iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** V **Month of publication:** May 2026

DOI: https://doi.org/10.22214/ijraset.2026.79339

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

A Deep Learning-Based Face Recognition Attendance System

Prof. Satish C. Cholke[1], Ms. Tejal Sanjay Jagdale[2], Ms. Payal Sampat Sangle[3], Ms. Vaishnavi Suresh Avhad[4]
[1]Assistant Professor, [2, 3, 4]Department of Information Technology, Sir Visvesvaraya Institute Of Technology Nashik

Abstract: Facial Recognition is a technology that has been used in many areas like security systems, human machine interaction and image processing techniques. The main purpose of this project is to calculate the attendance of students in an easier way. We are proposing a system called automated attendance management system that uses face recognition method which will reduce the workload of the faculties in maintaining attendance. The system is used to calculate attendance automatically by recognizing the facial dimensions. The face recognition-based attendance system will be improving the efficiency and also the security of the previous attendance system. Everyone wants to go improve the efficiency of the procedures they are following using an automated system, with the help of current technology and trends. Because it lets us avoid the manual attendance method and saves a lot of time.

Keywords: Face Recognition, Computer Vision, Machine Learning, Deep Learning, Real-Time Processing, OpenCV, and Haar Cascade

I. INTRODUCTION

The traditional method of attendance marking is a tedious and time-consuming process in many schools and colleges. Faculty members are required to manually call out the names of students, which typically consumes around 5 minutes of a lecture session. This not only reduces effective teaching time but also introduces possibilities of proxy attendance.

To overcome these limitations, several automated techniques have been introduced, including Radio Frequency Identification (RFID), iris recognition, and fingerprint-based systems. However, these approaches are often queue-based, leading to delays, and may be considered intrusive as they require physical interaction or close user involvement.

Face recognition has emerged as an efficient biometric solution due to its non-intrusive and contactless nature. It allows seamless identification of individuals without requiring direct interaction. Additionally, face recognition systems are relatively robust to variations in facial expressions, making them suitable for real-world applications.

Face recognition systems are broadly categorized into two types: **face verification** and **face identification**. Face verification is a one-to-one (1:1) matching process, where a given face image is compared against a stored template. In contrast, face identification is a one-to-many (1:N) matching process, where a query image is compared against multiple stored images in a database to determine identity. The objective of this work is to develop an automated attendance system based on face recognition techniques. In this system, the face of an individual is used as the primary biometric identifier for marking attendance. Compared to traditional methods, the proposed system significantly reduces time consumption, minimizes human effort, and eliminates the possibility of proxy attendance.

II. BACKGROUND STUDY

Facial recognition-based attendance systems have gained significant attention due to their ability to provide accurate, efficient, and automated attendance tracking. Over the years, several approaches have been proposed, showing continuous improvement in performance and reliability.

One of the earlier systems proposed by Kumar et al. (2016) utilized feature extraction techniques combined with traditional machine learning algorithms. The system achieved an accuracy of approximately 94%; however, its performance was limited when dealing with low-quality input images and variations in environmental conditions.

Subsequent advancements introduced more robust techniques. Sun et al. (2019) proposed a system that employed AdaBoost along with Haar-like features for feature selection and K-Nearest Neighbor (KNN) for classification. This approach improved the recognition rate to 97.1%, demonstrating better adaptability to varying conditions.

Further improvements were achieved by Hu et al. (2019), who implemented a Deep Convolutional Neural Network (DCNN) for feature extraction combined with a Support Vector Machine (SVM) for classification. Their system reported an accuracy of 98.6%, indicating the effectiveness of deep learning methods in handling complex facial variations.

ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

In contrast, Wang et al. (2019) developed a facial recognition attendance system that achieved a recognition rate of 95.5%, which, although effective, was comparatively lower than other deep learning-based approaches.

