***Abstract* — Biometric authentication methods like face recognition, fingerprint recognition, etc. are growing rapidly as one of the promising authentication methods. We can use them in our mobile phones, laptops, offices, etc. But the attendance of the students is still being taken manually which takes a lot of time and is indeed a tedious task. So, in order to save that precious time, we need an automated attendance system which is efficient and reliable.**

**We have developed a face recognition-based attendance system using OpenCV. The face detection is done by using the Haar Cascade algorithm and the face recognition is done by the LBPH algorithm.**

***Keywords*** *—* ***Face Recognition, Attendance, LBPH algorithm, Haar Cascade, OpenCV***

# INTRODUCTION

For identification of a person, the most important part of the human body is the face. Face recognition is a type of biometric identification that takes all of a person's facial features and stores them as a unique face print in order to uniquely identify the person. Because of its application and range, Biometric face recognition has gained quite some popularity among researchers. Also its contact free procedure, that is why it outsmarts all other recognitions like fingerprint, iris print and palm print. Face recognition technology is also able to detect a person from far away, without coming in physical contact with anyone.

The photo that has been captured will be stored in the database. After that we can use this project in many ways like face recognition systems will be used in crime related reports. For face recognition we require a very large dataset and complex features to identify a person in all conditions like change of illumination, age, pose etc. During the recent few years, a good improvement has been made in the field of facial recognition systems. In comparison to the last decade, one can observe a very huge development in the world of facial recognition.

Today, A lot of the facial recognition systems perform very well with the limited faces in the frame. Moreover, these types of methodologies have been tested under very limited and very controlled lighting conditions, very proper face poses and usually non-blurry images. Face recognition-based attendance systems will be based on the technology of face recognition and it will be very useful for teachers as it will provide a very convenient way of marking attendance of the students for teachers. Using the LBPH algorithm for face recognition and the Haar Cascade algorithms for face detection, this attendance-based system can be used by teachers for their convenience.

# Literature Review

R. C. Damale and B. V. Pathak [1] in this paper, a computer vision and machine learning based face recognition system is proposed. They have trained a module with a lot of images. For helping in face recognition SVM, MLP and CNN are used as a classifier. The SVM, MLP and CNN nearly achieve the testing accuracy of about eighty seven percent, eighty six percent and ninety eight percent respectively.

A. Arjun Raj, M. Shoheb, K. Arvind and K. S. Chethan [2] proposed a system for face recognition using the LBPH algorithm. First, a database is created by enrolling students. They are taking several images of students using cameras. Then the images are converted into grayscale by using the LBP algorithm and after that a histogram is generated. A video stream is given to the system by which the LBPH algorithm matches the live image taken from the video stream with the images present in the database. When a face is detected successfully, attendance is marked.

S. Dev and T. Patnaik [3] proposed a system which meets the objective of achieving high precision and less computational complexity. The system is cost efficient and less manual work is needed. The use of Gabor filters greatly improves the accuracy of the system. Three algorithms are used for face recognition: KNN, CNN, and support vector machines. KNN algorithm has proved to have the highest accuracy of 99.27%, CNN has low computational complexity and SVM algorithm proved to be less efficient.

S. S. Pawaskar and A. M. Chavan [4] proposed a system of conducting attendance by using face recognition and detection algorithms like LBPH and Haar Cascade. Haar Cascade provides a very high level of accuracy irrespective of the illumination. The system can give an accuracy of about 96.88%. The faculty can also manage the class data along with the record of the attendance through this system.

R. Hartanto and M. N. Adji [5] proposed an attendance system by face recognition using the Haar cascade algorithm. Face images are captured using a camera. To determine the location of the face the face detection process was performed using the Haar Cascade algorithm. When the face image had been detected, it would make

or draw a box that covers the whole face as an ROI (Region of Interest). Image feature extraction was performed using the LBPH algorithm. Compare the detected faces with all the faces in the database to find the one closest to the detected face. The database was stored using the CSV file format to present the names and the directories of faces that exist in the database.

T. A. Kiran, N. D. K. Reddy, A. I. Ninan, P. Krishnan, D. J. Aravindhar and A. Geetha [6] recommended a PCA based facial recognition for the attendance system. A single image of each person's face is captured. Then the image is transferred to a grayscale image and stored in the database.After entering class, the camera will place a frame over the face and convert the face into gray scale and submits to the next phase. The fisher face method is used for the extraction of the features.Face recognition is performed by comparing all the retrieved values ​​with the values ​​already stored in the database. In that event where the qualities are a match, at that point the face is recognized and the name related with that face is shown. After recognition of the face, data is saved into a facedata.xml file.

