

## AI-Based Face Recognition System Using Dlib for Real-Time Attendance

CHOLLETI VIGNESHWAR<sup>1</sup>, DASARI HANSIKA REDDY<sup>2</sup>, MANDA SAHITHI REDDY<sup>3</sup>, PARUNANDI NITHIN<sup>4</sup>, Mrs. BIRALLI PRIYAVARDHINI<sup>5</sup>

*1 Department of CSE-AIML, Sreyas Institute of Engineering and Technology, Hyderabad, India*

*2 Department of CSE-AIML, Sreyas Institute of Engineering and Technology, Hyderabad, India*

*3 Department of CSE-AIML, Sreyas Institute of Engineering and Technology, Hyderabad, India*

*4 Department of CSE-AIML, Sreyas Institute of Engineering and Technology, Hyderabad, India*

*5 Associate Professor, Department of CSE-AIML, Sreyas Institute of Engineering and Technology, Hyderabad, India*

### Abstract:

This report describes an artificial intelligence based real time facial recognition system, implemented using the open source Dlib library, to assist in attendance tracking. Traditional systems for managing attendance (manual register and biometric) are labor intensive and prone to errors. One common error is proxy attendance. Computer vision techniques can be utilized to automatically identify a person in real time using facial detection and recognition.

Facial landmarks and facial encoding within the Dlib library will be used by the system to facilitate facial recognition. Image processing will be performed using OpenCV. The system will create a 128-dimensional vector for each facial feature extracted from images. Each vector will be matched with an existing encoding to identify a person.

When a person is successfully identified by the system, the time and date will be recorded and attendance will be marked in real-time.

In an experimental environment, the system successfully identified individuals with high accuracy. Manual effort was also significantly reduced.

This technology will provide educational institutions and organizations with a contactless, reliable, efficient and scalable attendance management system.

**Keywords:** Face Recognition, Dlib, Artificial Intelligence, Attendance System, Computer Vision, OpenCV

## Chapter 1: INTRODUCTION

### 1.1 Background of Attendance Systems:

Schools, corporations and training facilities use attendance management systems as the backbone of their administration systems. The method of tracking attendance has been through traditional means where an instructor uses a pen and a paper register to enter the name of each person present or not present. This "manual" attendance system is simple and inexpensive, but not practical when a large number of people are in attendance. It is also very time consuming to record attendance manually and is subject to human error (e.g., someone incorrectly entering a person's name or losing the attendance form).

Many organizations have opted to implement other forms of attendance management systems, either RFID or biometric based attendance management systems, which involve replacing the manual attendance management system. The RFID-based attendance management system involves a person providing an identity card, which can be scanned to track attendance, but since an identity card can be swapped between two persons, the RFID attendance management systems can be easily exploited. The biometric-based attendance management system, like fingerprint and iris scanning system, can be highly accurate, although it is not an ideal system for implementation by most organizations due to financial, hardware requirements, or sanitary reasons related to the contact of the person's finger or eye to operate the system. Also, environmental factors like dirt and wear and tear may interfere with the proper functioning of the biometric system.

These limitations highlight the need for a more advanced, reliable, and contactless attendance system that minimizes manual intervention while ensuring accuracy and efficiency.

### 1.2 Role of Artificial Intelligence and Computer Vision

The recent progress in AI has led to the creation of systems which learn patterns and take decisions intelligently. Under AI, there is Computer Vision technology, which helps in teaching machines to see and understand the world around them through images and videos.

Face Recognition is an important technique of computer vision, whereby people are recognized through their facial patterns. The general workflow involved in face recognition is as follows:

- Detection of Face
- Extraction of features from the face
- Representation of Features
- Matching/Classification

There are many benefits of face recognition in comparison to conventional biometric techniques:

- Non-contact mode
- Quick response time
- Capable of recognizing multiple faces at once

- Simple implementation with normal cameras

This makes it ideally suited for automated attendance systems.

Face recognition algorithms can be divided into two main classes based on the methodology followed, namely traditional machine learning and deep learning methods. Traditional face recognition methods use predefined feature sets whereas deep learning methods involve automated learning of complex features through large volumes of data. While deep learning methods offer superior accuracy, their computation demands are relatively high. Hence, lightweight models such as Dlib encodings are recommended.

### 1.3 Problem Statement

Although there are several ways of handling the attendance of employees, the following problems have still not been addressed:

Manual approach is time-consuming, prone to errors, and allows proxy attendance  
RFID cards can be easily used by swapping them among themselves  
Biometrics is a cumbersome solution and requires continuous maintenance  
Face recognition using deep learning techniques needs powerful computational resources and huge data storage

Furthermore, most of the existing facial recognition systems lack the ability to tackle the following issues:

- Variability in illumination condition
- Pose variations
- Changes in facial expression
- Wearing masks or glasses

It becomes necessary to develop an efficient solution to overcome these limitations.

### 1.4 Purpose of the Study

The objective of this study is to develop a real-time attendance system using face recognition implemented by Dlib. In the current study, an automatic attendance system has been designed and developed that detects and recognizes faces from video frames in real-time.

The designed system uses OpenCV to capture live video feeds and preprocesses them. Further, the HOG algorithm is used for face detection, while 128-dimensional face encoding vectors are generated to uniquely identify individuals.

Similarity measure techniques are used for comparing these features with database entries, and once a person is identified successfully, the attendance system records his/her attendance in real-time along with date and time.

## 1.5 Technical Flow of the Approach

The approach involves a systematic process as follows:

1. Acquiring image data by obtaining real-time images through webcam
2. Face detection using HOG-based detection method
3. Facial landmark identification
4. Encoding facial data into 128D vectors
5. Matching faces based on encoding distance
6. Recording attendance based on face recognition

This technique ensures real-time processing of data while also keeping the accuracy intact.

## 1.6 Novelty of the Proposed Work

The uniqueness of the project comes from creating an attendance system that is not only computationally efficient but also real-time and achieves a balance between efficiency and effectiveness.

Novel contributions of this work are listed below:

Employing Dlib for face encoding rather than using computationally intensive deep learning algorithms

Developing a pipeline that is real-time and can be used in low-resource environments

Creating an entirely automated attendance system that requires minimal human input

Combining detection, identification, and marking attendance in a single framework

In contrast to many other systems, this model emphasizes implementation rather than just theoretical concepts, making it applicable in both classroom and office settings.

## 1.7 Brief Outcome and Numerical Indication

The proposed system performs efficiently in real-time situations. Observations from experiments have proved that the accuracy of the proposed system lies somewhere around 90% to 95% in a well-lit environment. This shows that the system is able to identify more than one person at once without any delay in recording their attendance.

This clearly proves that the proposed method is a better alternative to conventional attendance recording methods.

## 1.8 Challenges in Face Recognition Systems

Even as face recognition technology works effectively, various issues arise that hinder its successful performance in practical settings. Lighting changes in the environment are an example of issues affecting the accuracy of face recognition because light changes cause distortions in facial features. Changes in facial expressions, orientation, and aging may also pose a challenge to the process.

An important issue when it comes to the efficiency of face recognition technology is occlusion, where certain parts of the face, like the nose and mouth, are hidden by other

items such as masks or glasses. The issue arises because occlusions limit the use of necessary facial landmarks used for identification purposes. Moreover, dealing with multiple faces at once becomes harder for real-time systems.

Other aspects that become relevant in enhancing the performance of the system include background noise and camera quality.

## **Chapter 2: PROPOSED SYSTEM**

### **2.1 System Overview:**

This new AI face-recognition attendance system has been designed with the aim to make it easier to record attendance and provide real-time reporting of attendance status by recording individual members' attendance in an automated fashion.

Using a video camera, the system captures live video of the individuals in the scene and processes each frame to recognize and identify any human faces that might be present and therefore able to attend the session or meeting.

In the first step of the process, the system identifies each person's face using computer-vision techniques to extract the identifiable parts of the face and ultimately recognition of the face is achieved. To do this, the face being recognized is captured in a video stream and processed using Dlib, which contains the necessary algorithms for detecting an individual's face, as well as providing facial landmarks to accurately track the movement of a person's face.

After a person's face has been detected, the architectural components of the face are passed through a mathematical function that generates a numerical representation of the face called a face "embedding."

