# ATTENDANCE CAPTURE SYSTEM USING FACE

**MANGALAPALLI MANISHA[1], ATHUKURI DHARANI[2], MANDALA CHANDANA[3]**
**KALLU MOUNIKA[4], SUPERVISOR Dr M MURUGESAN[5]**
**Professor**
**Depatment of CSE**
**ANURAG ENGINEERING COLLEGE**
**AUTONOMOUS**
**(Affiliated to JNTU-Hyderabad,Aprroved by AICTE-New Delhi)**
**ANANTHAGIRI (V) (M), SURYAPETA (D), TELANGANA-508206**

***Abstract****: Keeping up with regular activities while also keeping track of everyone's attendance might be difficult. Traditional roll call requires time and effort and may easily be circumvented by a proxy student. The following method uses facial recognition to keep track of students' attendance. The administrator already has a record of the students' daily attendance broken down by course. Students who are recognised by the system's face detection and recognition technology at the designated time and for the designated topic are automatically tagged as present, and their attendance records are updated to reflect the new information. To create this system, we relied on deep learning methods, namely the histogram of oriented gradient approach for detecting faces in photos and the aforementioned deep learning method for computing and comparing feature facial of students to recognise them. Multiple faces may be recognised simultaneously by our technology in real time.*

## I. INTRODUCTION

A reliable attendance tracking system is a need for any institution that has to keep track of its pupils. some use a piece of paper and a pen, calling out names during class time, while others have embraced biometrics systems like fingerprint readers, RFID card readers, or iris scanners to register attendance. The time-consuming process of manually calling pupils' names is the norm. Students in the RFID card system are given individual cards that include their unique identifiers, however these cards may be misplaced or stolen and used to take phoney attendance. Fingerprint, iris, and voice recognition biometrics, on the other hand, each have their own set of limitations and aren't fool proof. Smart attendance management systems make use of facial recognition for this reason. Compared to other methods, face recognition is both more precise and quicker, and it also decreases the possibility of proxy attendance. Face recognition offers passive identification, meaning the individual being recognised doesn't have to do anything to be confirmed.

Face recognition is a two-stage process that begins with the detection of faces and continues with the matching of those faces to a pre-existing database. Multiple approaches for finding and identifying faces have been developed. There are two main types of facial recognition systems: those that rely on the face's outward look (appearance-based) and those that rely on the face's underlying geometric characteristics (feature-based), such as the eyes, nose, brows, and cheeks.

A high-quality camera is needed to collect the students' photographs, and the detection is handled by a histogram of oriented gradient in our system, which employs a face recognition technique to address the shortcomings of the current system via the use of machine learning. Also, deep learning for performance recognition.

An IPC (Inter Personal Communication) bridge is built to facilitate communication between the frontend (client side) and the backend (server side), both of which are

based on GUIs built using electron JS. The camera's output is sent into an analytic system, where it is matched to a database of individual student photos to determine who was there.

## II. LITERATURE SURVEY

Their system provides optimal answers. The students' attendance is logged automatically by the camera that snaps a picture of each one as they enter the classroom in Kawaguchi, Japan, according to a novel approach called continuous monitoring. Since just two cameras are needed to monitor the classroom wall, the system's construction is straightforward. The first is a capturing camera that takes pictures of students in class, while the second is a sensor camera that finds out where each student is sitting so that they may be photographed. In order to perfect attendance, the system compares photos taken with a camera to a database of previously taken photos and scanned faces. Facial recognition-based automatic attendance systems are a kind of biometric system that can accurately and reliably keep track of who is in the room at any given time. In an ideal system, this data would be sent to a server where it could be used to track students' attendance and update a database.

Compared to manual attendance systems and other biometric attendance systems, automated systems are more dependable, rigid, and efficient, which in turn improves teacher and student production and time management.

