiation emitted by the object. The infrared radiation is then focused onto the thermopile detector by a lens. The thermopile detector is made up of multiple thermocouples connected in series, which generate a voltage proportional to the temperature difference between the object and the thermopile. The voltage generated by the thermopile is then amplified and converted into a digital temperature reading by an onboard analog-to-digital converter. The MLX90164 temperature sensor also features an onboard EEPROM memory that stores calibration data, allowing the sensor to provide accurate temperature readings without the need for manual calibration. To use the MLX90164 temperature sensor, it needs to be connected to a microcontroller or computer using a communication protocol such as I2C. The sensor can provide temperature readings in either Celsius or Fahrenheit, and it can be configured to operate in a continuous measurement mode or a one-shot measurement mode.

### 5.2.4 Liveliness Detection

Liveliness detection in facial recognition systems works by detecting and verifying he presence of certain biometric and behavioral features that are unique to live humans Livelinessdetection is a critical component of facial recognition technology that helps to ensure the authenticity of a facial recognition system. Liveliness detection is the ability ofa facial recognition system to determine whether the facial biometric data being captured is from a live person or a fake representation of a person, such as a photograph or video. The techniques that can be used for Liveliness detection in facial recognition system is TextureAnalysis.

### 5.2.5 Texture Analysis:

This technique involves analyzing the texture of the skin on a person's face to determine whether it is consistent with that of a live person. This can be done by detecting changes in skin color, texture, and temperature, which can indicate whether the person is real or not.

Overall, Liveliness detection in facial recognition systems works by verifying the presence of biometric and behavioral features unique to live humans. Hence, Liveliness detection is an important component of facial recognition technology, as it helps to prevent fraudulent activity and ensures the accuracy and reliability of the system.

### 5.2.6 Real Time Tracking

Real-time tracking of facial recognition systems involves continuously analyzing livevideo feeds and identifying individuals by comparing their facial features to a database of known faces. It is an essential feature in face recognition systems that enables continuous identification and monitoring of individuals in a given environment. This feature allows the system to track the movement and location of individuals in real-time using video footage captured by cameras. Here, tracking in face recognition involves the use of complex algorithms that analyze and compare facial features to detect and track individuals. These algorithms enable the system to identify individuals from different angles and under differentlighting conditions. The use of real- time tracking in face recognition systems has revolutionized security and surveillance applications, making it easier to monitor and track individuals in real-time.

