

Face Recognition Based Attendance System Using Supervised Machine Learning

Vasudh (Assistant Professor)
Faculty of Computer Science and Engineering
Lovely Professional University
Punjab, India
Vasudha.23636@lpu.co.in

Sunil Kumar
(Student)
Lovely Professional University
Punjab, India
sunildahiya773@gmail.com

Abstract: Face Recognition has been a keen topic of Artificial Intelligence. Face Recognition works similar to our brain, how could you recognize a mango, apple or a human face within seconds. Whenever we met a new person, brain doesn't recognize his/her face at that time because your brain database is not updated with that face. Now, when the person starts talking with you, your brain will start extracting the unique features like size of nose, eyes etc. When we discuss about facial recognition or expression recognition, we always surprised that how a machine can predict the faces of human being or know their current emotion without knowing the internal implementation. There are certain algorithms available which do the same thing for us. We have used the similar algorithm in our project that makes the face recognition easier i.e. LBPH (Local Binary Pattern Histograms). LBPH algorithm is very efficient, fast and provide good accuracy even in the low light as compared to others algorithm.

Keywords: Face Recognition, LBPH Algorithm, MySQL database, Face Detection, Attendance System

I. INTRODUCTION

Face Recognition or Detection is a biometric system that is used for uniquely identifying a person based upon their unique characteristics. Unique characteristics may consist of their behavioral or physical characteristics. Take an

example, like every person has their different ways of speaking, voice texture. We present the system where the faculty/teacher is able to mark the attendance of the student based upon face recognition. This process reduces the error rate, provides more accuracy and time saving method. Maintaining attendance on the regular basis is very important for monitor the performance of the students as well as the faculty. Manual attendance is very time consuming process and not so efficient because a student can mark the attendance of another student by just signing on the respective attendance sheet. Let's see the phases associated with the system^[1]

1. Preparation of Data Set: The camera captures the sample images of the person about 50 to 100 images and stores into the database to train the model.
2. Face Detection: OpenCV provides the separate Cascade Classifier that will use to separate facial part from rest of the body.
3. Feature Extraction: In this system, different parts of the face will be extracted like eye, nose and mouth etc. Based upon

that a training model will be ready that will be used for face recognition

4. Face Recognition: Now, Camera capture the image of the person and then the captured image features will be compared with the existing image features. If the features matched then the attendance of that particular student will be marked accordingly^[2]

II. PROFILE OF THE PROBLEM

The problem is that manually taking attendance is very inconvenient task and time consuming. Traditionally, attendance is taken by the faculty by just passing the attendance sheet among the students. Verifying the student by calling his name where the strength of students are more than 60 is a very difficult process.

1. Manually taking attendance is time consuming and more hectic.
2. A single person can't do it alone, more number of persons are required for that
3. Percentage calculation manually takes lots of time and there might be a chance of error
4. Sometimes there might be happen to hire a extra person to take care of attendance system (salary of person, pen and paper, their expenditure)

What's new in the system to be developed
Our system is completely biometric. Here the user doesn't need to interact with the system, Face biometric is used to find the unique characteristics of the face. This process is

more reliable, efficient and time saving as compared to manual attendance system.

- Our system will going to detect multiple faces at the same time.
- Our system will find the attendance of each student in the respective month of the year
- There are two options for the student to mark their attendance either by manually by browsing the image from the computer or live by opening the camera.
- In our system, there is an auto attendance option where we can set the specific time when we want to mark the attendance. At that time the webcam will run in the background and mark the attendance of the person who comes in front of the webcam.
- We can access the attendance record of the one year.^[3]

III. OBJECTIVE

1. Face recognition provides an automated attendance system which is more reliable, practical and saves lots of time. It is more faster than the traditional methods.
2. Can calculate the percentage of a student attendance easily without the user interaction depending on their recorded attendance rate
3. Will use in the future for the government purposes, can identify the face of a kidnapper and check the criminal status of any person