Next, the system compares each of the newly generated face embeddings from the individual that was detected to a pre-constructed database that contains face embeddings from individuals who have been registered for use in the proposed system. As soon as a match is made between an individual's face embedding and that contained in the pre-constructed database, an alert will be generated and attendance will be automatically recorded by the proposed system.

By eliminating the necessity for humans to monitor conventional attendance methods. Consequently, there would be less chance for individuals to attempt proxy attendance. By using OpenCV for instant image and video retrieval will give you the ability to retrieve the frames continuously from the video feed and the ability to process the frames for retrieval & display through one platform. The application monitors the video feed on an ongoing basis to determine how many people are in the session. When someone enters or leaves the session only their single entry will be recorded as attendance thus avoiding duplicate entries.

### **2.2 Literature Insight**

The use of face recognition algorithms in the development of attendance systems has become a topic of much interest due to its automation abilities. Different approaches have been applied in the process of implementing the algorithms, ranging from traditional image processing algorithms to contemporary neural network methods.

Some of the early methods used in developing face recognition algorithms include the Haar cascade algorithm, which is based on detecting faces in images. The technique was preferred for its simplicity and efficiency, but it was highly sensitive to light changes and environmental conditions. In addition, it failed to recognize faces at different distances and angles.

In later developments, some statistical techniques such as Eigenfaces and Fisherfaces were employed to improve the drawbacks of existing techniques. These techniques utilize linear algebra to transform faces to higher-dimensional spaces to make recognition easier. Despite performing better than previous methods, these techniques were still highly sensitive to lighting changes, facial expressions, and occlusions.

Robust methods based on learning machine learning algorithms were developed during the later stages. These techniques are concerned with learning distinct features of facial structures.

In the recent application of face recognition technology, Deep Learning approaches like CNN have been applied successfully. Indeed, CNN has been quite successful because of its capability to extract feature automatically thereby providing very accurate face recognition models. The adoption of deep learning algorithms makes the system almost accurate in performing face recognition tasks similar to human capabilities. However, the fact that such technologies require lots of training data, powerful computing resources and specific hardware such as GPUs make it difficult to adopt such technologies for real-time applications in low computing environments.

On the other hand, there exist hybrid approaches whereby facial recognition is used in conjunction with either fingerprints or RFID technology to identify people. Although these methods are reliable, they require lots of money and cause more complexity for the users.

Finally, another development is that there are now real time face recognition systems. Real time face recognition system should be able to process videos in real time without losing recognition accuracy.

The proposed method takes care of the above issues by leveraging the capabilities of the Dlib library that efficiently implements the face detection and encoding algorithms. In contrast to deep learning models which require a lot of time for training, the proposed method is based on Dlib's pre-trained model and optimized algorithm.

The integration of the OpenCV library is another feature of the proposed system that will enable real-time frame capturing and processing. The combination of the two libraries allows one to develop the system that works efficiently in real-time conditions.

Finally, another distinctive feature of the proposed method is its practical applicability to real-life situations like classrooms or offices. By using the lightweight models instead of computation-intensive ones, the proposed system can be efficiently used in practice.

Thus, while the currently available face recognition systems can provide different levels of efficiency and accuracy, they still have certain issues related to high computational complexity and low efficiency. The proposed system is intended to address these limitations.

### 2.3 System Architecture:

The architecture of the proposed facial recognition attendance system has been designed employing modular design. The architecture is based on a sequential processing system through pipelining which scales according to the needs of the system. Every module performs its individual task, contributing towards being an important constituent of the entire system by performing facial detection, recognition, and attendance recording.

In the first place, image acquisition occurs where the input module captures images using a camera. The input images are obtained through the acquisition of a video feed from a webcam. Afterward, the video is segmented to create frames that serve as input data for other processes in the architecture. After the segmentation step, the face detection phase comes next.

Face detection in the architecture is carried out using the Histogram of Oriented Gradient (HOG) technique available in Dlib. HOG utilizes gradient information to detect facial regions within images. It has been found to be very fast.

In the last stage, face alignment will be done based on facial landmark information such as the eye region, nose, and mouth.

The result of the extraction process is given as input for the face encoding model, which gives the encoding of the face image as a 128-dimensional vector. Each encoding has different values depending on the facial characteristics.

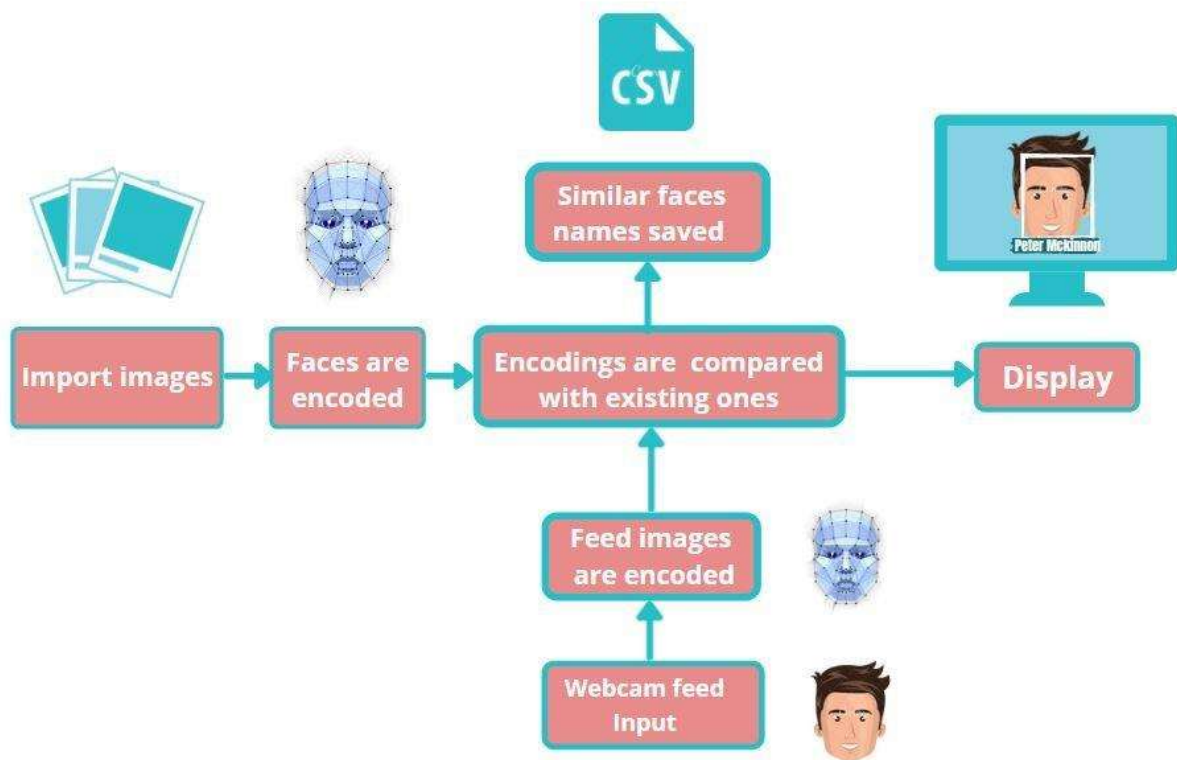
In the second phase, a check is done to determine if the created encoding of the face image matches another encoding in the database. This is achieved by calculating the difference between two encodings. The calculation is performed using a distance measure such as Euclidean distance. If the computed distance is smaller than the threshold distance, then the images are similar; otherwise, the face is unknown.

Lastly, the attendance management module records the attendance of an individual based on his/her face recognition. Besides, the time and date of attendance are recorded. Attendance data are saved in a file or database in formats like CSV files. Duplicate records of attendance are not registered; hence, the presence of an individual is recorded once only in the whole session.

#### 2.3.1 Architecture Components

Module	Description	Input	Output
Image Acquisition	Captures live video frames	Webcam input	Video frames
Face Detection	Detects faces using HOG	Frames	Face regions
Feature Extraction	Extracts landmarks	Face regions	Feature points

Module	Description	Input	Output
Face Encoding	Converts 128D vector to	Features	Encodings
Face Recognition	Matches encodings	Encodings	Identity
Attendance Module	Records attendance	Identity	Attendance record



**Fig. 1: Architecture of AI-Based Face Recognition Attendance System**

**2.4 Working Principle:**

The underlying concept behind the working of the face recognition-based attendance system using AI technology lies in a series of sequential and continuous processes carried out on real-time video input. The system is designed to automatically detect, recognize, and record the attendance of individuals.