# CHAPTER 6

# RESULTS AND DISCUSSION

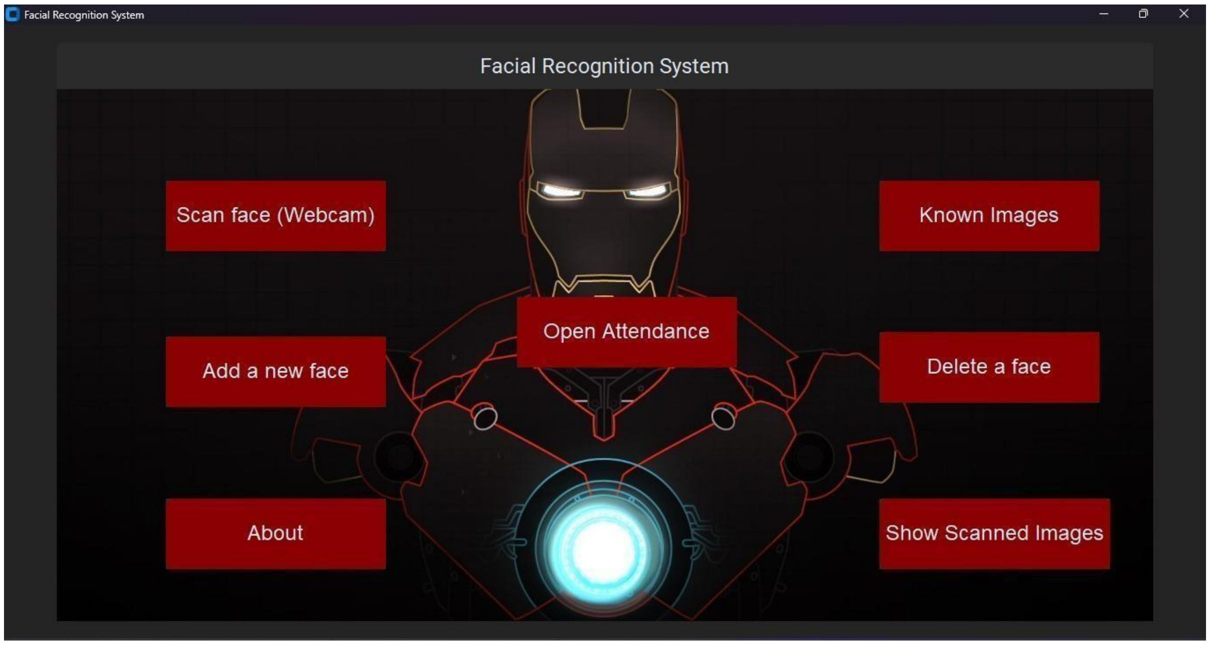


**Fig. 6.1 Home Page**

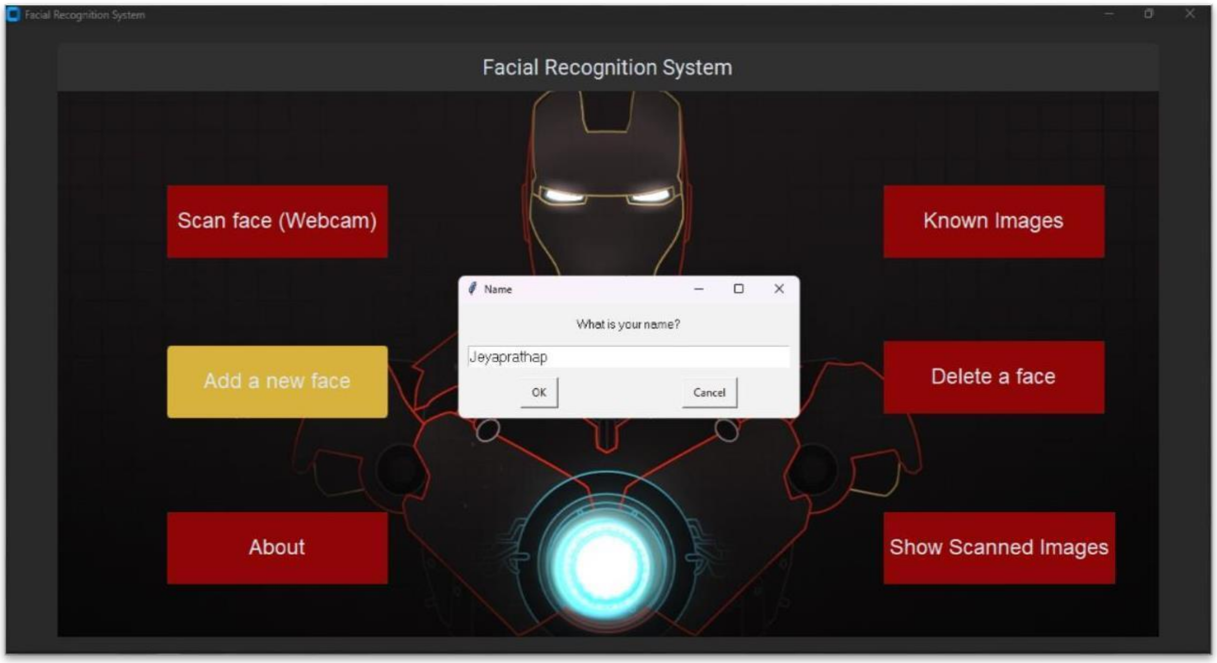