4. Identify the individual person based upon their existing status

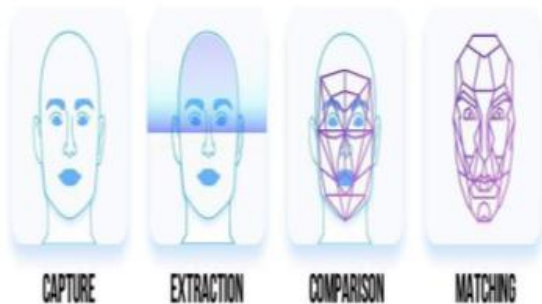


Figure 1. Process of Face Recognition

IV. GOALS

The main goal of this system is to make the attendance system more reliable and time saving process

1. Help the faculties or lecturers to manage or maintain the attendance of students more efficiently
2. Reduce some calculation errors like percentage error
3. A system should be valuable attendance system for both teachers and students
4. Increased the security, so that the student can't be able to mark the attendance of their friends
5. A well maintain report for the month will be generated at the end of the month. Teacher can print it for future purposes
6. A better way to punch the attendance and also can edit the attendance.

V. DESIGN

Here we are going to discuss about the flow chart of our system. There are various steps that need to be taken care before implementing the project.

Security: Security plays an important role in any project/software in today's era. We know customer wants a project where all his/her desired functionalities met and a programmer wants that all the test cases are passing for the functions which he has implemented. During the process of implementation, programmer leave some error in the program, that leads to the fault and faults lead to the failure like system crash.

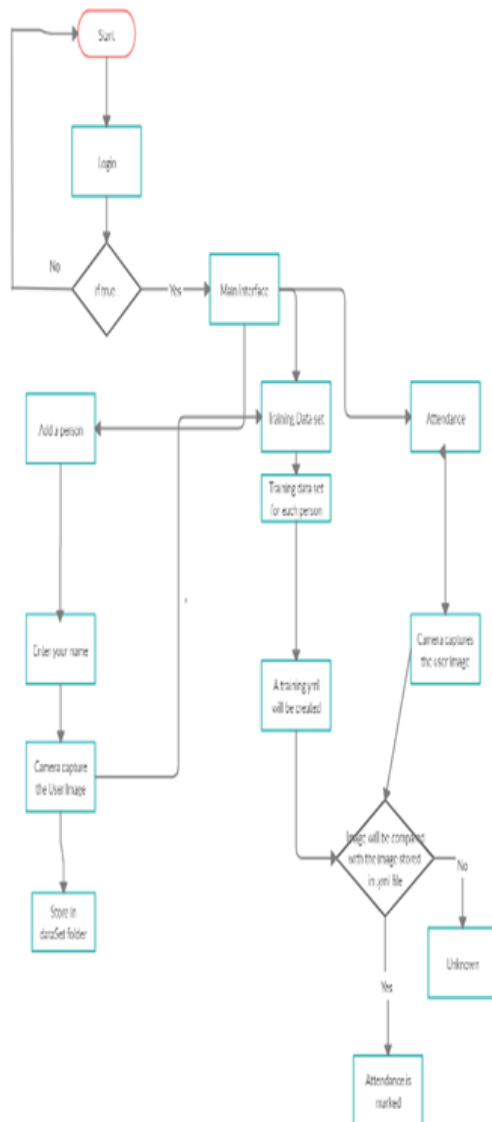


Figure 2. Flow Chart of the Implemented System

VI. PSEUDO CODE

Pseudocode defines that how our code is working internally. Write pseudocode is most important part for any software. Pseudocode allows us to write error-free code and the developer can easily implement after writing

the pseudocode. Let's write Pseudo code for our project.

1. First part is Login the system, Admin will get the privileges to access the system. Login part will be designed in QT Designer
2. The admin will login to the system then there will be the main interface for the attendance system
3. Now the main interface is having various features, like add person, training, open camera, capture the image and attendance
4. When the person will click on add person, a dialog box will appear on the screen, where the user has to ask his name
5. After the creation of new user, there will be responsibility to train that user so that system can predict the user next time when he came in front of camera
6. Now there are two options to mark the attendance 1. Manually – Manually method is like the user browse the image from their computer, after that click on attendance button to save the attendance in MySQL server if the user is a authorized or trained user.