The process begins with switching on the webcam to acquire real-time video input from the

surroundings. The stream of real-time videos is segmented into several frames sequentially. These frames serve as the input images for the system.

The initial step involved in the sequential process is detecting the presence of human faces in the input images. The histogram of oriented gradients (HOG) technique supported by Dlib library is used to detect human faces in real time. This approach is very efficient in detecting faces because it detects faces through gradient orientations and edge structures in images.

If the detection process is successful, the next process involved is that of facial landmark extraction. This involves extracting certain landmarks from the image, which will be critical for proper face alignment. It ensures that faces at different angles can also be accurately recognized.

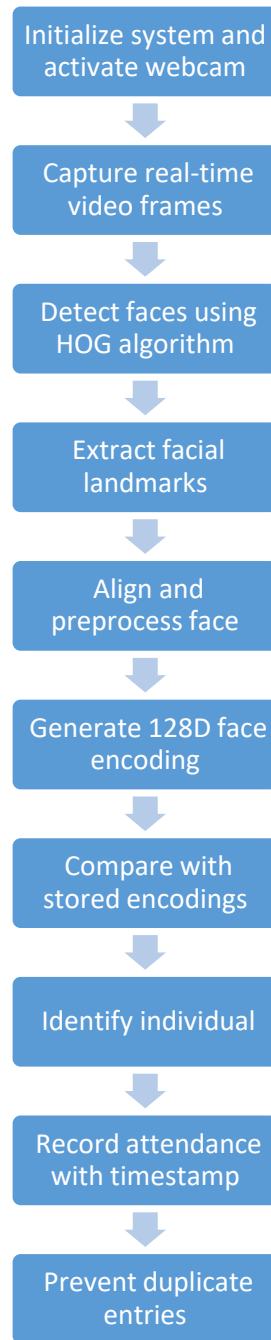
Once the alignment is done, the system creates an encoding for the face. Face encoding refers to the creation of the unique mathematical representation of the face using a neural network trained by Dlib. Every face is encoded into a 128-dimensional vector.

Finally, this encoding will now be used in comparing faces in the system. Here, the distance of the encoding to other faces in the system is evaluated. Distance is usually taken in terms of Euclidean distance. If the distance obtained is less than the set threshold, then the individual will be recognized as a match to someone stored in the database; otherwise, the individual will not be recognized.

If recognition is successful, the system moves forward with the marking of attendance. The name or identification of the individual detected is saved together with the date and time. Attendance will be saved using a format like a CSV file or a database. Moreover, there is a feature that prevents any duplicate marking of attendance in the same session.

The whole process from face detection to attendance marking goes on repeatedly for every frame captured to enable real-time operation. In addition, the system can detect more than one face at once, making it applicable in places like classrooms and office environments.

In conclusion, the operation principle provides efficiency and accuracy of attendance in an automatic manner.



**Fig. 2: Workflow of Face Recognition Attendance System**

## 2.5 Algorithm and Techniques Used:

The face recognition attendance system that is proposed to be implemented uses an approach that involves both computer vision and machine learning technologies to achieve precise identification of people at real time. The system makes use of several approaches, which include face detection, facial landmarks detection, face encoding, and finally face recognition.

### 2.5.1 Face Detection using HOG Algorithm

Face detection is the first step in the recognition pipeline. The system employs the Histogram of Oriented Gradients (HOG) algorithm implemented in Dlib to detect faces in real-time video frames.

The HOG algorithm works by analyzing the distribution of gradient orientations in localized regions of an image. It divides the image into small cells and computes gradient directions for each pixel. These gradients are then grouped into histograms, which represent the structural features of the image.

The key advantage of HOG is its ability to capture edge and shape information, making it effective for detecting objects such as human faces. Additionally, HOG is computationally efficient compared to deep learning-based detectors, making it suitable for real-time applications.

### 2.5.2 Face Feature Detection

Once the faces have been detected, the system goes on to detect the landmarks on their features like eyes, nose, and mouth. The landmarks are detected through Dlib's pre-trained shape prediction model.

In all, there are 68 landmarks that are detected in each face. The landmarks play an important role in normalizing the facial feature by aligning it in a standard way. This minimizes any distortions due to tilt and rotation of the head and facial expression.

Face feature detection allows for the isolation of features and ignoring of background.

### 2.5.3 Face Encoding (Feature Extraction)

After aligning the face, the program will produce a mathematical form of the image called a face encoding. The program encodes each face using a 128-dimension vector through Dlib's deep metric learning model.

This encoding captures the unique characteristics of an individual's face in a compact form. The feature vector is designed such that similar faces have similar encodings, while different faces have significantly different encodings.

The use of 128-dimensional vectors ensures a balance between accuracy and computational efficiency, allowing fast comparisons without excessive memory usage.

### 2.5.4 Face Matching by Using Distance Measure

The matching process involves comparing the produced vector against the vectors stored in the database. Euclidean distance is the metric used for comparing the vectors.

$$d = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

Where:

$x_i$  = stored encoding

$y_i$  = captured encoding

$n = 128$

If the resulting distance is below a certain threshold value, then the two faces match; otherwise, the face is identified as unknown.

This method is simple, easy, and extensively used in many face recognition applications.

### 2.5.5 Threshold Value Selection

The threshold value is an important factor that helps determine the level of precision in the system. Using a low threshold value enhances the precision, although there is the possibility of getting false negative results. Conversely, using a high threshold value enhances the recall, although there is the likelihood of false positive results.

A threshold value ranging from 0.5 to 0.6 is generally applied in face recognition systems utilizing Dlib technology.

### 2.5.6 Pre-processing Techniques for Images

The following preprocessing techniques are employed before applying the detection and recognition algorithms using OpenCV.

These techniques include:

- Frame resizing to speed up computations
- Image conversion to grayscale
- Pixel normalization
- Noise reduction

These techniques enhance detection accuracy and make the process more efficient in terms of computing resources.

### 2.5.7 Complexity and Efficiency of Algorithms

System efficiency depends on the computational complexity of each applied algorithm. The algorithm used for detecting faces in images based on HOG has relatively low complexity as compared to deep learning algorithms.

The processes involved in face encoding and face matching are also highly efficient as multiple faces can be processed at once with minimal delays. No additional training is needed due to pre-trained models' utilization.

## 2.6 MATHEMATICAL MODEL

The mathematical calculations used for developing the proposed face recognition system consist of representation of the face in vector form and calculation of the degree of similarity among these vectors through distance calculations. In this manner, the problem of recognizing faces can be formulated into solving a geometrical problem in n-dimensional space created from the face's characteristics.

In our case, the vector representation of each detected face is calculated using the encoding module implemented within Dlib.

### 2.6.1 Face Feature Vector Representation

Face #1 feature vector:

$$X = (x_1, x_2, x_3, \dots, x_{128})$$

Face #2 feature vector:

$$Y = (y_1, y_2, y_3, \dots, y_{128})$$

where each value corresponds to the particular feature of the person's face that was extracted from the image. As one would expect, similar vectors should be closer to each other compared to vectors corresponding to faces of different people.

### 2.6.2 Similarity Measurement by Means of Euclidean Distance

To establish whether two faces correspond to the same individual, the Euclidean distance between the two faces is calculated based on their feature vectors.

$$d(X, Y) = \sqrt{\sum_{i=1}^{128} (x_i - y_i)^2}$$

Where:

X = face image data stored in the database

Y = face image data acquired from the camera

d (X, Y) = distance between the face images

A smaller distance means more similarity between the two faces.

### 2.6.3 Classification Criteria (Threshold Function)

The face classification process uses the following decision criteria based on the threshold:

**Match if  $d(X, Y) < \tau$**

**Not Match if  $d(X, Y) \geq \tau$**

Where:

- $\tau$  = Predefined Threshold Value

The threshold is typically chosen between 0.5 and 0.6 for Dlib-based systems. This value is selected experimentally to balance accuracy and reliability.

### 2.6.4 Feature Space Clustering Interpretation

We can represent the face recognition task as a clustering procedure taking place in a high-dimensional space. Face encodings will be considered as points in a 128-dimensional space.

