Attendance Management System

Bachelor of Technology
In
Information Technology

Submitted by

Serial No.	Name	Enrollment No.
1	Navneet Singh	IIT2019056
2	Navdeep Kumar	IIT2019082
3	Gautam Kumar	IIT2019087
4	Harshal Meshram	IIT2019104

Under the supervision of
Dr. Anjali Gautam

Department of Information Technology
INDIAN INSTITUTE OF INFORMATION TECHNOLOGY,

ALLAHABAD 211015,(U.P.) INDIA

Declaration

We hereby declare that the word presented in this Project report entitled ”**Attendance Management System using Face Recognition**” submitted towards the fulfillment of the 5th Semester Project Report (2021) of **B.Tech** in **Information Technology** at the **INDIAN INSTITUTE OF INFORMATION TECHNOLOGY ALLAHABAD**, U.P., INDIA is an authenticated record of my original work carried out from July 2021 to Dec 2021 under the supervision of **Dr. Anjali Gautam.** The thesis is being accomplished in full compliance with the requirements and constraints of the prescribed curriculum.

Place: Allahabad
Date:03/12/2021

(Submitted by)
Navneet Singh (IIT2019056)
Navdeep Kumar (IIT2019082)
Gautam Kumar (IIT2019087)
Harshal Meshram (IIT2019104)

Certificate

It is certified that the work contained in the Project titled "Attendance System utilizing Face Recognition" by "Navneet Singh, Navdeep Kumar Gautam Kumar, Harshal Meshram" was completed under my/our supervision and that no other work has been submitted.

Date: .....

Signature of Supervisor
Dr. Anjali Gautam

Acknowledgment

We would like to express our gratitude to the mercy of Almighty that gave me enough strength to complete this project work. We are very thankful to our supervisor "**Dr. Anjali Gautam**", Assistant Professor, Department of Information Technology, IIIT-Allahabad to guide us throughout the project work and always motivating us to work hard. The regular discussions and extensive comments helped to improve our project and a direct impact can be seen on the project. We are thankful to **Dr. Pavan Chakraborty**, **Head of Department (IT), IIIT-Allahabad**, for providing valuable guidance at the beginning of the project. We are also grateful to all the faculty members and supporting staff who have been very supportive in providing resources to complete this project.

Abstract

To maintain the attendance record of the students of any institution/organization manually is a very challenging task. The traditional method of calling the names of the students is very time-consuming & a difficult process, especially for large groups of students/employees and there is always a chance of proxy attendance. This report reviews a different method that can be adopted to reduce the time wasted while taking attendance. This method is fast, economical, time-saving, and easy to use. Here we've used OpenCV, Haar cascade classifier, and LBPH algorithm for face detection and face recognition. For the GUI we've used Tkinter as it is easy to use and gives us a clean and understandable GUI.

Contents

1. **Introduction**
2. **Problem Statement**
3. **Literature Survey**
4. **Datasets**
5. **Prerequisites/Learning**
6. **Implementation**
7. **Results and Conclusion**

1. Introduction

Attendance marking during a lecture in a classroom is not only a difficult but also a time-consuming process. There will always be a chance of proxy attendance if there is an exceptionally large number of students present during the lecture(s). It's proven difficult to track attendance using traditional ways. In the field of face recognition, the growing demand for efficient and automatic ways for registering attendance is posing a significant difficulty. In recent years, common biometrics such as fingerprints and Radio Frequency Identification tags have been widely used to address the problem of automatic attendance marking. However, these solutions lack the element of reliability. Face detection and identification algorithms are used in this suggested project to create an automated attendance marking and management system.

The automatic attendance system was created to track the attendance of students. Educators and students are given their own logins and passwords, which may be used to mark and record daily attendance in a centralized location. A student attendance system, which is an automated solution that not only marks attendance but also finds relevant trends in order to monitor student performance around important metrics and contribute to student security, can save valuable class time.

The attendance management system will be used to store the attendance of the students automatically with the help of facial recognition(using LBPH and haar cascade classifiers)[9] to keep track of student attendance. Student attendance on a daily basis will be recorded and stored already in the database. As a student arrives the system will automatically match his/her face with the images present in the database by applying face detection and recognition techniques and the recognized students will be they've been marked as present, and their attendance has been recorded will be updated with the corresponding time. Digitalization of the system would also help in better visualization of the data using tables to display the number of Students present today, as well as the total number of hours each student has spent in the premises.