2. **Live-** In this method, camera will be open and attendance will be taken in real time if the person is authorized, otherwise Unknown text will be displayed on the Image.

Now there will be an option for the admin to set the time, the camera will be opened at that time and mark the attendance for those who are at front of the camera.

VII. IMPLEMENTATION

1. **Image Acquisition:** Image is taken as raw data. Here we are predefined features in OpenCV that can be used to capture the image

```
cam = Cv2.VideoCapture(0)
```

This function is used to open the camera where 0 specifies that the inbuilt camera is used for image capturing ^[4]

2. **Histogram Normalization:** We cannot directly do operations on color image, so we need to convert it into gray scale image.

3. **Face Detection:** Now here is the next step is to face detection,

- Now the face will be detected by marking the rectangle on the faces of the student
- Next step is to crop each detected face. There are certain parameters to crop the face image like (x-axis, y-axis, w-width, h-height). ^[5]
- The face detection algorithm will take the sample images from different angles

4. **Training Dataset:** Now we have taken the samples of the user. Every sample has the different features so we have to find

the feature matrix for each and every image. It means if the samples is 100 then 100 feature matrix will be generated. Now we have also stored the label of the persons to separate the images of different persons. ^[10]

Now, we are going to use

LBPHrecognizer.create() function that is inbuilt in OpenCV to train the model, let's store this function in recognizer

```
Recognizer.train(feature_matrix,  
label_matrix)  
Recognizer.write('training.yml')
```

In the first step we have trained the system with the feature matrix and the labelled matrix

And in the second step we are writing the features in an yml called as training.yml ^[6]

5. **Face Recognition:** Now this is the last step in the program, again the input image will be taken from the camera and a feature matrix will be generated for that image.

Now we will compare this feature matrix with the existing yml file whether the same feature matrix is available or not. If it is there, then we will get the label name as an output.

```
If(label==1)
```

```
    Name='sunil'
```

```
Else
```

‘Unknown Person’ // if we have store only the one person sunil in the database

6. There will be two option for the user, one is to mark the attendance manually and other with camera.
7. Manually attendance system will take the image from the local system and according to that attendance will be marked.

This section will familiarize you with the overall interface of the software including the main page and other Graphical User Interface page. The code was written in PyCharm that makes our work easy. PyCharm has the function debugging mode, using this method we can easily find out the error in the code. It will go line by line with the flow of program and show the output in the output section.^[7]

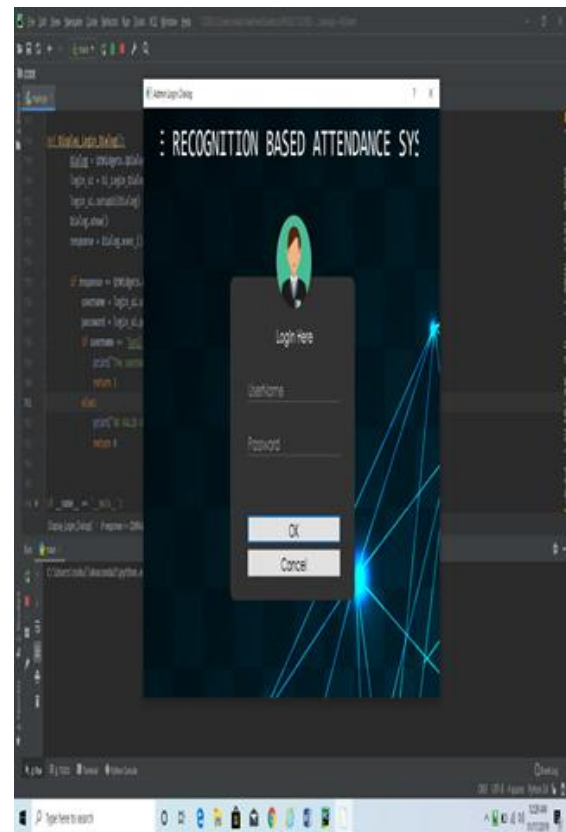


Figure 3. Login Interface of the System

Here the admin will enter into the software by providing his id and password. QT Designer software is used to make the GUI for the python.