**Fig. 6.2 Register Page to Enroll New User**

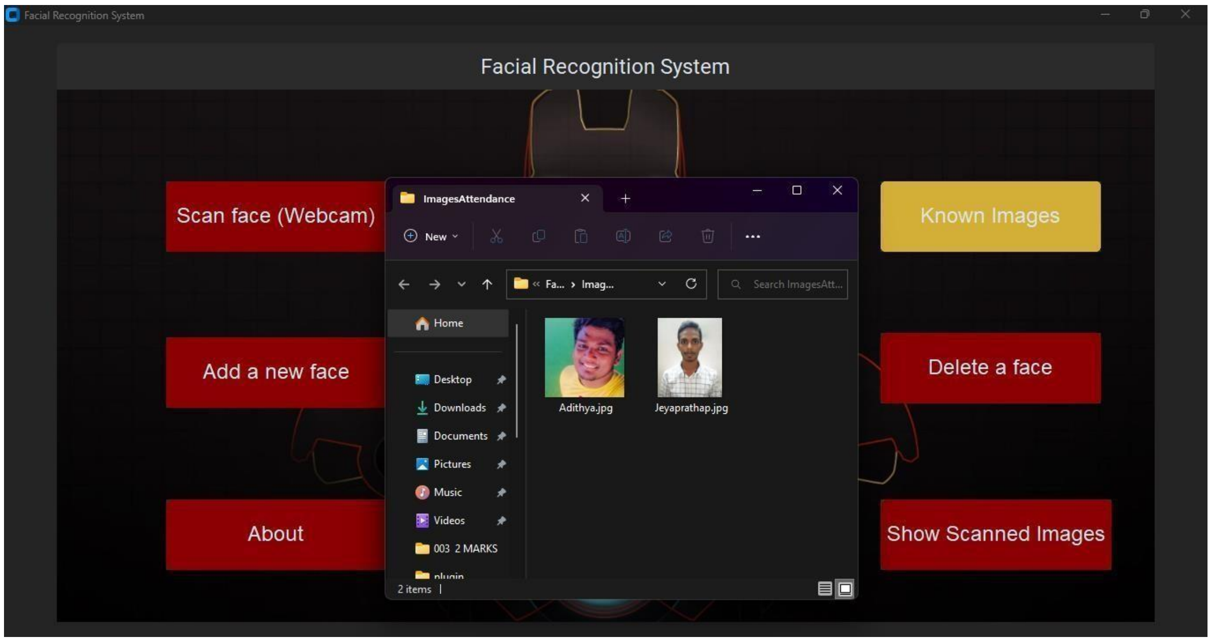


Fig. 6.3 Known User Page