2. Problem Statement

To Build an automatic attendance marking system that will mark the attendance of the registered users/students/faculties by identifying them through facial recognition and also noting their timings.

3. Literature Survey

In [1] Nandhini R, Duraimurugan N, S.P.Chokkalingam made ”FACE RECOGNITION BASED ATTENDANCE SYSTEM” used a machine learning approach to implement the idea of two technologies: student attendance and management system. This technology keeps track of student attendance and automatically marks them as a present. As a result, by identifying the student's face, the student's attendance can be determined. Following recognition, the attendance details are stored in a database, and we receive an overall report of the student's total attendance time, including in and out times.

In [2] Kaneez Laila Bhatti et al. made ”SMART ATTENDANCE MANAGEMENT SYSTEM USING FACE RECOGNITION” in which they used HOG method to detect human faces in a given image and they took 15 images of each individual for the purpose of their face recognition in their model and the applied deep learning to compute 128-d facial features and stored it in their local database using JavaScript Object Notation(JSON).

In [4] Dhanush Gowda H.L et al. made ”FACE RECOGNITION BASED ATTENDANCE SYSTEM” in which they converted an input image into a feature vector using HOG and used OpenCV for frame extraction after converting an input image from BGR to RGB format because Dlib uses RGB format as default and dlib for face recognition.

In [7] Venkata Kalyan Polamarasetty et al. made ”ATTENDANCE SYSTEM BASED ON FACE RECOGNITION” in which they took 500-100 images of each individual using hd camera for better accuracy and they used Viola Jones algorithm for face detection and HOG for recognition and they made GUI using MATLAB tools.

In [10] Prajakta Lad, Sonali more et al. made “STUDENT ATTENDANCE SYSTEM USING IRIS DETECTION” in which the student stands in front of the camera in order for the system to recognize his/her iris and record his or her attendance in this proposed method. GrayScale Conversion, Six Segment Rectangular(SSR) Filter, and Skin Pixel Detection are all utilized to detect the iris. It avoids proxy issues and maintains the student's attendance effectively, but it is a time-consuming process to detect Iris.

In [11] Radhika C. Damale made “FACE RECOGNITION-BASED ATTENDANCE SYSTEM USING MACHINE LEARNING ALGORITHMS”

using different methods such as Support vector machine(SVM), MLP and CNN for face detection. For SVM and MLP approaches, the features like PCA and LDA were extracted using extraction algorithms. In the CNN approach, images were taken directly from CNN Module as a feature. The approach shows Good detection accuracy on self- generated databases.

4. Datasets

4.1. *Data Collection*

The initial requirement is to collect the images of individual students which can be taken automatically using this model after entering your Enrollment Id and name. For this, we have used OpenCV video capture which takes our images from the frames of the video as we go in front of the camera and crop the useless part automatically. And for now, we're storing these images in the local database itself and they're stored in JPG format, their dimensions are 150x150 (approx) and the size of the images ranges from 15-20 kb and these values can vary depending upon the distance between the camera and the student.