## III SYSTEM ANALYSIS

**Existing Technology:** A biometric system that uses facial recognition technology to keep track of who's in the room at any one time is called an automatic attendance system. In a perfect world, this data would be sent to a server where it could be used to track students' attendance and update relevant records. Compared to manual attendance systems and other biometric attendance systems, automated attendance systems are more dependable, strict, and efficient, which in turn improves teacher and student production and time management.

**Proposed System**: PRISMA, which stands for Preferred Reporting Items for Systematic Reviews and Meta Analyses, is the methodology used for this project. The PRISMA approach allows us to collect high-quality focus reports from a wide range of studies. This methodology is useful for creating reviews of face recognition attendance systems.

**System Analysis :**

The viability of the project is examined at this stage, and a business proposal outlining the project's broad strokes and providing some ballpark cost estimates is presented. The suggested system's viability is to be investigated during system analysis. This is necessary to guarantee that the suggested solution won't cost the business too much. Understanding the system's primary needs is crucial for conducting a feasible analysis.

Three key considerations involved in the feasibility analysis are :

1.ECONOMICAL FEASIBILITY

2.TECHNICAL FEASIBILITY

3.SOCIAL FEASIBILITY

1.Economical Feasibility: This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus, the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

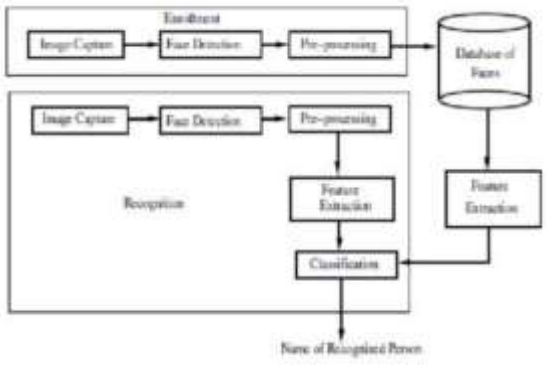
2.Technical Feasibility: This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The

developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

3.Social Feasibility: The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

## IV SYSTEM DESIGN

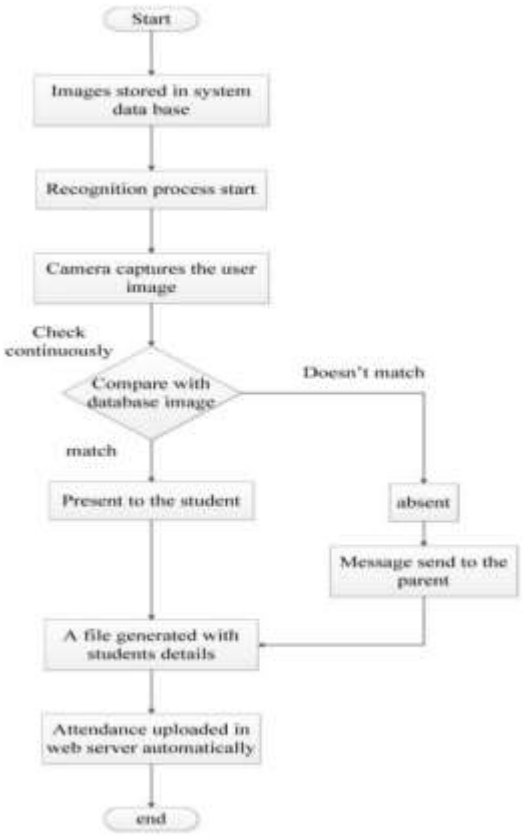
### SYSTEM ARCHITECTURE




**Har cascade Algorithm :**

Har cascade is a feature-based object detection algorithm to detect objects from images. A cascade function is trained on lots of positive and negative images for detection. The algorithm does not require extensive computation and can run in real-time.