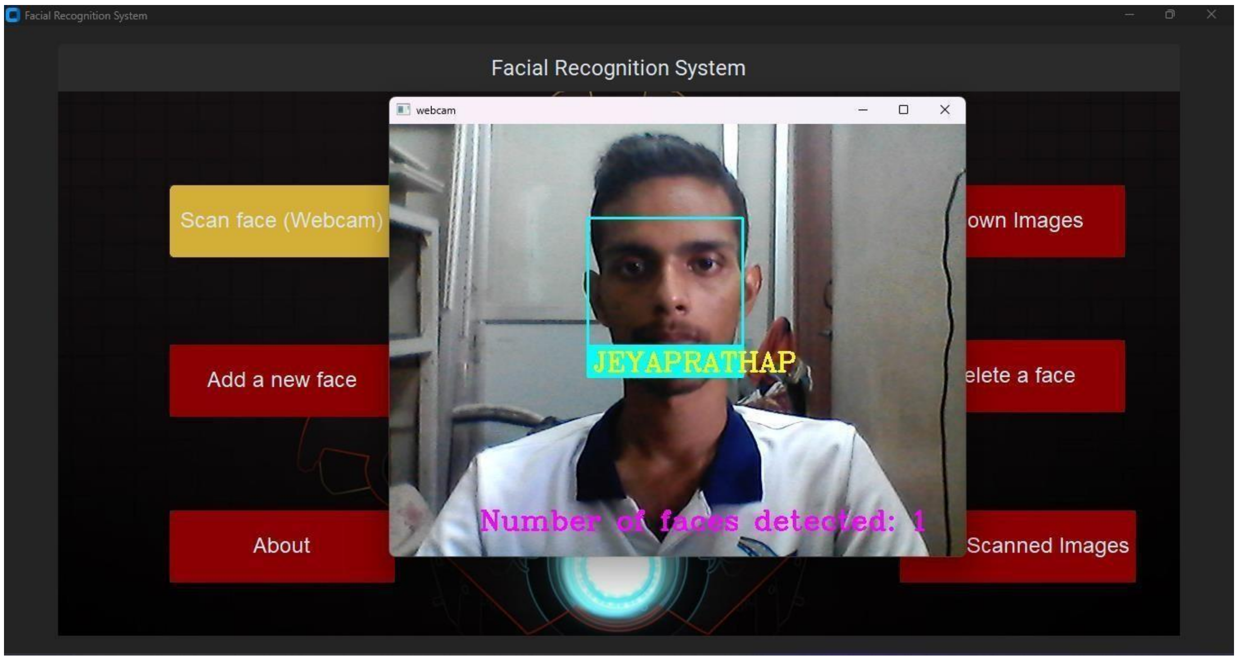


Fig. 6.4 Attendance System Page

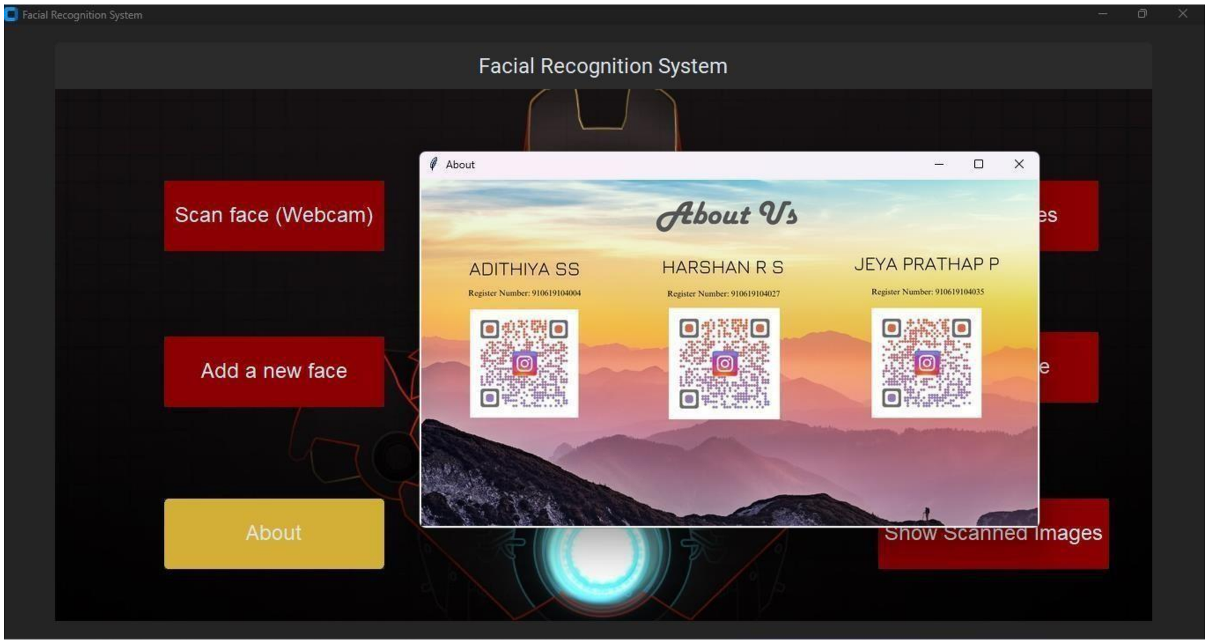
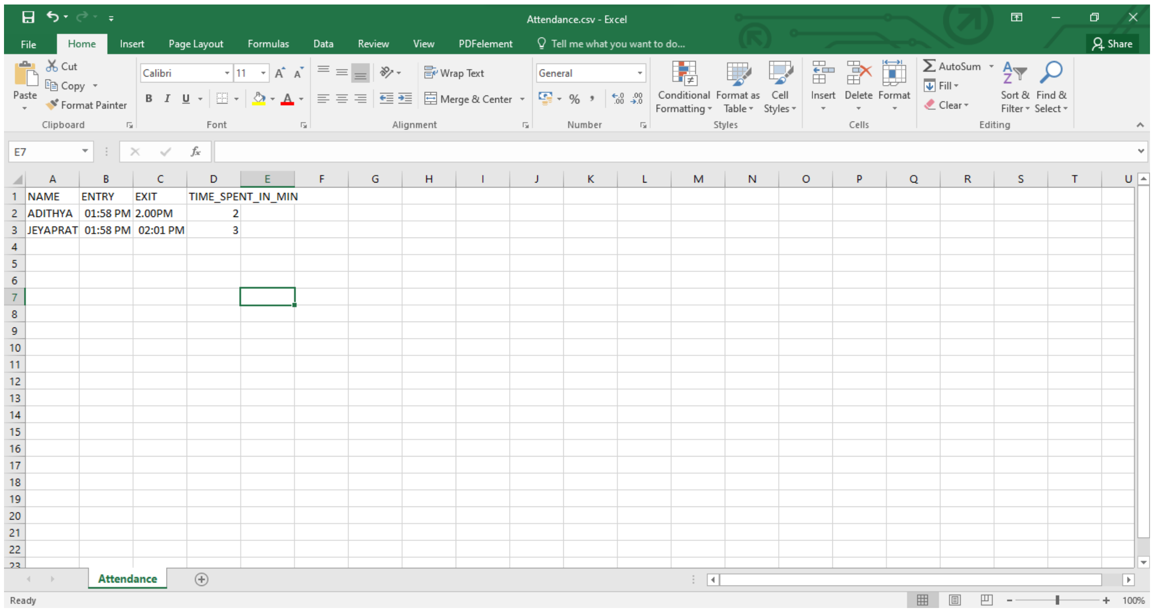