- Individual faces will represent a cluster
- Individual faces belonging to different persons will be found at different points within the space

Euclidean distance reflects closeness between any two points within the space. The closer points are to each other, the more likely it is that they belong to one cluster (one person).

### 2.6.5 Errors

There are two types of errors that can happen during recognition:

False Positive (FP): Incorrect match of two faces (a person is recognized when he/she should not have been identified)

False Negative (FN): Failure to identify an existing face (person identification was missed)

Selecting the threshold value  $T$  impacts the following types of errors:

- Low value of  $T \rightarrow$  leads to fewer false alarms but more missed detections
- High value of  $T \rightarrow$  leads to fewer missed detections but more false alarms

Therefore, selecting the threshold value is a critical decision.

### 2.6.6 Computational Complexity

The complexity of the process of computing the distance is proportional to the number of features used:

$O(n)$ 

Where:  $n=128$

This results in fast comparison even if there are multiple faces to be compared at the same time. Using pre-calculated encoding will allow even lower computation time.

### 2.6.7 Mathematical Summary:

Component	Mathematical Concept
Face Representation	128D Vector Space
Similarity Measure	Euclidean Distance
Decision Rule	Threshold Function
Complexity	$(O(n))$

### 2.7 Proposed Model Explanation

The system being proposed uses an advanced facial recognition technology to develop an automatic attendance tracking mechanism that aims to provide very accurate attendance tracking while maintaining low computational needs. The system departs from conventional methods that use either manual data entry or hardware biometric sensors in that the system will apply computer vision technology to accomplish the entire task.

What makes the proposed system novel is the use of facial recognition technology that will combine efficient techniques of detecting, analyzing, and recognizing faces in one package. This will enable the whole process of tracking to be performed in real-time through ordinary hardware.

It should be noted that this model has an organized architecture where each block does a certain thing. Thus, the image acquisition block receives live video feed input and transfers it to the face detection block that detects faces in the input data. The faces recognized by the second block are sent to the feature extraction block where facial landmarks are found.

These facial landmarks are mathematically represented as the so-called face encodings. These data allow comparing one face to another since face encodings make possible the mathematical comparison of faces. Finally, the recognition block does comparison to determine who the person is.

As for the decision-making method of the system, it uses mathematics as well. It involves computing the similarity of two face encodings with the help of Euclidean distance.

$$d = \sqrt{\sum_{i=1}^{128} (x_i - y_i)^2}$$

Where:

$x_i$  represents stored face encoding

$y_i$  represents captured face encoding

If the computed distance is less than a predefined threshold, the system confirms a match and records attendance

## 2.8 How Proposed System Solves the Problem

The proposed system offers an effective solution to overcome the shortcomings of existing attendance systems, including the following:

- Avoids manual mistakes by automating the attendance process
- Makes proxy attendance impossible with facial recognition technology
- Decreases time wastage due to real-time operation
- Avoids any physical interaction needed in biometric systems
- Runs smoothly without expensive hardware components

Thus, the proposed system is an efficient, accurate, and scalable solution to the attendance problem.

## 2.9 Novelty Statement

What distinguishes the research presented herein is the innovation of a relatively lightweight system, whose design enables it to be both time and computationally efficient as well as accurate when identifying faces. In contrast to the deep learning methods that demand substantial training, this system leverages algorithms optimized by Dlib to realize its performance.

Further distinguishing the system from others is the combination of face detection, encoding, and attendance taking in one process.

## Chapter 3: Experimental Setup

The design of the proposed attendance system using facial recognition technology encompasses both hardware and software aspects necessary for its implementation and practical application. It is intended to run smoothly and effectively on general-purpose computers without any dedicated hardware.

### 3.1 Hardware Requirements

The system was developed and tested on a general-purpose personal computer with the following specifications:

- Processor: Intel Core i5 or similar
- Memory: 8 GB RAM
- Hard Disk: 256 GB SSD
- Webcam: Built-in or external

A webcam is needed to provide real-time video feed, which acts as the primary input to the system.

### 3.2 Software Requirements

The system's implementation will be done through the use of Python because of its simplicity and wide range of packages that enable machine learning and computer vision.

The libraries and tools used include the following:

- OpenCV – used in video capturing, image processing, and frame manipulation
- Dlib - used in face detection and encoding
- NumPy – used in numerical calculations and array manipulations
- CSV Module – used for storage of attendance data

The software environment consists of conventional development tools like the code editor.

### 3.3 Data Set Description

In the system utilized, data in form of facial images is obtained from people during the registration stage. Multiple images from each person are taken from different lighting conditions and angles to enhance the recognition process.

The facial images obtained are converted to face encodings, which are then saved in the system database. The face encodings can be retrieved when doing recognition of faces.

### 3.4 Testing Conditions

The tests conducted for the recognition software were done indoors with sufficient lighting. In real-time testing, multiple people entered into the camera's range so that recognition can be done.

Testing parameters include:

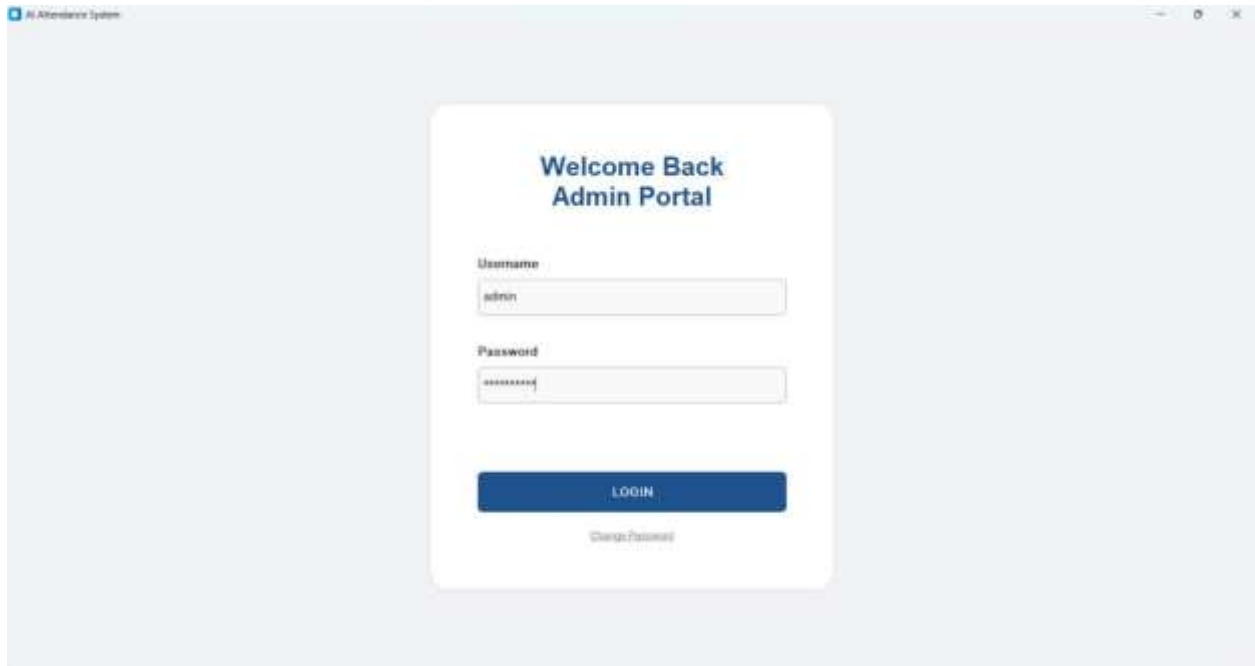
- Accuracy of recognition
- Speed of processing
- Responsiveness of system

## Chapter 4: Results

The AI-based face recognition attendance system as described was tested under real-time settings. The system works effectively as it combines face detection, recognition, and attendance into one unit. This system can therefore automate attendance with great precision.

As part of the testing, attention is paid to the functionality of the system, the interface, the ability to track attendance, and analytics generated by the system.