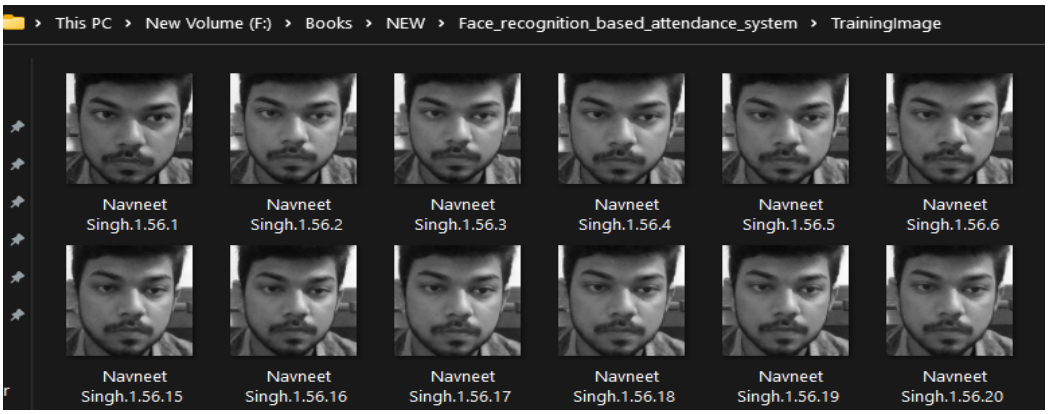


Fig. 2. Training Images for Data set

4.2. *Data Transformation*

After data collection, the next step is to transform those images into processable data. The reason for this is to make it easier for analysis and to mark the attendance easily. Here, we have used Local binary pattern (LBP) is a method of labeling pixels in an image by thresholding each pixel's neighborhood and using the result as a binary number.

5. Prerequisites/Learning

5.1. LBPH -The Local Binary Pattern Histogram (LBPH) is a Face-Recognition technique for identifying a person's face.[12][15]

5.2. **OpenCV** - Open Source Computer Vision (OpenCV) is an open-source library of programming functions used for real-time computer vision. It is used for image processing, video capture, and analysis for features like face and object recognition.[14]

5.3. **Haar Cascades** - It's an Object Detection Algorithm that recognizes faces in real-time videos and photos. Edge/line detection features were provided by Viola and Jones, which were utilized in the approach.[9]

6. Implementation

After some research, we propose an effective method for capturing photos of each student's face and storing them in a database for attendance purposes. The student's face must be captured in such a way that all of the features of the student's face, such as the right side of the student's face or the left side of the student's face, are detected. The teacher does not need to physically take attendance in class because the system will detect the face of the student and it'll try to find a matching image of it from the database where the images of registered students is stored.

I. Algorithm

1. Capture the face of each student using OpenCV and haar cascades face detector.
2. Store it in the local database for using them further.
3. The student's face should be captured from every angle such that all the features of the student's face can be detected.
4. After face recognition, face identification will be done using a local binary pattern histogram(LBPH) face recognizer.
5. In face identification, a one-to-many mapping for a given face against a database of known faces will be done.
6. If the captured face is found in the database then it'll mark that student present and note in time.

II. LBPH Model

We may represent the facial photos with a simple data vector using the LBP [12][15] mixed with histograms.

LBPH uses four parameters:

1) ***Radius***: It's the radius that's utilized to make the circular local binary pattern, and it's normally set to one.
2) ***Neighbors***: It's the number of sample points used to create the circular local binary pattern, which is typically set at eight.
3) ***Grid X***: It's the horizontal cell count, which is generally set to eight.
4) ***Grid Y***: It's the vertical cell count, which is generally set to eight.

III. Steps

1. To train the model, we must first upload face photographs of the students we want to recognize to our local database, then assign an ID (i.e. roll number) to each image so that the model can recognize the students and display their names on the screen. Images of the same person must also have the same ID.
2. The training materials have already been developed. The first step of LBPH is to create an intermediate image that highlights the facial aspects of the original image to better describe it. The algorithm does this by utilizing the concept of a sliding window, which is influenced by the radius and properties of the neighbors.

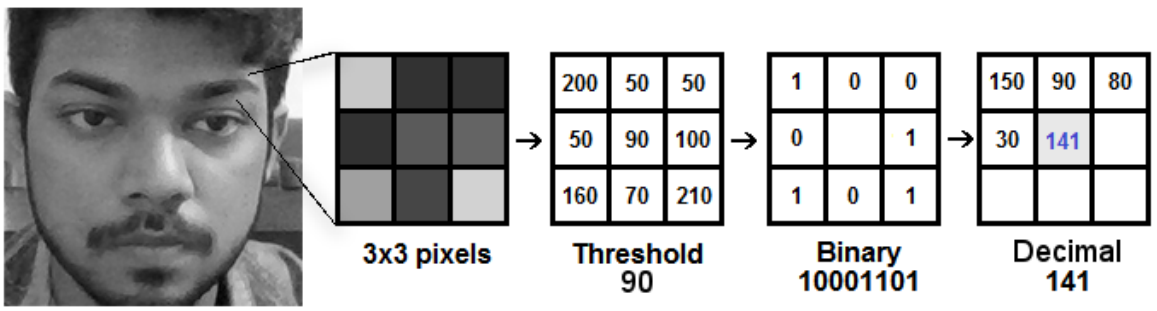
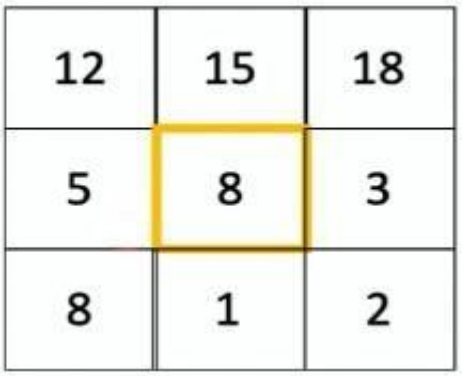