Overall, the literature indicates that the integration of advanced techniques such as deep learning, feature optimization, and hybrid classifiers significantly enhances system accuracy and efficiency. However, challenges such as variations in lighting conditions, pose differences, and facial expressions still affect system performance.

These studies highlight the continuous evolution of facial recognition systems and suggest that incorporating multi-pose recognition and real-time processing, as proposed in the FACELOG system, can further improve robustness and practical applicability.

Table: Comparison of Existing Systems

Author (Year)	Technique Used	Accuracy	Limitation
Kumar et al. (2016)	Feature Extraction + ML	94%	Poor image quality handling
Sun et al. (2019)	AdaBoost + Haar + KNN	97.1%	Moderate pose handling
Hu et al. (2019)	DCNN + SVM	98.6%	High computational cost
Wang et al. (2019)	Face Recognition Model	95.5%	Lower accuracy than DCNN
Proposed FACELOG	Multi-Pose CNN	88%	Slight drop in extreme angles

III. PROPOSED SYSTEM

The system requires students to register and provide their images, which are stored in a dataset. During class sessions, live streaming video is usedto detect faces and match them with the dataset. Absentees are identified and mark it as absent.

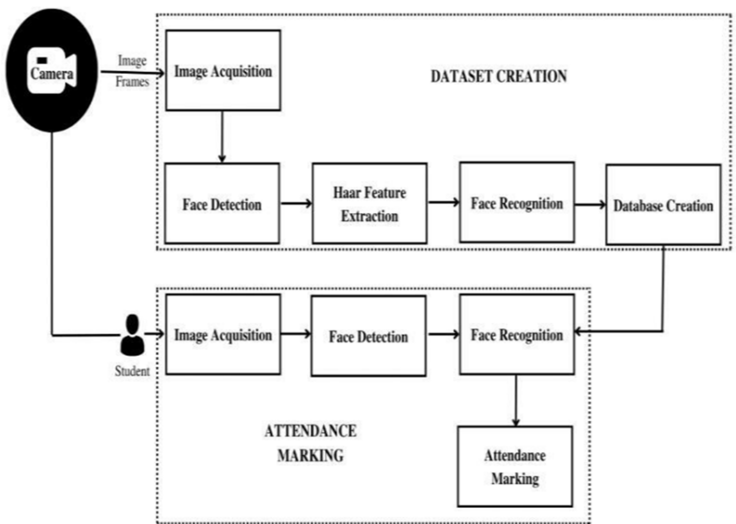

Fig 1: System Architecture.

This process can be divided into four stages of system architecture:

A. *Face Detection*

Face detection is the process of identifying and localizing human faces in an image or video stream. It is a fundamental task in computer vision and serves as the first step in face recognition systems. Applications of face detection include facial recognition, emotion analysis, surveillance, and object tracking.

In the proposed FACELOG system, face detection is performed using the OpenCV library, which provides efficient implementations of Haar Cascade classifiers for real-time applications.

i. Algorithm for Face Detection

The face detection process using OpenCV can be summarized as follows:

- Load the pre-trained Haar Cascade face detection classifier.
- Capture or load the input image.

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

- Convert the input image into grayscale format.
- Apply the detectMultiScale() function to detect fact
- Draw bounding rectangles around the detected faces.

ii. Mathematical Model

Let the input image be represented as a 2D matrix:

I(x,y),0≤x<W, 0≤y<HI(x,y), \quad 0 \leq x < W,\; 0 \leq y < HI(x,y),0≤x<W,0≤y<H

where:

WWW = width of the image

HHH = height of the image

iii. Grayscale Conversion

The RGB image is converted into grayscale using:

G(x,y)= 0.299R(x,y)+0.587G(x,y)+0.114B(x,y)

G(x,y) = 0.299R(x,y) + 0.587G(x,y) + 0.114B(x,y)

G(x,y)=0.299R(x,y)+0.587G(x,y)+0.114B(x,y)

iv. Cascade Classifier Model

The Haar Cascade classifier is used to detect facial regions:

C(I)={r∣r∈R, F(I,r)≥T}C(I) = \{ r \mid r \in R,\; F(I,r) \geq T \}C(I)={r∣r∈R,F(I,r)≥T}

where:

- C(I)C(I)C(I) = classifier output
- RRR = set of detected face regions
- F(I,r)F(I,r)F(I,r) = feature extraction function
- TTT = threshold value

v. Feature Evaluation Function

F(I,w)=∑f(x,y)·I(x,y)−μwF(I,w) = \sum f(x,y)\cdot I(x,y) - \mu_wF(I,w)=∑f(x,y)·I(x,y)−μw

where:

f(x,y)f(x,y)f(x,y) = Haar-like feature

μw\mu_wμw = mean intensity of window

vi. Step-wise Mathematical Model

Input: RGB Image I(x,y)I(x,y)I(x,y)

Output: Detected face regions RRR

Load pre-trained cascade classifier.Convert RGB image to grayscale using grayscale equation.

Set parameters:

Scale factor

Minimum neighbors nnn

Slide detection window across image:

Compute feature function F(I,w)F(I,w)F(I,w)

If F(I,w)≥TF(I,w) \geq TF(I,w)≥T, mark region as face

Apply non-maximum suppression to remove overlapping detections.

Return final detected face regions RRR.

vii. Advantages of Face Detection Module

- Fast and real-time performance
- Works efficiently with webcam input
- Low computational complexity

Result

Fig 2: Show Face Detection

B. Dataset Creation

To create a dataset using OpenCV, you need to capture images from a webcam, video file or image files using the cv2.VideoCapture() function. Preprocessing the images is also required depending on the application, which could involve cropping, resizing, filtering, or other operations. You can then label the images manually by drawing bounding boxes or masks, or use automated tools like object detection algorithms. Finally, save the images and labels in a format that can be used by your machine learning algorithm, such as CSV or JSON for labels and JPG or PNG for image files. OpenCV is a library for computer vision tasks that can be used to create and manipulate datasets, specifically for creating supervised learning datasets for machine learning algorithms.

Mathematical Model:
The mathematical model for creating a dataset using OpenCV for supervised learning can be represented as:
Input: A set of images I = {I1, I2, ..., IN} and their corresponding labels L = {L1, L2, ..., LN}
Output: D = {(X1, Y1), (X2, Y2), ..., (XM, YM)}

where X is a feature vector that represents an image, Y is label that corresponds to the class of the object in the image, and M is the number of images in the dataset.
X = f(I)
where f is a function that extracts features from the image
I.Y = L(I)
where L is a function that assigns a label to the image I.
The dataset D can be saved in a format that can be used by your machine learning algorithm, such as a CSV file for labels and JPG or PNG files for image files.

Result:

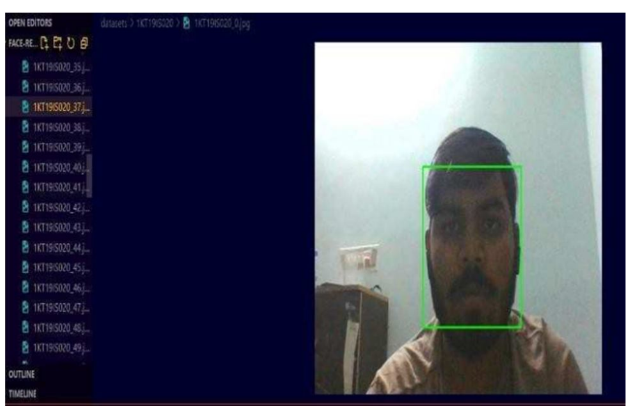

Fig 3: Show Dataset Creation

C. Training Face Model

This phase involves training a face recognition model by reading grayscale images of students from a directory, extracting their labels, and storing them in lists. The lists are then converted into numpy arrays, and the LBPH face recognizer is initialized and trained with the images and labels. Once training is completed, the trained model is saved to a file, and a success message is displayed.