Here the admin have to give their credentials to enter into the system (Username and Password), if the credentials are ok then only it will move to the next screen.

QLineEdit.text.accepted() this is the function runs whenever admin provides the correct id and password and pressed on ok button else QLineEdit.text.accepted() ^[8]

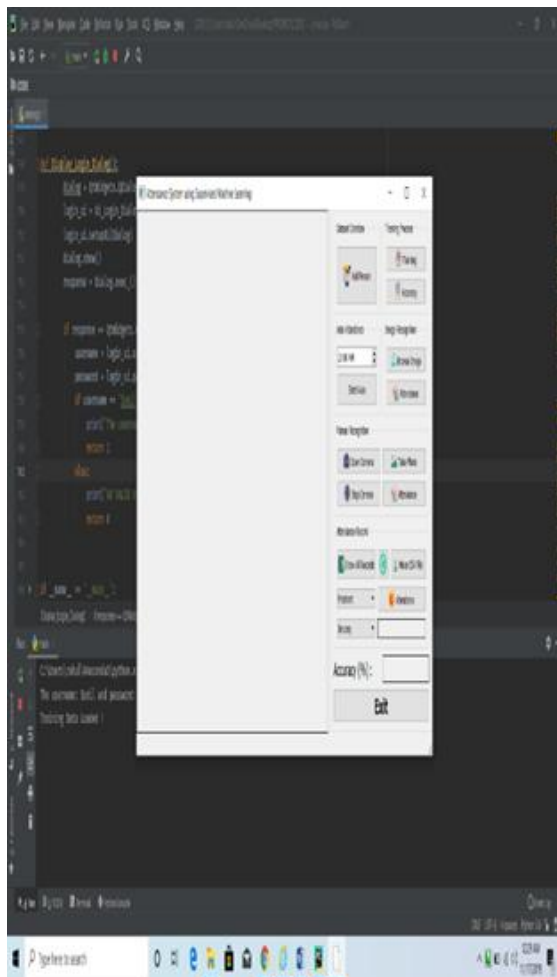


Figure 4. Main Interface of the System

function will call.

Self.addbutton.clicked.connect(self.createnew person()) // this is the example

Here self is a global structure, it shows that whenever add button will be clicked a function create new person will be call.

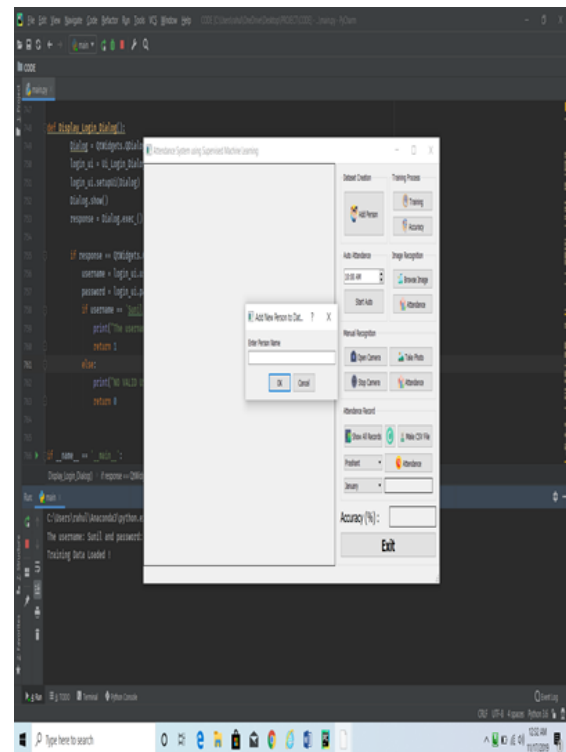


Figure 5. Process of Adding New Person

After login into this system, admin will get the interface where all the features of the face recognition can be done.