O. A. R. Salim, R. F. Olanrewaju and W. A. Balogun [7] suggested a class attendance management system using face recognition using Raspberry Pi. He used the LBP algorithm for face recognition. When you are facing the camera, the image of your face is captured. The first processing step is to detect and trim the area of ​​interest, also known as the ROI, which is the human face. This can be done simply by applying the Haar feature-based Cascade algorithm. The LBP is then used to extract the features of the image and the LBP algorithm compares the extracted features to the trained dataset. If the student's face is recognized the door will open for the recognized student by using a servo motor. The attendance results are stored in the MySQL database, so it can be accessed by the web server.

R. Mehta, S. Satam, M. Ansari and S. Samantaray [8] have proposed a face recognition based automated attendance system which will have an in-built 2-tier authentication method. Faster R-CNN algorithm is used for face detection. The system is designed to operate in a classroom that has two cameras on opposite ends so that every student is captured by at least one of the cameras. Each student has a unique ID through which the student is mapped into the database. Once a student satisfies the required criterion of the two-tier authentication process, their corresponding ID is marked as present.

# Methodology/Experimental

## Materials/Components/Flowchart/Block Diagram/Theory

Diagram

Description automatically generated

**Fig. Block Diagram**

**Face Detection:** It detects the faces (position and size) in an image and extracts them so that the face recognition algorithm can use them.We used the Haar Cascade algorithm to detect faces in the image. This is an object detection algorithm that detects faces in images and real-time videos.

**Face Recognition:** The face recognition algorithm is responsible for discovering features that best describe the image once the facial images have been retrieved, cropped, scaled, and usually converted to grayscale. The LBPH (Local Binary Pattern Histogram) algorithm is a face detection algorithm that detects human faces. It is well-known for its performance and ability to distinguish a person's face from both the front and side.

|  |  |
| --- | --- |
| **Function** | **Task done** |
| VideoCapture() | For starting the camera |
| cvtColor() | Converts the input image into a specified format, e.g., grayscale, hsv, etc. |
| detectMultiScale() | Detects different-sized items in the provided image. |
| cv2.imwrite() | To save the images into the dataset |
| cv2.face.LPBHFaceRecognizer  create() | Used to load the recognizer. |
| import cv2 | To import the OpenCV module |

# Results and Discussions

The developed project is a face recognition-based student attendance system which takes the student details and images as input. The images are trained using Haar Cascade algorithm and a YML file is generated. To recognize the face, LBPH algorithm compares the histograms of the trained images with the images taken from the video stream. If the face is detected, attendance is marked in the csv file.

The Whole process basically starts by capturing the images from the input video stream. The Haar Cascade algorithm makes a box around the face of the student and starts capturing the images of the student. The image will be stored in the dataset once it converted to grayscale. Then, the model is trained with the help of those images using LBPH algorithm and a yml file is generated. Then, we start the tracking process and for face recognition we use the LBPH algorithm. If the face is recognized, attendance is marked in the csv file otherwise the image is marked as an Unknown Image.

Explanation of all the steps is as follows:

## Start Camera

The first step is to start the webcam and get the video stream as the input for the system. After starting the video, the main task is to detect the human face in it so that the images of the face can be taken, stored, and then used for face recognition.

## Face Detection

The face detection is performed using the Haar Cascade algorithm. This algorithm for object detection is capable of running in real-time.

Diagram

Description automatically generated

**Fig. Haar cascade feature(a)**

It's a machine learning-based method that involves training a cascade function using a large number of photos.

Alogrithm requires basically both positive and negative images .for face or object detection. Then it is extract to the features in it. Eachone of the feature is a single value that is basically calculated by the both subtracting the sum of pixels and the beneath of the white rectangle in the total of pixels beneath the black rectangle.

A picture containing text

Description automatically generated

**Fig. Haar cascade features(b)**

To calculate a large number of features, all feasible sizes and positions of each kernel are now used. A 24x24 window yields more than 160000 characteristics. These characteristics are extremely important in the context of face detection. Some of the basic examples of these features are:

* The area around the eyes is darker than the area near the cheeks.
* The areanear the nose is basically brighter as compared to the area around the eyes.

Therefore, given these rectangular regions and their corresponding difference of sums, it can form features that can classify parts of a face. Then, for an entire dataset of features, it uses the AdaBoost algorithm to select which ones correspond to facial regions of an image.