**Flow Chart:**




**UML DIAGRAMS:** UML stands for Unified Modelling Language. UML is a standardized general-purpose modelling language in the field of object-oriented software engineering. The standard is managed, and was created by, the Object Management Group. The goal is for UML to become a common language for creating models of object-oriented computer software. In its current form UML is comprised of two major components: a Meta-model and a notation. In the future, some form of method or process may also be added to; or associated with, UML. The Unified Modelling Language is a standard language for specifying, Visualization, Constructing and documenting the artifacts of software system, as well as for business modelling and other non-software systems. The UML represents a collection of best engineering practices that have proven successful in the modelling of large and complex systems. The UML is a very important part of developing object-oriented software and the software development process. The UML uses

mostly graphical notations to express the design of software projects.

**Goals:** The Primary goals in the design of the UML are as follows:

1. Provide users a ready-to-use, expressive visual modelling Language so that they can develop and exchange meaningful models.
2. Provide extendibility and specialization mechanisms to extend the core concepts.
3. Be independent of particular programming languages and development process.
4. Provide a formal basis for understanding the modelling language.
5. Encourage the growth of tools market.
6. Support higher level development concepts such as collaborations, frameworks, patterns and components.
7. Integrate best practices.

**Use Case Diagram:** A use case diagram in the Unified Modelling Language (UML) is a type of behavioural diagram defined by and created from a Use-case analysis. Its purpose is to present a graphical overview of the functionality provided by a system in terms of actors, their goals (represented as use cases), and any dependencies between those use cases. The main purpose of a use case diagram is to show what system functions are performed for which actor. Roles of the actors in the system can be depicted.


## V IMPLEMENTATION

**System Implementation:** Three basic steps are used for implementing the proposed system.

1. Detect and extract the face image and save the details in an xml file.
2. Calculate eigenvalue and eigenvector for that image.
3. Recognize the face and match it according to eigenvalues and eigenvectors stored in xml file
4. Store the name of the face displayed in Microsoft Access Database.

**Face detection and extraction:** The function open CAM is called for starting the camera to capture the image Next, Extract Face is used to extract the frontal face in a video frame. The Extract Face uses OpenCV Har cascade method to load the face.xml (harcascade file) as the classifier. The output of the classifier is in binary form and outputs "1" if face is found and "0" otherwise. After the face is detected, it is clipped into a grayscale image of 50x50 pixels which is done by "Add Face" button in the face recognition module.

Learning and Training Face Images: The function Learn performs the PCA algorithm on training datasets.

The Learn implementation involves four steps:

1. Load the training data.
2. Find a subspace by doing PCA on training data.
3. Project the training faces onto the PCA subspace.
4. Save the training information such as :

a. Eigen values
b. Eigen vectors
c. Average training face image
d. Projected face image
e. Person ID numbers

The function calcite Objects is an inbuilt function in OpenCV and it is called for performing PCA subspace calculation .To do PCA the dataset must be centred. Inters of face images this means finding the average image which is nothing but an image in which each pixel contains the average value for that pixel across all face images in the training set. The data set can be centred by subtracting the average pixel values of faces from each training image.

This is what is happening inside cv Cal Eigen Objects . Now we have calculated subspace for PCA using cv Calc Eigen Objects. We can convert the training images into points in this sub space. This process is called “projecting” the image. This is done with the help of OpenCV function called cv Eigen Decomposition. Then the data for learned faces is stored in an xml file using OpenCV’s built in function.

**Recognition and Identification:** In OpenCV, we have a function called recognize which will implement the recognition of the eigenfaces. It has three steps in which two of them are already done that is loading the face image and projecting onto the subspace. The function load face image array ,loads the face image into face image array, as listed in xml file. Here, the number of face images is stored in a separate textbox named” Number .of faces in the scene” and the number of faces, is automatically counted according to the number of faces detected. The global variables such as no Eigen’s, training image average and eigen vector array should be loaded. OpenCV locates and loads each data value in xml file by the name given. The final step in the recognition face is projecting each test image onto PCA subspace and locating the closest projected image. eigen Decomposition is used to project the test image. Before doing final step, we should pass no Eigen’s (number of eigenvalues)and Eigen Vector Array(array of eigenvectors).This time instead of training image we pass a test image as the first parameter. The output from cv Eigen Decomposition is stored in a local variable. The system uses OpenCV matrix for storing the projected test face.