1. **Add Person:** Add person button is used to add the student in the database. After clicking in this button, a dialog box will be open which will ask the name of the student or person.
2. **Training:** Training button will be going to train the data set and finally save all the data into the Training.yml file.
3. **Auto Attendance:** Auto attendance is the extra feature in this software where we have to set the time when we have to take the attendance of the person or student.^[9]

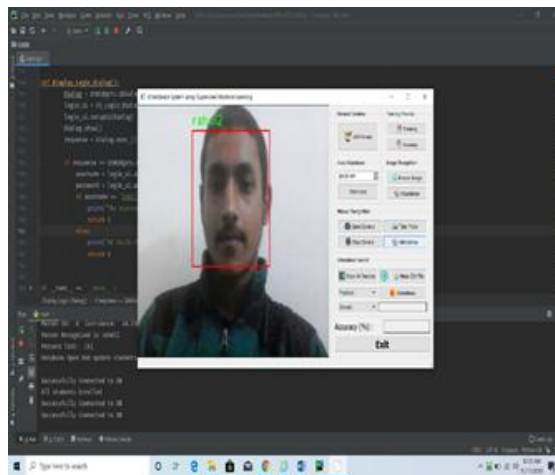


Figure 6. Recognizing the single face

Now there are two options for user to mark the attendance one is manually and other is live.

1. **Browse Image:** When we click on browse image button then the option will be come to select the image from the personal computer or system.
2. **Attendance:** After selecting image through browse image click on attendance then the rectangle will be mark on the face of the user and attendance will be marked automatically
3. **Open Camera:** Camera will be opened when we clicked on the open camera button and show the image of the user
4. **Take Photo:** Take photo button is used to capture the picture of the person or the student. After this we have to click on the Attendance button to mark the attendance of the person automatically. If the image of the person already there in existing database then a rectangle box will

be appear on the face of the user and will show his name, otherwise it will show UNKNOWN.

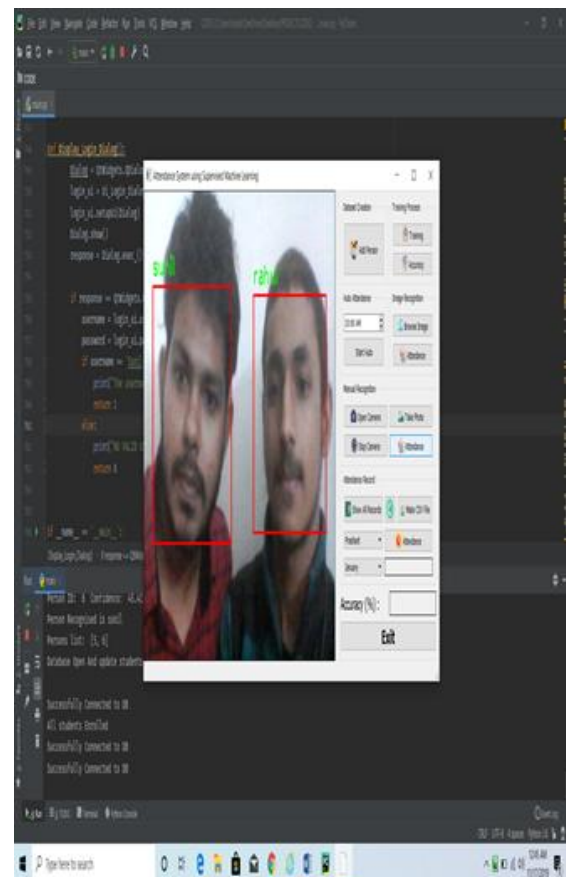


Figure 7. Recognizing the multiple faces

Our software can be used to detect multiple students or the persons at the same time. This feature can save lots of time of the admin and will work efficiently. Here we have to first add two new persons and then we will click on training dataset. After training dataset click on open camera->take photo-> attendance. Then the attendance will be marked for both of the person at the same time.