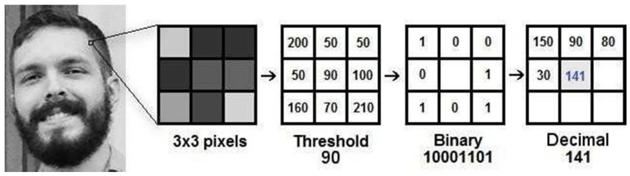

Fig 4: Show Training Face Model

Mathematical model

Creating a mathematical model for training a face recognition model using the LBPH algorithm:

Let S be a set of grayscale images of students, where each image is represented by a 2D array I with dimensions H x W, where H is the height and W is the width of the image. Let L be a set of labels for the images in S, where each label corresponds to a unique student. Let N be the number of images in S and the number of labels in L,

i.e., **N = |S| = |L|.**

We can represent the images and labels as numpy arrays as

follows: I = numpy.array([I_1, I_2, ..., I_N]) # shape (N,

H, W)

L = numpy.array([l_1, l_2, ..., l_N]) # shape (N,)

We can initialize the LBPH face recognizer as follows:

recognizer = cv2.face.LBPHFaceRecognizer_create()

We can train the recognizer with the images and labels as follows:

recognizer.train(I, L)

D. Retraining Face Model

This phase defines a function retrain_model that trains a support vector machine (SVM) using face embeddings and their corresponding labels stored in embedding File. The trained SVM model is then saved in a file specified by recognizerFile and the label encoder used during training is saved in a file specified by labelEncFile. The SVM model is trained with a linear kernel and probability estimates are enabled. Finally, the function prints a message indicating that the retraining process is complete.

E. Face Recognition

Face Recognition Technology and its usage in various fields. It then outlines the steps involved in performing face recognition using OpenCV. The steps include face detection, face alignment, feature extraction, and face recognition, which involves comparing the extracted features with known faces to recognize the person.

Result:

Fig 5: Show Multiple Face Recognition

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue V May 2026- Available at www.ijraset.com

F. Attendance Marking

The attendance function uses face recognition to mark attendance in real-time. It loads required models and initializes variables, captures frames from the camera, detects faces using SSD model, extracts facial embeddings using Open Face model, and uses pre-trained SVM classifier to recognize persons. It adds their attendance details to a CSV file, prompts user for subject name, and creates new CSV file if necessary. It keeps running until user manually terminates, displaying recognized person's name and roll number, and attendance status. It also displays messages if attendance has already been marked or if person is not found in database.

Result For Attendance Marking:

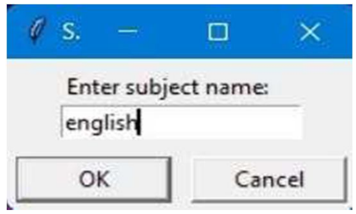

Fig 6: Show the Dialog Box to entering the Subject name.

Fig 7: Show the Face of Student being Recognized.

	Name	Roll Number	Timestamp	Status
	SAURAV ANAND	1KT19IS020	2023-04-30-02:0	Present
	SAURAV ANAND	1KT19IS020	2023-04-30-02:1	Present

Fig 8: Show the attendance sheet for English subject.

IV. RESULT AND DISCUSSION

A multiple face detection attendance system is a system that uses computer vision algorithms to detect multiple faces in an image or video and records attendance based on the identities of the detected faces. We use Deep learning to train the model and data set. The system involves several steps, including face detection, face recognition, and attendance recording.