Facial Recognition System
Facial Recognition System
Scan face (Webcam)
Add a new face
About
Show Scanned Images
About
About Us
ADITHIYA SS
Register Number: 910619104004
HARSHAN R S
Register Number: 910619104027
JEYA PRATHAP P
Register Number: 910619104035

Fig. 6.5 About Page



| NAME | ENTRY | EXIT | TIME_SPENT_IN_MIN |
|---|---|---|---|
| ADITHYA | 01:58 PM | 2.00PM | 2 |
| JEYAPRAT | 01:58 PM | 02:01 PM | 3 |

Fig. 6.6 Attendance Report Page

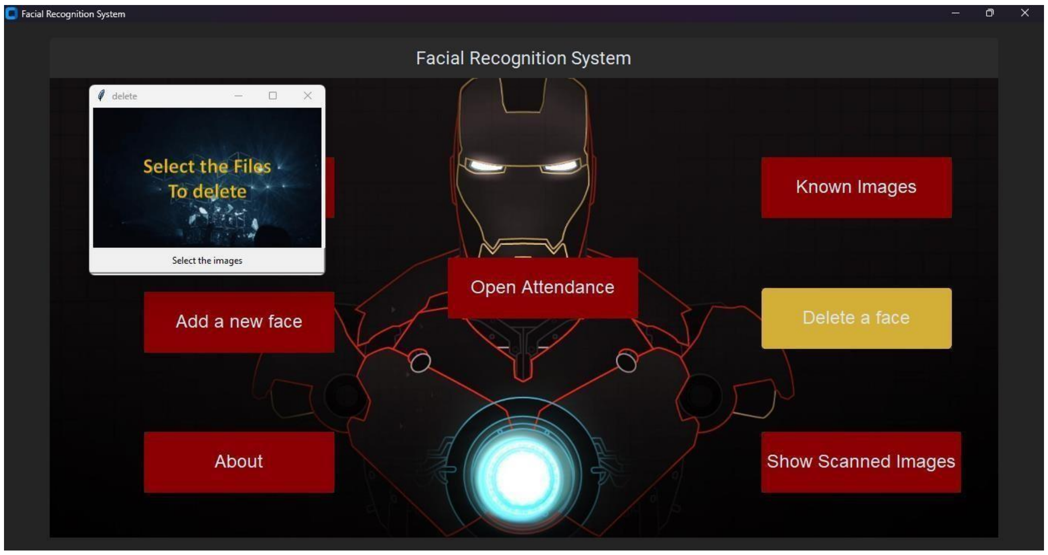
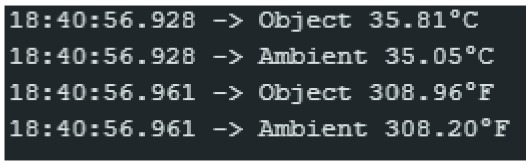


Fig. 6.7 Remove User Page



```
18:40:56.928 -> Object 35.81°C
18:40:56.928 -> Ambient 35.05°C
18:40:56.961 -> Object 308.96°F
18:40:56.961 -> Ambient 308.20°F
```