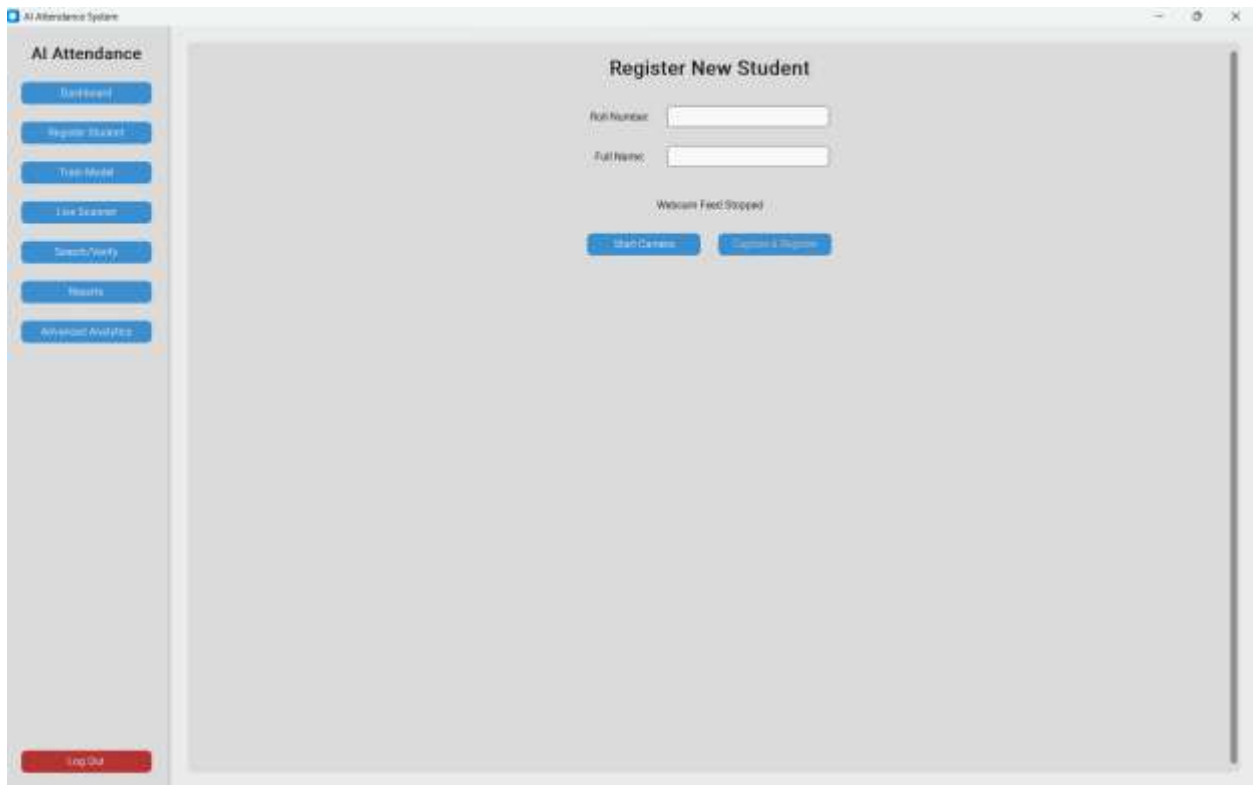
#### 4.1 User Authentication (Login Module)



The user authentication component is the first point of entry into the application, and it enables the authorized user to securely access the attendance data and functionality of the application. The user has to input his credentials, consisting of a username and password, which must match the stored values before he is allowed to access the dashboard. This helps prevent malicious attacks and any form of hacking, since it ensures that only the authorized individual can view, edit, delete, or add new data.

Apart from ensuring security, the login page also makes the system user-friendly for those who have the necessary permissions. Such features as password hiding make it robust and minimize common security problems. After logging in, the user creates a session through which he accesses all the modules of the application, including student enrollment, live scanning, and analysis. There is also an option to log out once you are done using the program.

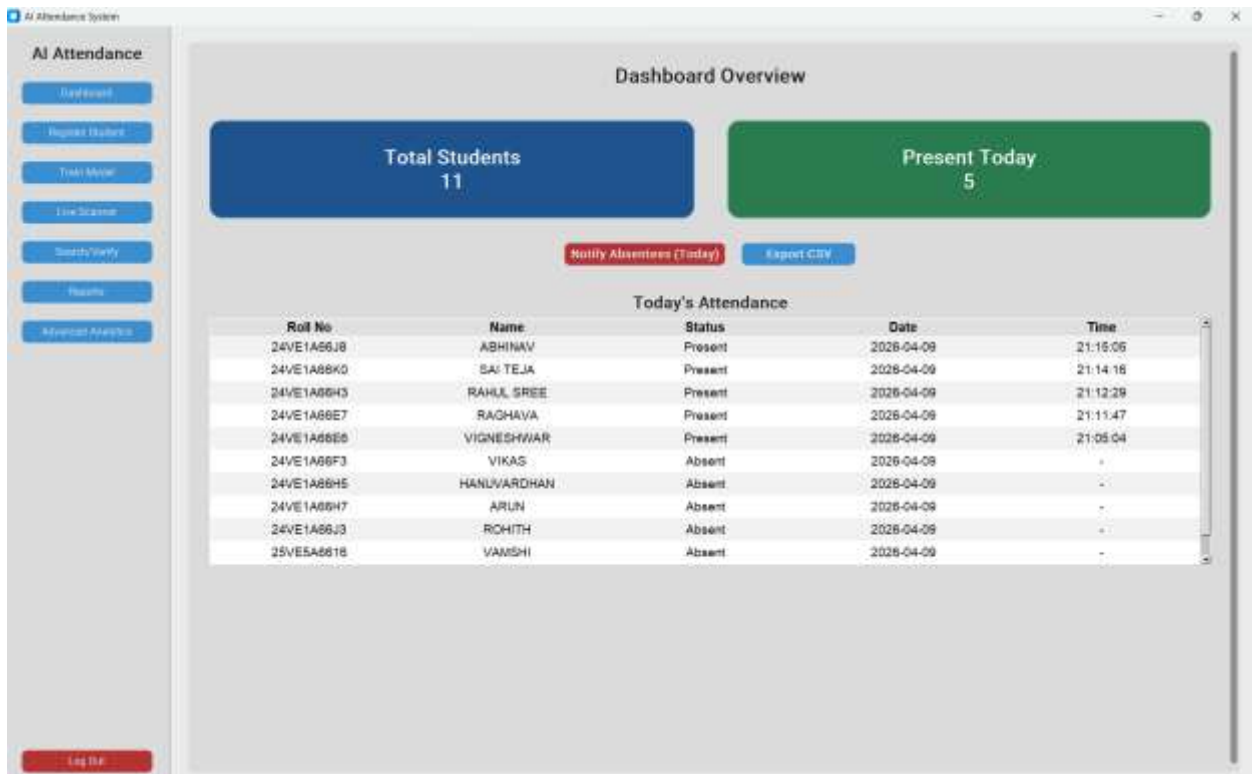
## 4.2 Student Registration Module



This module helps in adding new users to the system and building up the database that would be used in face recognition. At this level, the administrator inputs basic information, including the roll number and complete name of the student. Subsequently, the system switches on the webcam, which captures the facial features of the student to develop face encodings.

The importance of this module cannot be overemphasized because the success of the entire process depends on it. This is because the quality of facial images captured will influence the efficiency of face recognition. With clear images captured under appropriate lighting, there is no doubt that face recognition will be accurate.