## IV RESULTS

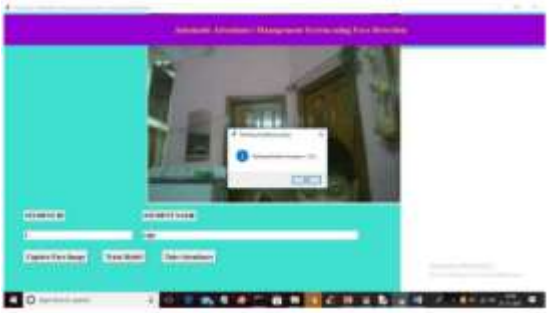
**Capture Face Image:** To run project double click on ‘run.bat’ file to get below screen

In above screen enter student id and name and then click on ‘Capture Face Image’ button to capture face like below screen.

**Train Model:**

In above screen face is detected and now click on ‘Train Model’ button to perform training like below screen

**Training Model Accuracy:**



In above screen training completed with accuracy 75% and then any time click on ‘Take Attendance’ to recognized student and mark attendance.

**Take Attendance:**

In above screen in text area, we can see student identified as Raju. Similarly, you can capture N student and take attendance.

## VII CONCLUSION

Attendance Capture System is designed to solve the issues of existing manual systems. We have used face recognition concept to mark the attendance of student and make the system better. The system performs satisfactory in different poses and variations. In future this system needs, be improved because this system sometimes fails to recognize students from some distance, also we have some processing limitation, working with a system of high processing may result even better performance of this system.

## VIII FUTURE ENHANCEMENT

The future scope of the project can be integrated with the hardware components for example GSM through which a monthly list of the defaulter students can be sent to the mentor. Additionally, an application can be developed to help students to maintain a track of their attendance. It can also be used in offices where a large group of employees sit in a hall and their attendance will be marked automatically by capturing a video but for this the accuracy of the recognition needs to be improved.

## REFERENCES

1.Kar, Nirmala, et al. "Study of implementing automated attendance system using face recognition technique." International Journal of computer and communication engineering 1.2 (2012): 100.

2. Roshan Tharanga, J. G., et al. "Smart attendance using real time face recognition (smart fr)." Department of Electronic and Computer Engineering, Sri Lanka Institute of Information Technology (SLIIT), Malabe, Sri Lanka (2013)

3. Selvi, K. Senthamil, P. Chitrakala, and A. Antony Jenitha. "Face recognition-based attendance marking system." Corresponding Author: S. Rajkumar*, Email: Rajkumar @ Gamal. com (2014).

4. Joseph, Jomon, and K. P. Zacharia. "Automatic attendance management system using face recognition." International Journal of Science and Research (IJSR) 2.11 (2013): 327- 330.

5. Patil, Ajinkya, and Mrudang Shukla. "Implementation of classroom attendance system based on face recognition in class." International Journal of Advances in Engineering & Technology 7.3 (2014): 974.

6. Karnalim, Oscar, et al. "Face-face at classroom environment: Dataset and exploration." 2018 Eighth International Conference on Image Processing Theory, Tools and Applications (IPTA). IEEE, 2018.

7. Mian, Ajmal. "Realtime face detection and tracking using a single pan, tilt, zoom camera." 2008 23rd International Conference Image and Vision Computing New Zealand. IEEE, 2008.

8. Mehta, Preeti, and Pankaj Tomar. "An Efficient Attendance Management System based on Face Recognition using MATLAB and Raspberry Pi 2." International Journal of Engineering Technology Science and Research IJETSR 3.5 (2016): 71-78