Fig 6.9 Temperature Detection in Mlx90614

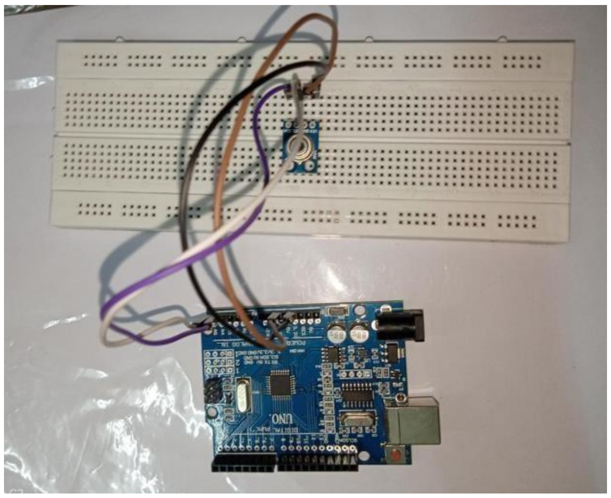


Fig 6.8 Arduino Uno for Temperature Detection

# CHAPTER 7
# SAMPLE CODING

**Attendance System Code**

```
import cv2
import numpy as np import face_recognitionimport os
from datetime import datetimeimport pyttsx3
import time
from imutils.video import VideoStream
from tensorflow.keras.utils import img_to_arrayfrom keras.models import load_model
import argparseimport imutils import pickle
print("Loading face detector...")
protoPath = "face_detector/deploy.prototxt"
modelPath = "face_detector/res10_300x300_ssd_iter_140000.caffemodel" net =
cv2.dnn.readNetFromCaffe(protoPath, modelPath)
print("Loading Model...")
model = load_model("Liveliness.model")
le = pickle.loads(open("le.pickle", "rb").read())
print("Starting Video Stream")v = VideoStream(src=0).start()time.sleep(2.0)
path = 'ImagesAttendance'images = []
personNames = [] myList = os.listdir(path)print(myList)
for cl in myList:
curImg = cv2.imread(f'{path}/{cl}') images.append(curImg)
personNames.append(os.path.splitext(cl)[0])print(personNames)
def findEncodings(images):encodeList = []
for img in images:
img = cv2.cvtColor(img, cv2.COLOR_BGR2RGB) encode =
face_recognition.face_encodings(img)[0] encodeList.append(encode)
return encodeList
def markAttendance(name):
with open('Attendance.csv','r+') as f:
myDataList = f.readlines()nameList = []
for line in myDataList:entry = line.split(',')
nameList.append(entry[0])if name not in nameList:
now = datetime.now()
dtString = now.strftime('%H:%M:%S')f.writelines(f'\n{name},{dtString}')
encodeListKnown = findEncodings(images)print('Encoding Complete')
while True:
frame = v.read()
frame = imutils.resize(frame, width=600)
frame1 = cv2.cvtColor(frame, cv2.COLOR_BGR2RGB)(h, w) = frame.shape[:2]
blob = cv2.dnn.blobFromImage(cv2.resize(frame, (300, 300)), 1.0,
(300, 300), (104.0, 177.0, 123.0))
net.setInput(blob) detections = net.forward()
for i in range(0, detections.shape[2]):confidence = detections[0, 0, i, 2]
f confidence > 0.5:
sbox = detections[0, 0, i, 3:7] * np.array([w, h, w, h])(startX, startY, endX, endY) =
box.astype("int")
startX = max(0, startX)startY = max(0, startY)endX = min(w, endX) endY = min(h, endY)
face = frame[startY:endY, startX:endX]face = cv2.resize(face, (32, 32))
face = face.astype("float") / 255.0face = img_to_array(face)
face = np.expand_dims(face, axis=0)
preds = model.predict(face)[0]j = np.argmax(preds)
label = le.classes_[j]
label = "{}: {:.4f}".format(label, preds[j])print(label)
if preds[j] > 0.60 and j == 1:
cv2.rectangle(frame, (startX, startY), (endX, endY), (0, 255, 0), 2)
_label = "Real: {:.4f}".format(preds[j])
cv2.putText(frame, _label, (startX, startY - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.5,
(0, 255, 0), 2)
# If the face is real, check for attendance
facesCurFrame = face_recognition.face_locations(frame1)encodesCurFrame =
face_recognition.face_encodings(frame1,facesCurFrame)
for encodeFace,faceLoc in zip(encodesCurFrame,facesCurFrame):matches =
face_recognition.compare_faces(encodeListKnown,encodeFace)
faceDis =
face_recognition.face_distance(encodeListKnown,encodeFace)
print(faceDis)
matchIndex = np.argmin(faceDis)
if matches[matchIndex]:
name = personNames[matchIndex].upper()print(name)
y1,x2,y2,x1 = faceLoc
y1, x2, y2, x1 = y1*4,x2*4,y2*4,x1*4 cv2.rectangle(frame,(x1,y1),(x2,y2),(0,255,0),2)
cv2.rectangle(frame,(x1,y2-35),(x2,y2),(0,255,0),cv2.FILLED)
cv2.putText(frame,name,(x1+6,y2-
6),cv2.FONT_HERSHEY_COMPLEX,1,(255,255,255),2)
markAttendance(name)
engine = pyttsx3.init()
engine.setProperty("rate", 120)
engine.say(name)
engine.say("Welcome To K L N college of Engineering")
engine.runAndWait()
else:
cv2.rectangle(frame, (startX, startY), (endX, endY), (0, 0, 255), 2)
_label = "Unauthorized Identity !!! Fake/Spoofed:
{:.4f}".format(preds[j])
cv2.putText(frame, _label, (startX, startY - 10), cv2.FONT_HERSHEY_SIMPLEX, 0.5,
(0, 0, 255), 2)
cv2.imshow("Frame", frame)
key = cv2.waitKey(1) & 0xFF
if key == ord("q"):break
cv2.destroyAllWindows()
v.stop()
```

# CHAPTER 8
# SYSTEM TESTING

System testing is the stage of implementation, which aimed at ensuring that system works accurately and efficiently before the live operation commence. Testing is the processof executing a program with the intent of finding an error. A good test case is one that has ahigh probability of finding an error. A successful test is one that answers a yet undiscoverederror.