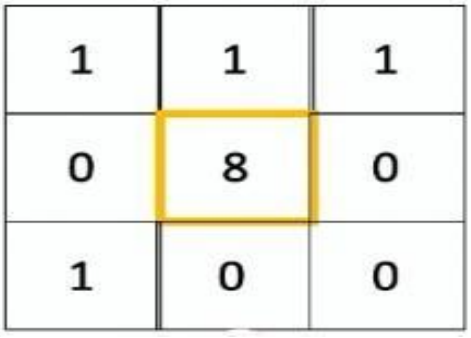
Fig. 3. LBPH creates an intermediate image

3. As we know an image is made up of a group of pixels and each pixel is made up of little squares. We can build the whole image by putting them side by side. A single-pixel contains the least amount of information possible of an image. The value of pixels in each image ranges from 0 to 255.

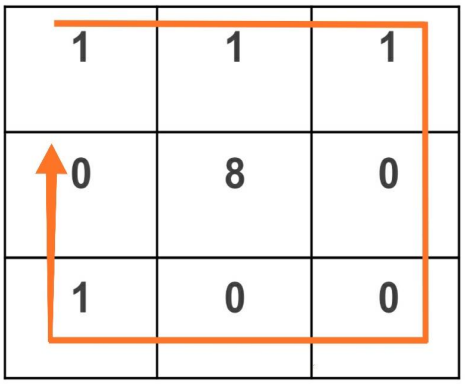
4. Let’s take the following example



5. There are three rows and three columns in this image, with a total of nine pixels. Let's apply a condition on the middle pixel, which is value eight. The result is '1' if the value is greater than or equal to 8. Otherwise, the result is zero if the value is less than eight. New matrix after applying conditions stated above will look like the following-



6. Select the matrix's central element now. We produce the number 11100010 as a binary value. The algorithm will begin applying the condition from the top left corner element and work its way up to the first element of the second row, as shown below.



7. We get Decimal Number = 226 after converting the Binary value to decimal. It means that all of the pixels in the area around the center value are equal to 226.

8. Now we convert the binary value to a decimal number, which will be assigned to the matrix's center value, which is a pixel from the original image. At the end of this approach, we have a new image that better captures the qualities of the original image.

9. Now, dividing the image generated in the above step into multiple grids using GridX and GridY, as shown below-

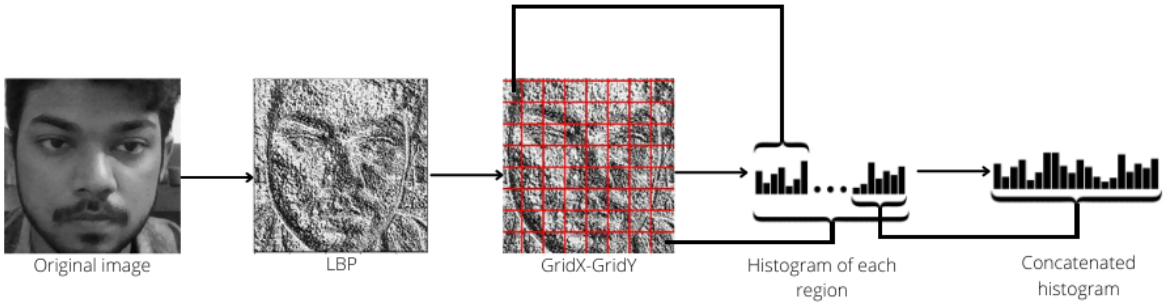


Fig. 4. Dividing the image into multiple grids

10. We can extract the histogram of each region on the basis of the above image as follows:

    A. .Because we're working with a grayscale image, each histogram (from each grid) will only have 256 locations (0~255) to record the occurrences of each pixel intensity.