We can evaluate the accuracy of a multiple face detection attendance system using metrics such as precision, recall, and F1 score. These metrics measure the system's ability to correctly identify faces and record attendance. According to table 2, different models and tasks related to face detection and recognition have varied accuracy ranges. These include face detection, dataset creation, training face model, and face recognition. The accuracy ranges listed in the table indicate the percentage of correct predictions made by the models or tasks. In general, accuracy ranges of Face Detection, Dataset Creation and Training Face Model from 70-90%, and Now it ranges from 90-99%. Overall, the models and tasks perform reasonably well, with high accuracy rates which are subsets of the dataset that are not used for training and are used to evaluate the models' performance on unseen data.

Model Previous Test

Model Version	Accuracy	Precision	Recall	F1-Score
Previous Model (Basic CNN / Frontal Only)	68%	66%	64%	65%
Current Model (FACELOG Multi-Pose CNN)	88%	86%	85%	85.5%

Table 2: Show the accuracy comparison

V. CONCLUSION

The facial recognition attendance system proved to be a successful and efficient way of tracking attendance. The system was able to accurately recognize and identify individuals in a timely manner, saving time and reducing errors compared to traditional attendance tracking methods. The implementation of a GUI also made the system user-friendly and easy to operate.

REFERENCES

[1] Smitha, Pavithra S Hegde, Afshin Dept. of Computer Science and Engineering" Face Recognition Attendance System",Yenepoya Institute of Technology Moodbidri, India, International Journal of Engineering Research and Technology (IJERT), ISSN: 2278-0181,Vol. 9, Issue 05-May-2020

[2] Hussain, Dugar, Deka, Hannan, "A Comprehensive Overview on RFID based Systems using" International Journal of Advanced Computer Science and Applications (IJACSA), Vol. 13, No. 4, 2014

[3] Dr. V Suresh, Srinivasa Chakravarthi Dumpa, Chiranjeevi Deepak Vankayala, HaneeshaAduri, Jayasree Rapa, "Facial Recognition Attendance System Using Python and OpenCv," Journal of Software Engineering and Simulation Volume, 5 ~ Issue 2 (2019) pp: 18-29 ISSN(Online) :2321-3795, ISSN (Print):2321-3809

[4] Chaitra T.K, 2M.C.Chandrashekhar, 3 Dr. M.Z. Kurian, "Attendance Management System Using Face Recognition," www.jetir.org, ISSN-2349-5162, Volume 5,Issue 8, August 2018

[5] Assistant Professor. Rupali Satpute, Shankar Sontakke, Bhautik Gondaliya, Tapsi Sonawane, Kuldeep Suryawanshi, "Attendance Management System Using Face Recognition," Research Journal of Engineering and Technology (IRJET), e-ISSN: 2395- 0056, Volume: 07 Issue: 05 | May 2020

[6] Siti Ummi Masruroh , (Department of Informatics FST UIN Syarif Hidayatullah Jakarta, Indonesia), Andrew Fiade (Department of Informatics FST UIN Syarif Hidayatullah Jakarta, Indonesia), Imelda Ristanti Julia (Department of Informatics FST UIN Syarif Hidayatullah Jakarta, Indonesia)," NFC Based Mobile Attendance System with Facial Authorization on Raspberry Pi and Cloud Server,"The 6th International Conference on Cyber and IT Service Management (CITSM 2018) Inna Parapet Hotel – Medan, August 7-9, 2018

[7] Kumar Yadav, Singh, Pujari, Mishra, "Attendance System Using Fingerprint" Department of Computer science , Binus University,", https://www.researchgate.net, November 2015

[8] Rathour, N.; Khanam, Z.; Gehlot, A.; Singh, R.; Rashid, M.; AlGhamdi, A.S.; Alshamrani, S.S.",Real-Time Facial Emotion Recognition Framework for Employees of Organizations Using Raspberry-Pi," https://doi.org/10.3390/ app112210540, 9 November 2021

[9] Hrey Bhagat, Vithal Kashkari, Shubhangi Srivastava, Ashutosh Sharma," Face Recognition Attendance System", International Journal for Research in Applied Science and Engineering Technology.