Testing is vital to the success of the system. System testing makes a logical assumption that if all parts of the system are correct, the goal will be successfully achieved. The candidate system is subject to variety of tests-on-line response, Volume Street, recoveryand security and usability test. A series of tests are performed before the system is ready forthe user acceptance testing. Any engineered product can be tested in one of the following ways. Knowing the specified function that a product has been designed to from, test can be conducted to demonstrate each function is fully operational. Knowing the internal workingof a product, tests can be conducted to ensure that “al gears mesh”, that is the internal operation of the product performs according to the specification and all internal componentshave been adequately exercised.

## 8.1 Unit Testing

Unit testing is the testing of each module and the integration of the overall system is done. Unit testing becomes verification efforts on the smallest unit of software design in the module. This is also known as ‘module testing’. The modules of the system are tested separately. This testing is carried out during the programming itself. In this testing step, eachmodel is found to be working satisfactorily as regard to the expected output from the module. There are some validation checks for the fields. For example, the validation check is done for verifying the data given by the user where both format and validity of the data enteredis included.  It is very easy to find error and debug the system.

### 8.1.1 Test Strategy and Approach

Field testing will be performed manually and functional tests will be  written in detail.

- Test Objectives
- Identify the person’s face properly.
- Module must be activated by all the time.
- The entry screen, messages and responses must not be delayed.

### 8.1.2 FEATURES TO BE TESTED

- Verify that the entries are of the correct format.
- No duplicate entries should be allowed.
- All entries should be noted and recorded.

## 8.2 Integration Testing

Data can be lost across an interface, one module can have an adverse effect on the other sub function, when combined, may not produce the desired major function. Integrated testing is systematic testing that can be done with sample data. The need for the integrated test is to find the overall system performance. There are two types of integration testing. Theyare:

- Top-down integration testing.
- Bottom-up integration testing.

## 8.3 Functional Testing

Functional tests provide systematic demonstrations that functions tested are available as specified by the business and technical requirements, system documentation, and user manuals.

Functional testing is centered on the following items:

- Valid Input - identified classes of valid input must be accepted.
- Invalid Input - identified classes of invalid input must be rejected.
- Functions - identified functions must be exercised.
- Output - identified classes of application outputs must be exercised.
- Systems/Procedures - interfacing systems or procedures must be invoked.

Organization and preparation of functional tests is focused on requirements, key functions, or special test cases. In addition, systematic coverage pertaining to identify Business process flows; data fields, predefined processes, and successive processes must be considered for testing. Before functional testing is complete, additional tests are identified and the effective value of current tests is determined.

## 8.4 Testing Techniques/Strategies

### 8.4.1 White Box Testing

White Box testing is a test case design method that uses the control structure of the procedural design to drive cases. Using the white box testing methods, we derived test cases that guarantee that all independent paths within a module have been exercised at least once.

### 8.4.2 Black Box Testing

- Black box testing is done to find incorrect or missing function
- Interface error
- Errors in external database access

- Performance errors
- Initialization and termination errors

In ‘functional testing’, is performed to validate an application conforms to its specifications of correctly performs all its required functions. So this testing is also called ‘black box testing’. It tests the external behavior of the system. Here the engineered product can be tested knowing the specified function that a product has been designed to perform, tests can be conducted to demonstrate that each function is fully operational.

## 8.5 Software Testing Strategies

### 8.5.1 Validation Testing

After the culmination of black box testing, software is completed assembly as a package, interfacing errors have been uncovered and corrected and final series of software validation tests begin validation testing can be defined as many, but a single definition is that validation succeeds when the software functions in a manner that can be reasonably expectedby the customer.

### 8.5.2 User Acceptance Testing

User acceptance of the system is the key factor for the success of the system. The system under consideration is tested for user acceptance by constantly keeping in touch withprospective system at the time of developing changes whenever required.

## 8.6 Output Testing

After performing the validation testing, the next step is output asking the user about the format required testing of the proposed system, since no system could be useful if it does not produce the required output in the specific format. The output displayed or generated bythe system under consideration. Here the output format is considered in two ways. One is screen and the other is printed format.

The output format on the screen is found to be correct as the format was designed in the system phase according to the user needs. For the hard copy also output comes out as the specified requirements by the user. Hence the output testing does not result in any connection in the system.