    B. Then, to make a new, larger histogram, we must concatenate each histogram. In the case of 8x8 grids, the final histogram will have 8x8x256=16,384 locations. The original image's attributes are represented by the final histogram.

11. **Now, as the algorithm is already trained.** Each histogram created is used to represent each image from the training dataset. So, given an input image, we perform the steps again for this new image and create a histogram that represents the image. So to find the image that matches the input image we just need to compare two histograms and return the image with the closest histogram.

III. Accuracy

For the accuracy part we have, *Recognizer.predict(gray[y:y+h,x:x+w])* function which returns-

- The prediction index that shows the number of the person in the training set.
- Confidence for all persons in the training set with a confidence value.
- The calculated distance, which can be used as a '**confidence**' measurement.
- The **lower the confidences are better because it means the distance between the two histograms is closer.** We've set it for **<50**

Code-

```
serial, conf = recognizer.predict(gray[y:y + h, x:x + w])
#checks confidence
if (conf < 50):
    ts = time.time()
    date = datetime.datetime.fromtimestamp(ts).strftime('%d-%m-%Y')
    timeStamp = datetime.datetime.fromtimestamp(ts).strftime('%H:%M:%S')
    aa = df.loc[df['SERIAL NO.'] == serial]['NAME'].values
    ID = df.loc[df['SERIAL NO.'] == serial]['ID'].values
    ID = str(ID)
    ID = ID[1:-1]
    bb = str(aa)
    bb = bb[2:-2]
    attendance = [str(ID), '', bb, '', str(date), '', str(timeStamp)]

else:
    Id = 'Unknown'
```

Fig. 5. Checking confidence value

7. Language and Technology used

In our project, we used Python because it is lightweight, versatile, and simple, as well as OpenCV, Haar Cascade classifiers, and Local Binary Pattern (LBP) for face detection and recognition.

Tkinter was used to create the user interface.