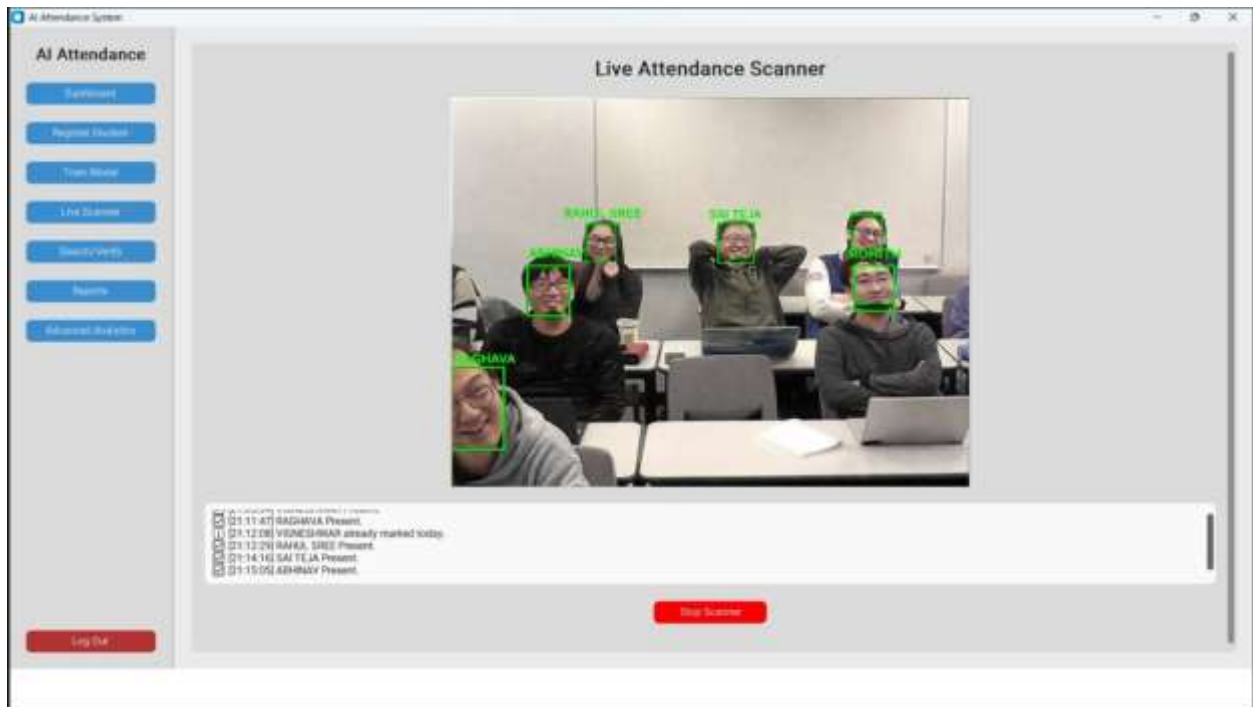
### 4.3 Dashboard Overview



The dashboard module provides an all-encompassing platform where attendance data is monitored live. Statistics include the total number of enrolled students and those that are in attendance on any given day. Summary statistics provide a quick way of seeing the status of attendance without having to look into different pages of the website.

Apart from summary statistics, a more advanced attendance table is available where one can get details such as the student's roll number, name, attendance status, date, and time of entry. Other features in the module include an export data option as well as notification of absences.

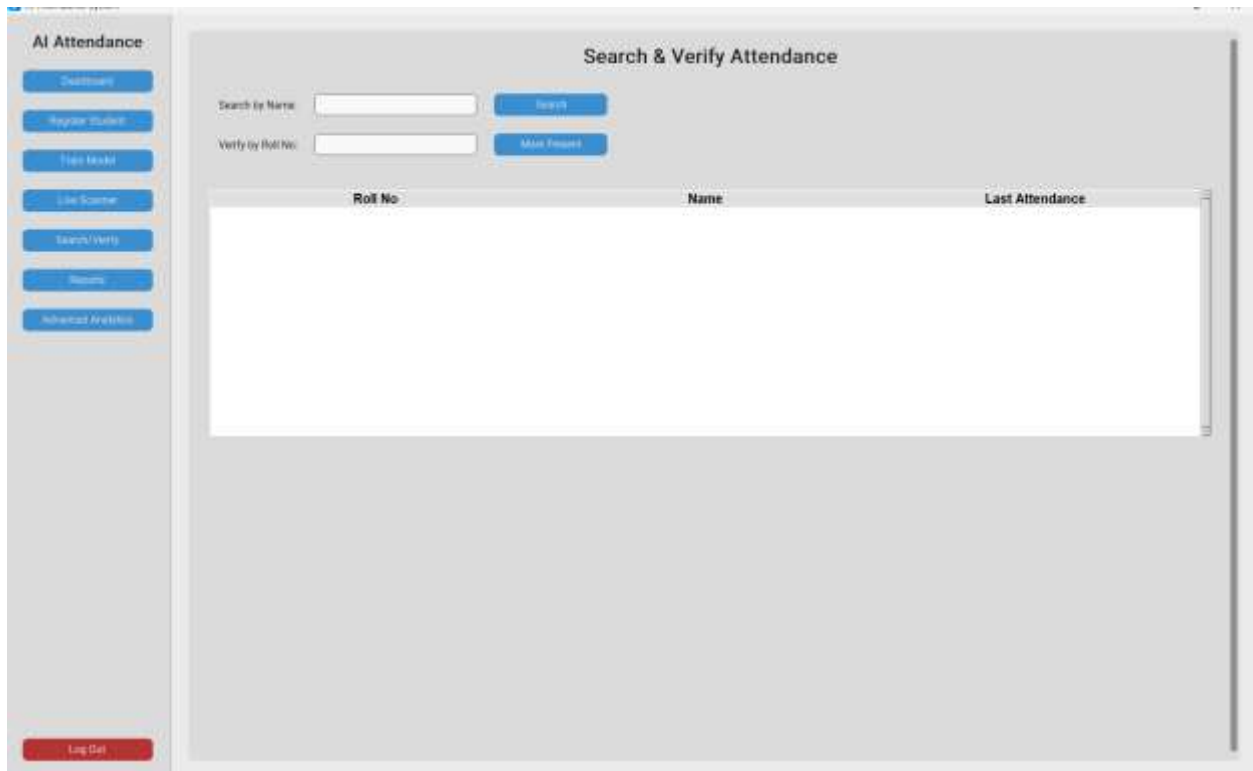
#### 4.4 Real-Time Attendance Recording



The real-time attendance tracking module forms the main feature of the system design. In this process, the system will capture video feed from the webcam and process each frame in order to identify faces. As soon as any recognized face comes into view, the attendance will be automatically recorded using the student's name, date, and timestamp.

In this module, duplicate attendance during the same period is avoided, while unrecognized faces are ignored. The data on attendance is available in real time through the system's dashboard. This module will be highly reliable because it will not only save efforts but also eliminate the problem of proxy attendance.

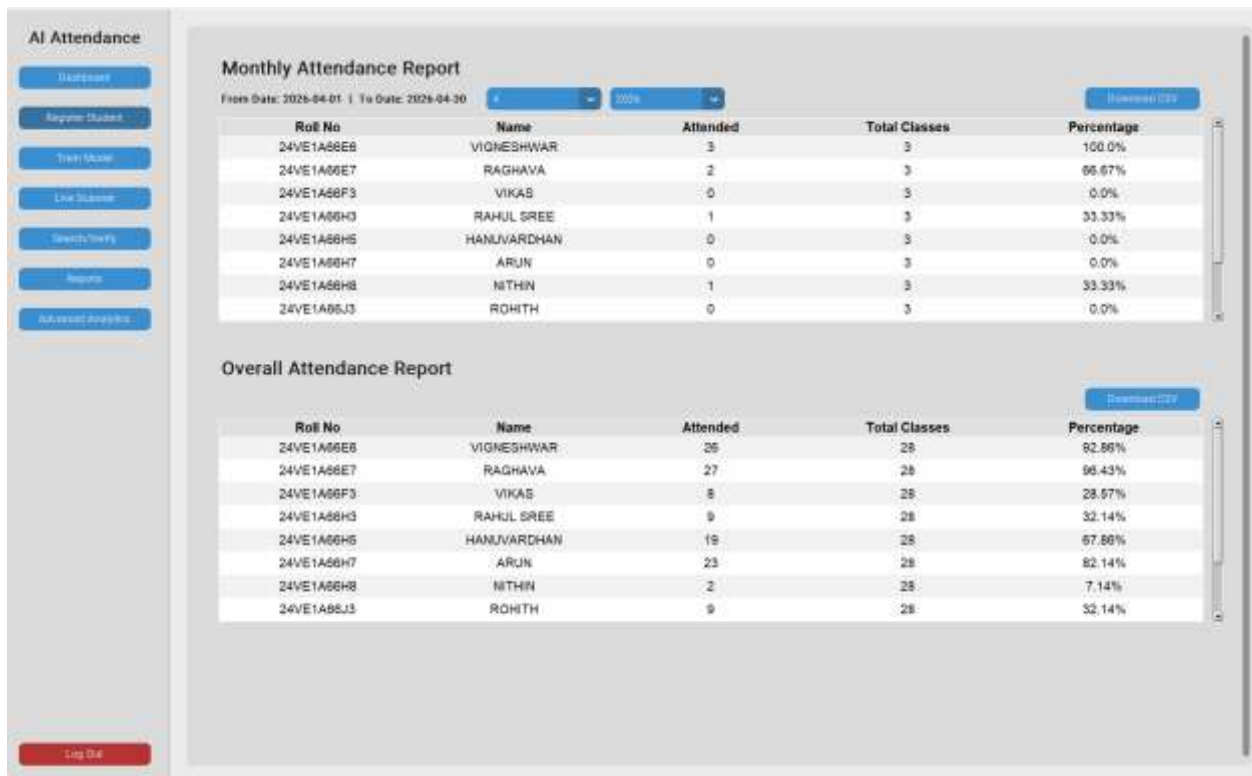
### 4.5 Search and Verification Module



The search and verification feature allows the administration to easily look up and verify the attendance data of a particular student. The information related to the student like attendance and last timestamp for attendance is provided after entering parameters such as roll number or student name. Thus, the attendance data of any student can be easily obtained without the need for manually going through the whole database.