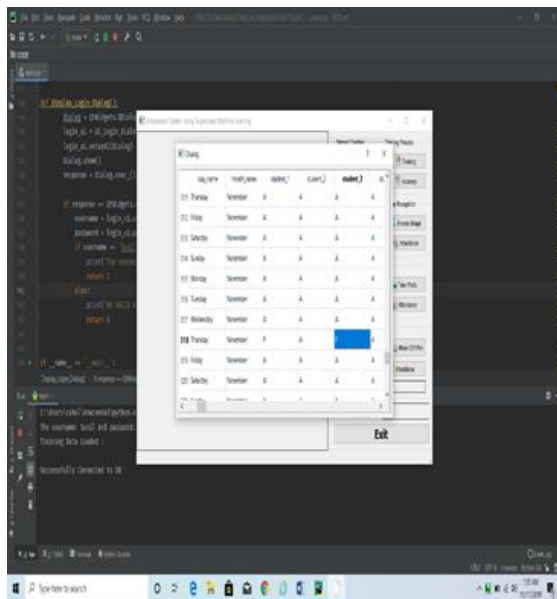


Figure 8. Marking the attendance of the recognized person

We can access all the attendance records by clicking on the button show records. Show record will show all the details of the months and student attendance with respect to that month. There is another option to download these reports and save the file in offline mode. You just have to click on CSV file and file will be downloaded in no time.

VIII. CONCLUSION

We know both Eigenfaces and Fisherfaces algorithm are not good and give false result in low lightning. We cannot say that the light condition will be good in the future that's why we have used LBPH (Local Binary Pattern Histograms) that works fine even in low lightning. LBPH face recognition is the improvement version of Eigenfaces and fisherfaces, and we can guarantee the accuracy of this algorithm. False rejection rate and false acceptance rates are very low. We have

MySQL database that is a plus point to expand the scope of the project as we know we can't store more number of students data in the system, so we need database management system.

IX. FUTURE SCOPE

- We will improve its accuracy through AWS Recognition and we will also use Oracle 12 C for student's database.
- After successfully testing database we will shift the database to AWS RDS for automatic online updating database keeping in mind IOT (Internet of Things).
- Creation of user friendly interface with password authentication.
- Making it more user friendly like splitting the branches and enrolling the students into different branches

X. REFERENCES

- [1] Bradski, Gary, and Adrian Kaehler. *Learning OpenCV: Computer vision with the OpenCV library*. " O'Reilly Media, Inc.", 2008.
- [2] Turk, Matthew A., and Alex P. Pentland. "Face recognition using eigenfaces." *Proceedings. 1991 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 1991.
- [3] Shoewu, O., and O. A. Idowu. "Development of attendance management system using biometrics." *The Pacific Journal*

of *Science and Technology* 13.1 (2012): 300-307.

[4] Reichenbach, Stephen E., Stephen K. Park, and Ramkumar Narayanswamy. "Characterizing digital image acquisition devices." *Optical Engineering* 30.2 (1991): 170-178.

[5] Rowley, H. A., Baluja, S., & Kanade, T. (1998, June). Rotation invariant neural network-based face detection. In *Proceedings. 1998 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (Cat. No. 98CB36231)* (pp. 38-44). IEEE.

[6] Ou, Yang-Yen, et al. "An integrated vision system on emotion understanding and identity confirmation." *2017 International Conference on Orange Technologies (ICOT)*. IEEE, 2017.

[7] Islam, Quazi Nafiul. *Mastering PyCharm*. Packt Publishing Ltd, 2015.

[8] Seifert, Sascha, Oliver Burgert, and Rüdiger Dillmann. "MEDIFRAME—an extendable software framework for medical applications." *Grenoble, France: Surgetica* (2002).

[9] Kim, Bong-Gi. "An Implementation of Auto Attendance Management System based on App using NFC Technique." *Journal of the Korea Academia-Industrial cooperation Society* 17.2 (2016): 719-723.

[10] Kamencay, Patrik, et al. "Accurate wild animal recognition using PCA, LDA and LBPH." *2016 ELEKTRO*. IEEE, 2016.