Used python libraries-

- **Pandas**
- **Numpy**
- **CSV**
- **PIL**

8. Flow Chart

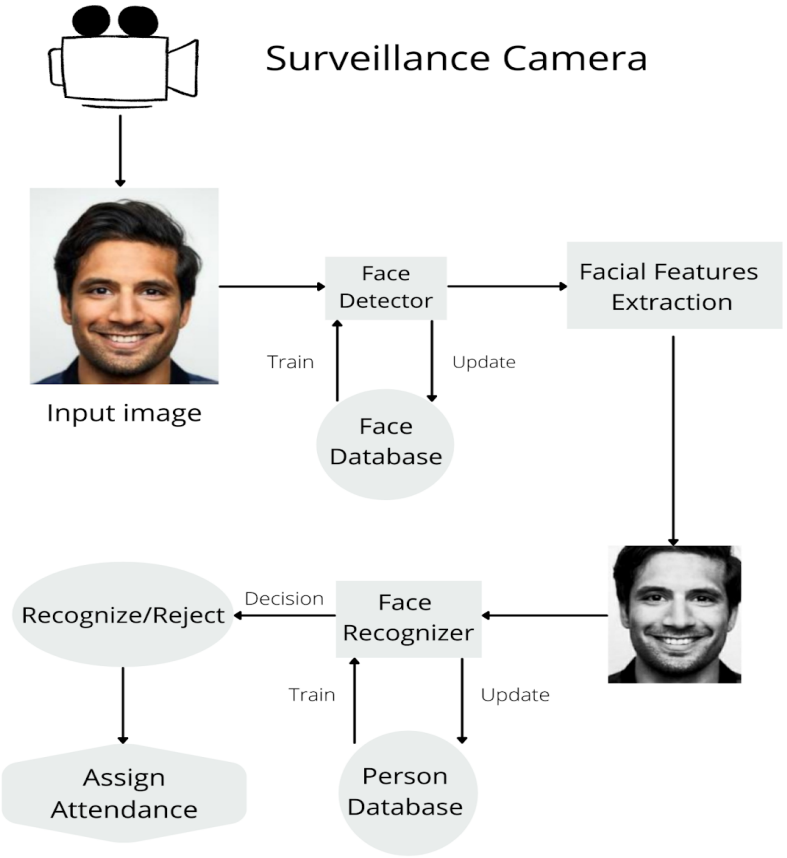


Fig. 6. Workflow for face detection

9. Results

1. Registering yourself in the local database by entering your Name and Enrollment ID

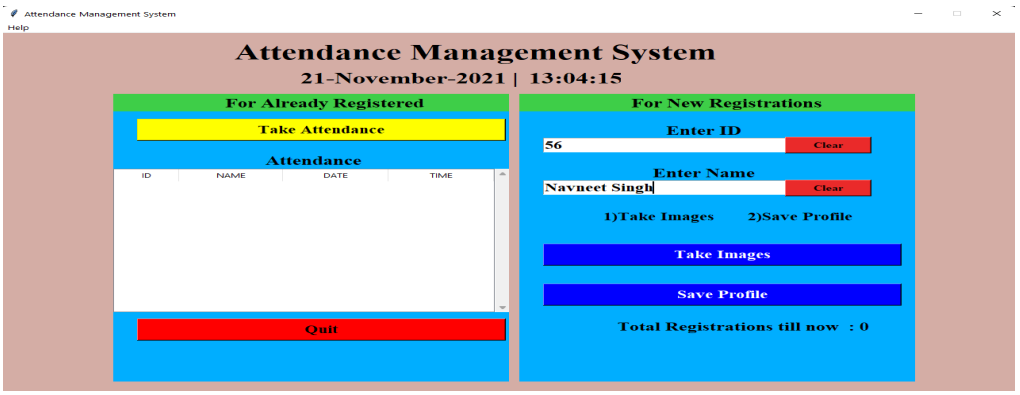


Fig. 7. Registering and enrolling by Name and ID

2. Taking images for the local database which will be used later for recognizing your face for marking your attendance

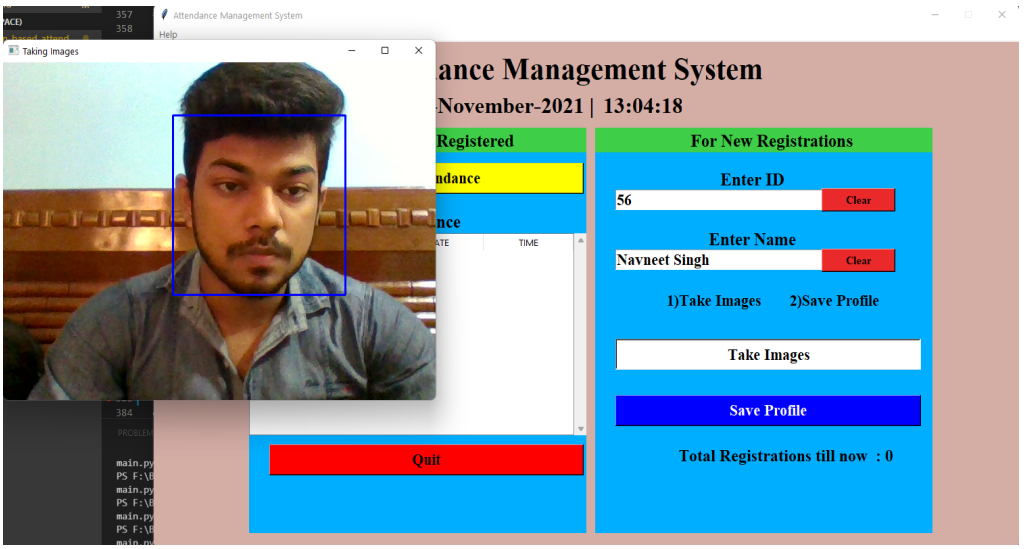


Fig. 8. Taking images for database

3. Clicking on the Save profile button and Entering a password(which will be provided by us) so that your identity can be confirmed