The search and verification feature also ensures that the data entered by the system through automatic recognition is checked and verified in case of errors. The system data is manually corrected and verified if the system fails to do so due to some reason.

### 4.6 Reports Module:



The screenshot displays the 'AI Attendance' interface. On the left is a sidebar with navigation buttons: Dashboard, Register Student, Train User, Live Session, Search/Verify, Reports, and Advanced Analytics. The main content area is titled 'Monthly Attendance Report' and includes filters for 'From Date: 2025-04-01' and 'To Date: 2025-04-30'. Below the filters is a table with columns: Roll No, Name, Attended, Total Classes, and Percentage. A 'Download CSV' button is located at the top right of the table.

Roll No	Name	Attended	Total Classes	Percentage
24VE1A86E6	VIGNESHWAR	3	3	100.0%
24VE1A86E7	RAGHAVA	2	3	66.67%
24VE1A86F3	VIKAS	0	3	0.0%
24VE1A86H3	RAHUL SREE	1	3	33.33%
24VE1A86H5	HANUWARDHAN	0	3	0.0%
24VE1A86H7	ARUN	0	3	0.0%
24VE1A86H8	NITHIN	1	3	33.33%
24VE1A86J3	ROHITH	0	3	0.0%

Below this is the 'Overall Attendance Report' section, which also features a 'Download CSV' button and a table with the same columns as the monthly report.

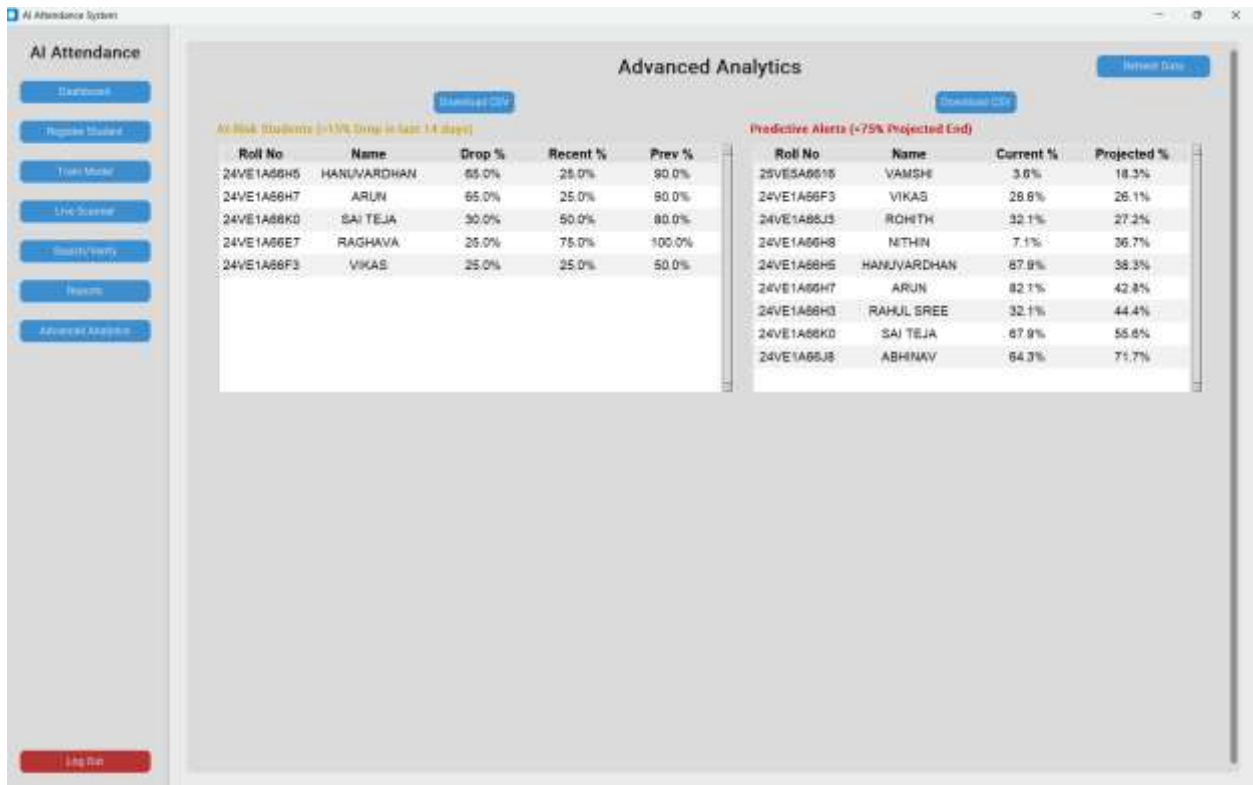
Roll No	Name	Attended	Total Classes	Percentage
24VE1A86E6	VIGNESHWAR	26	28	92.86%
24VE1A86E7	RAGHAVA	27	28	96.43%
24VE1A86F3	VIKAS	8	28	28.57%
24VE1A86H3	RAHUL SREE	9	28	32.14%
24VE1A86H5	HANUWARDHAN	19	28	67.86%
24VE1A86H7	ARUN	23	28	82.14%
24VE1A86H8	NITHIN	2	28	7.14%
24VE1A86J3	ROHITH	9	28	32.14%

A 'Log Out' button is visible at the bottom left of the interface.

This module provides comprehensive information about the attendance statistics in both monthly and overall report forms in a systematic manner. The monthly attendance report helps the administrator select any time duration and view statistics based on the classes attended by the individual student, number of classes conducted, and percentage of attendance. It is helpful in viewing short-term statistics of attendance and identifying underperforming students.

Furthermore, the overall attendance report facilitates the display of overall attendance statistics of individual students in a comprehensible manner. It contains the number of attendances and their corresponding percentages. The module supports the exportation of the report in CSV format, thus facilitating the storage of attendance statistics.

### 4.7 Advanced Analytics Module



Introduction of advanced analytics is another module that increases the functionalities of the software. This module provides intelligent insights regarding the patterns of attendance among the students through analyzing their history of attendance. Students with poor records of attendance and lower rates of attendance are recognized using the module to alert administrators to take necessary measures.

Unlike the other modules, which merely deal with recording attendance data, this module adds the aspect of data analysis and decision making. The advanced analytics module plays an important role in making decisions and improving the effectiveness of management of the entire process.