# CHAPTER 9
# CONCLUSION AND FUTURE ENHANCEMENT

## 9.1 Conclusion

The System has been envisioned for the purpose of reducing the errors that occur in the traditional (manual) attendance taking system. The aim is to automate and make a system that is useful to the organization such as an institute. The efficient and accurate method of attendance in the office environment that can replace the old manual methods. The System is secure enough, reliable and available for use. No need for specialized hardware for installing the system in the office, which can be constructed using a camera and computer. This is particularly useful for detecting fevers and therefore possible infections of COVID-19 and other illnesses. An easy-to-use temperature control system. These systems can be used in the entrance of any organization and only the people with control temperature can entry the rooms.

## 9.2 Future Enhancement

Temperature detection adds an additional layer of personal data collection and processing, which raises concerns about privacy. To address these concerns, future enhancements to facial recognition systems with temperature detection should prioritize privacy and ensure that data is only collected and used for authorized purposes. The systemwill use advanced temperature sensors to detect changes in body temperature, and if the temperature is found to be above a certain threshold, the system will alert the user and provide them with information on recommended next steps. In cases where the temperatureis excessively high, the system will automatically contact a healthcare center to ensure thatthe individual receives prompt medical attention. This system is designed to be highly effective and efficient, and could play a crucial role in detecting and preventing the spreadof infectious diseases ina variety of settings, including schools, workplaces, and public spaces.

# CHAPTER 10
# REFERENCES

[1] J. Deng, J. Guo, N. Xue, and S. Zafeiriou, “Arcface: Additive angular margin loss for deep face recognition,” in IEEE Conference on Computer Vision and Pattern Recognition, 2019.

[2] C. Peng, N. Wang, J. Li, and X. Gao, “Dlface: Deep local descriptor for cross-modality face recognition,” Pattern Recognition, vol. 90, pp. 161–171, 2019

[3] R. He, X. Wu, Z. Sun, and T. Tan, “Wasserstein cnn: Learning invariant features for nir-vis face recognition,” IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 41, no. 7, pp. 1761– 1773, 2018.

[4] L. Song, M. Zhang, X. Wu, and R. He, “Adversarial discriminative heterogeneous face recognition,” in AAAI Conference on Artificial Intelligence, 2018.

[5] H. Zhang, B. S. Riggan, S. Hu, N. J. Short, and V. M. Patel, “Synthesis of high-quality visible faces from polarimetric thermal faces using generative adversarial networks,” International Journal of Computer Vision, vol. 127, no.6-7, pp. 845–862, 2019.

[6] Z. Deng, X. Peng, and Y. Qiao, “Residual compensation networks for heterogeneous face recognition,” in AAAI Conference on Artificial Intelligence, 2019.

[7] X. Wu, L. Song, R. He, and T. Tan, “Coupled deep learning for heterogeneousface recognition,” in AAAI Conference on Artificial Intelligence, 2018.

[8] C. Fu, X. Wu, Y. Hu, H. Huang, and R. He, “Dual variational generation for low shot heterogeneous face recognition,” in Advances in Neural InformationProcessing Systems, 2019.

[9] D. Gong, Z. Li, W. Huang, X. Li, and D. Tao, “Heterogeneous face recognition:A common encoding feature discriminant approach,” IEEE Transactions on Image Processing, vol. 26, no. 5, pp. 2079–2089, 2017.

[10] X. Wu, H. Huang, V. M. Patel, R. He, and Z. Sun, “Disentangled variational representation for heterogeneous face recognition,” in AAAI Conference on Artificial Intelligence, 2019.
[11] B. Cao, N. Wang, J. Li, and X. Gao, “Data augmentation-based joint learning for heterogeneous face recognition,” IEEE Transactions on Neural Networks and Learning Systems, vol. 30, no. 6, pp. 1731–1743, 2018.
[12] T. R. Shaham, T. Dekel, and T. Michaeli, “Singan: Learning a generative modelfrom a single natural image,” in IEEE International Conference on Computer Vision, 2019.
[13] Y. Deng, J. Yang, D. Chen, F. Wen, and X. Tong, “Disentangled and controllable face image generation via 3d imitative-contrastive learning,” in IEEE Conference on Computer Vision and Pattern Recognition, 2020.
[14] Z. Deng, X. Peng, Z. Li, and Y. Qiao, “Mutual component convolutional neural networks for heterogeneous face recognition,” IEEE Transactions on Image Processing, vol. 28, no. 6, pp. 3102–3114, 2019.
[15] C. Peng, N. Wang, J. Li, and X. Gao, “Dlface: Deep local descriptor for cross-modality face recognition,” Pattern Recognition, vol. 90, pp. 161–171, 2019.