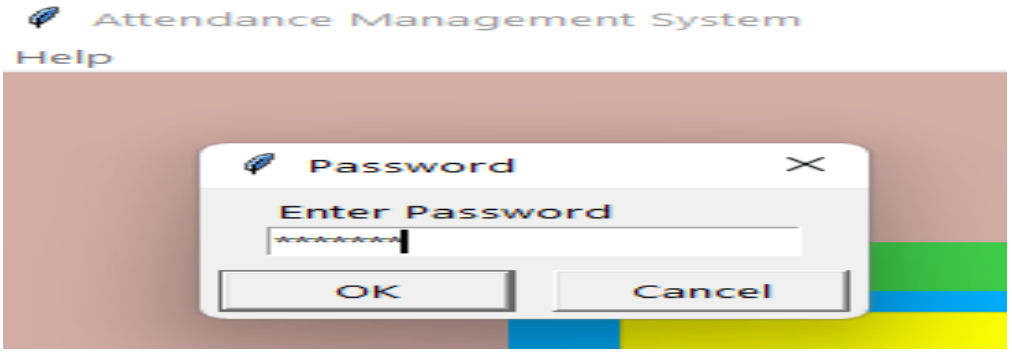


Fig. 9. Saving Profile by creating a password

4. Now, after your profile gets saved successfully, Click on the take attendance button after which your face will be detected through your webcam/ camera

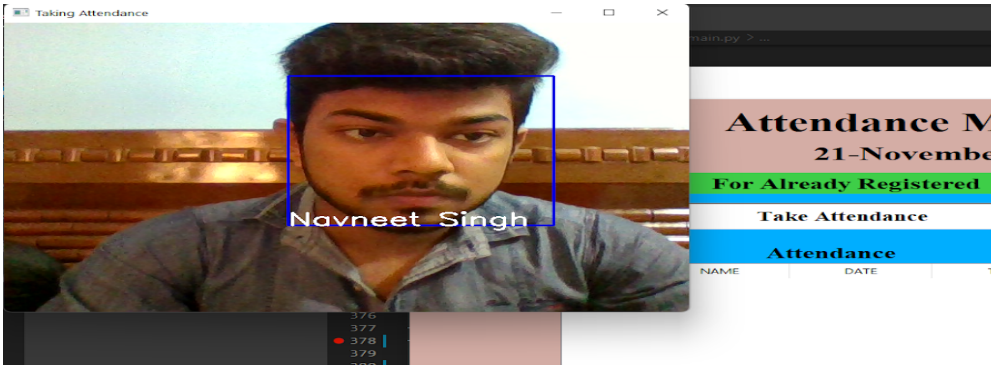


Fig. 10. After clicking on taking attendance our face gets recognized and name is displayed

5. After successfully detecting your face press ‘q’s and your attendance will be marked and displayed with time

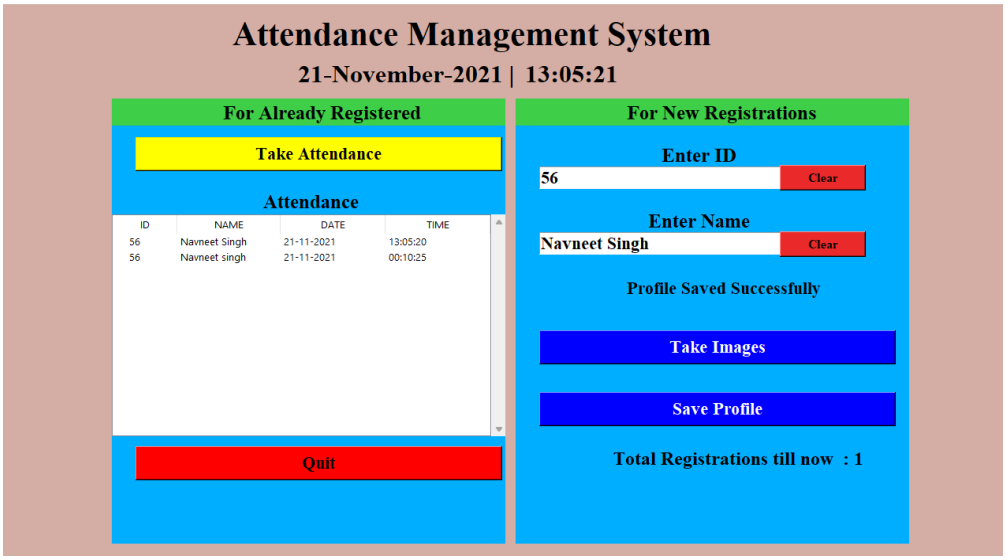


Fig.11. After detection and clicking on Quit, our attendance is marked and displayed.