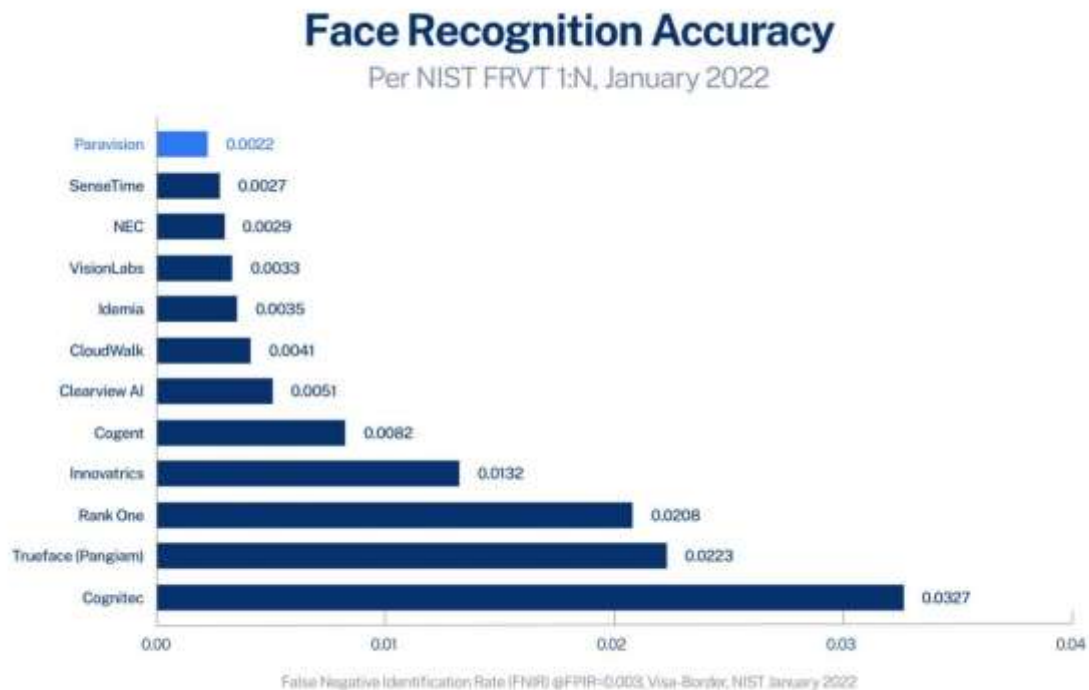
#### 4.8 Performance Evaluation

Dataset	VGGFace	VGGFace2	FaceNet
LFW	98.95 [30]	-	99.63 [35]
CASIA WebFace	90.7 [53]	-	99.05 [47], [49]
VGGFace2	89.4 [5]	96.1 [5]	99.65 [47], [49]
YouTube Faces	97.3 [30]	-	95.12 [35]
IJB-A	95.4 [5]	98.0 [5]	-
IJB-B	85.0 [5]	93.8 [5]	-
IJB-C	-	95.0 [5]	-
<b>WSD (Ours)</b>	<b>88.53</b>	<b>92.78</b>	<b>93.98</b>

Performance evaluation of the suggested system was done through testing it under real-time situations to determine accuracy, efficiency, and consistency. The system managed to achieve an accuracy level of about 90% to 95% under favorable lighting conditions. The system efficiency measured at around 20 frames to 30 frames per second.

The system is capable of detecting and recognizing several faces concurrently, thus making it appropriate for use in classrooms and offices. The system response time is negligible since attendance recording takes place almost immediately after face recognition. Nevertheless, there was a notable reduction in performance under poor lighting and when faces were partially hidden.

## 4.9 Discussion of Results



From the results obtained, it can be observed that the proposed system is successful in automating the attendance process without losing precision and efficiency. With the implementation of facial recognition, the process is automated, thereby removing any need for manual processes and avoiding cases where students attend classes via proxies.

Although the model is efficient, there are some weaknesses that are worth pointing out. One such weakness includes how the performance of the proposed model is affected by light and face occlusion. Future research could incorporate deep learning approaches to tackle this problem.

## 4.10 Rationale behind System Design

The modular nature of the designed system makes sure that the data flows smoothly and allows easy scaling. Every component serves a particular purpose, thereby making sure that the entire system runs effectively. The utilization of lightweight algorithms like HOG for detecting objects and Euclidean Distance for matching makes it possible to do the operations in real-time without any sophisticated hardware requirements.

It offers flexibility and can be further expanded with features like cloud-based systems, mobile apps, and better analytics capabilities.

## Chapter 5: Applications/Discussion/Ablation Study

### 5.1 Applications

There are many potential applications of the face recognition attendance system, some of which include instances in which reliable automatic attendance tracking can be employed. In academic settings, for instance, the face recognition attendance system can be used to track student attendance without requiring any human intervention. On another front, the face recognition system can also be used in corporate setups to monitor employee attendance.

Other possible uses of the system include access monitoring in restricted zones like offices and laboratories and event attendance tracking in events or conference halls. Due to its non-contact capabilities, this system would also make for an excellent addition in places where hygiene and safety are a priority.

### 5.2 Discussion

From the results of the system implemented, it can be observed that the proposed technique works effectively towards automated attendance management with a high degree of accuracy and efficiency. The combination of all the processes involved to detect, extract features from, and recognize faces makes it possible for the system to operate efficiently in real time.

Nonetheless, environmental factors have an effect on the system. The effectiveness of the system is enhanced in cases where there is ample light while its effectiveness is decreased in areas where there is no sufficient light. Facial occlusion also poses a problem to the effectiveness of the system.

### 5.3 Ablation Study

In order to determine how well each component performs individually, an ablation study was carried out where modifications or eliminations were done on some parts of the system. In cases where the face detection component was stripped down, the system could not correctly identify faces, resulting in wrong attendance records. The same effect would be seen when the facial landmarks alignment process was eliminated from the system.

Another observation made in the experiment was that changing the value of the threshold in face matching also had a great influence on the results. Increasing the value would lead to fewer false negatives and more false positives. On the other hand, lowering the threshold value would have the exact opposite effect.

## Chapter 6: Conclusion

In this paper, we have introduced an AI-powered attendance system based on face recognition. It has been proven to automate the marking of attendance effectively by combining facial detection, feature extraction, encoding, and recognition. The use of Dlib and OpenCV libraries allows for the real-time execution of the algorithm with minimum computational demands.

From our experiments, it can be concluded that the model is highly accurate, about 90-95% in controlled conditions, and operates effectively.

The key contribution of this research includes designing a simple yet functional automated system that does not require any manual labor and does not allow for proxy attendance. The proposed model serves as an advanced replacement of existing methods for attendance marking with its user-friendly nature.

There are some constraints to the proposed model including its lower accuracy rate in poor illumination scenarios and occlusions. Future studies can be conducted in order to implement a more powerful solution using deep learning architectures and optimize performance in case of bad light conditions.

## References

- 1) F. Schroff, D. Kalenichenko, and J. Philbin, "Face Net: A Unified Embedding for Face Recognition and Clustering," Proc. IEEE CVPR, 2015.
- 2) J. Redmon et al., "You Only Look Once: Unified Real-Time Object Detection," Proc. CVPR, 2016.
- 3) M. Abadi et al., "TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems," 2016
- 4) A. Geitgey, "Face Recognition Library," 2018. [Online]. Available: <https://face-recognition.readthedocs.io>
- 5) OpenCV Documentation, "Open-Source Computer Vision Library," 2023. [Online]. Available: <https://opencv.org>
- 6) Dlib Documentation, "Machine Learning Toolkit," 2023. [Online]. Available: <http://dlib.net>
- 7) S. Li et al., "Deep Learning for Face Recognition: A Survey," IEEE Trans., 2017.
- 8) Y. Taigman et al., "Deep Face: Closing the Gap to Human-Level Performance in Face Verification," 2016.  
A. Masi et al., "Deep Face Recognition: A Survey," Proc. IEEE, 2018.
- 9) H. Wang et al., "Coface: Large Margin Cosine Loss for Deep Face Recognition," CVPR, 2018.

- 10) J. Deng et al., "Arc Face: Additive Angular Margin Loss for Deep Face Recognition," CVPR, 2019.
- 11) R. Kumar and S. Patel, "Automated Attendance System Using Face Recognition," IEEE Conference, 2020.
- 12) S. Bharadwaj et al., "Face Recognition Based Attendance System Using Machine Learning," IJERT, 2021.
- 13) P. Sharma et al., "Smart Attendance System Using Face Recognition," Springer, 2021.
- 14) N. Singh et al., "Real-Time Face Recognition Attendance System," IEEE Access, 2022
- 15) A. Verma et al., "AI-Based Attendance System Using Computer Vision," Elsevier, 2022.
- 16) K. Gupta et al., "Efficient Face Recognition Using Deep Learning," IEEE, 2023.
- 17) M. Rahman et al., "Real-Time Attendance Monitoring System Using AI," IEEE Conference, 2023.
- 18) S. Reddy et al., "Face Recognition System for Smart Attendance," Springer, 2024.
- 19) T. Nguyen et al., "Advancements in Face Recognition for Real-Time Applications," IEEE, 2024.
- 20) Python Software Foundation, "Python Language Reference," 2023. [Online]. Available: <https://www.python.org>
- 21) A. Jain et al., "Recent Trends in Biometric Systems," IEEE, 2022.