10. Conclusion

In this report, a functional real-time Attendance management system has been made. By using different algorithms and techniques like Haar Cascade Classifier, Local binary patterns(LBP), and OpenCV for facial recognition operations the accuracy will increase and the chances of a proxy will decrease. In the future, this project can be enhanced and could be used at those places where we require surveillance like ATMs and banks. Using various concepts of image and video processing and fundamental properties of the image, we tried to develop this project.

11. References

1. Nandhini R, Duraimurugan N, S.P.Chokkalingam, ”Face Recognition Based Attendance System”, published in International Journal of Engineering and Advanced Technology (IJEAT), February 2019

2. Kaneez Laila Bhatti, Laraib Mughal, Faheem Yar Khuhawar, Sheeraz Ahmed Memon,”Smart Attendance Management System Using Face

Recognition", analysis in Dept. of Telecommunication Engineering, MUET, Jamshoro, PK, 29 October 2018

3. Yi-Qing Wang, CMLA, ENS Cachan," An Analysis of the Viola-Jones Face Detection Algorithm", published in Image Processing On Line (IPOL), 26 June 2014

4. Dhanush Gowda H.L, K Vishal, Keerthiraj B. R, Neha Kumari Dubey, Pooja M. R.," Face Recognition based Attendance System", analysis in Vidyavardhaka College of Engineering, Mysuru, Karnataka, India, published in International Journal of Engineering and Advanced Technology (IJEAT), 06 June 2020

5. Mehul K Dabhi, Bhavna K Pancholi," Face Detection System Based on Viola-Jones Algorithm", analysis in M. S. University Baroda, India, published in International Journal of Science and Research (IJSR),4 April 2016

6. Hajar Filali, Jamal Riffi, Adnane Mohamed Mahraz, Hamid Tairi," Multiple face detection based on machine learning", analysis in Department of computer science Faculty of Sciences Dhar-Mahraz Sidi Mohamed Ben Abdellah University, Fez, Morocco.2018

7. Venkata Kalyan Polamarasetty, Muralidhar Reddy Redeem, Dheeraj Ravi, Mahith Sai Madala," Attendance System based on Face Recognition", published in International Research Journal of Engineering and Technology (IRJET),4 April 2018

8. Dattatray D. Sawat, Ravindra S. Hegadi," Unconstrained face detection: a Deep Learning and Machine learning combined approach", analysis in Department of Computer Science, Solapur University, Solapur, Maharashtra 413255, India,15 December 2016

9. Paul Viola and Michael Jones," Rapid Object Detection Using a Boosted Cascade of Simple Features", analysis in MITSUBISHI ELECTRIC RESEARCH LABORATORY, May 2004

10. P. Lad and S. More, "STUDENT ATTENDANCE SYSTEM USING IRIS DETECTION," no. 2, pp. 3293–3298, 2017.

11. Radhika C. Damale,”FACE RECOGNITION-BASED ATTENDANCE SYSTEM USING MACHINE LEARNING ALGORITHMS”,analysis at Department of Electronics and Telecommunications,Cummins, published in IJERT,2020

12. Timo Ojala et al. ”Multiresolution Gray Scale and Rotation Invariant Texture Classification with Local Binary Patterns”,analysis in Machine Vision and Media Processing Unit Infotech Oulu, University of Oulu,Finland, 2002

13. Face detection using dlib HOG. If this is your first face… | by srujan jack | MLCrunch [01/12/2021]

14. OpenCV-python · PyPI [01/12/2021]

15. Understanding Face Recognition using LBPH algorithm [01/12/2021]

16. HOG [01/12/2021]