Text

Description automatically generated with low confidence

**Fig. Haar cascade**

We register a student into the system using Student Id and name and then start face detection process to generate the dataset.



**Fig. Face detection**

After face detection is done a StudentDetails.csv file is generated which contains the id and name of the student.

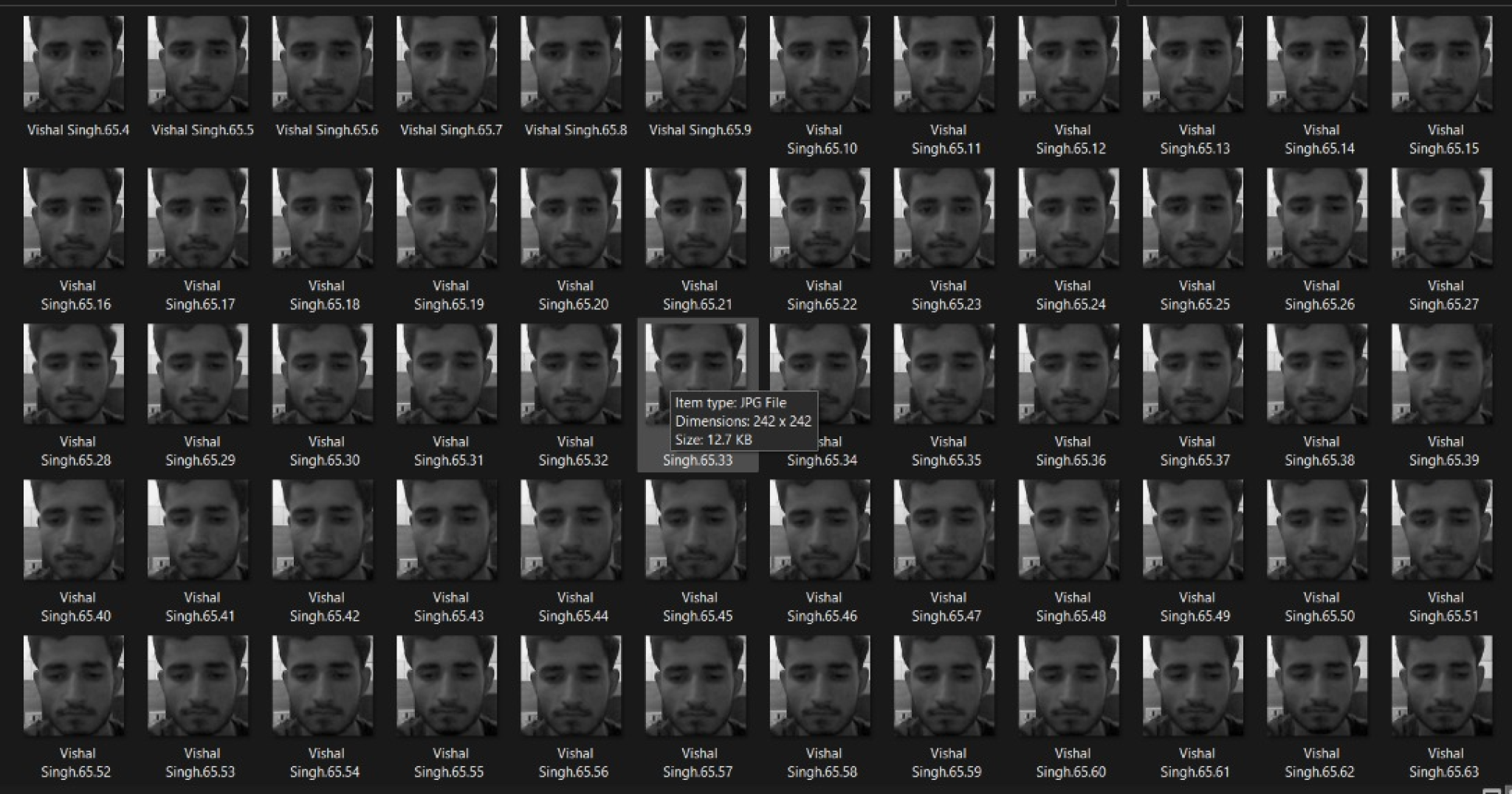
Graphical user interface, text, application

Description automatically generated

**Fig. StudentDetails.csv**

## Create Dataset

The next step performed by the system is to store the images into the dataset folder so that a histogram can be generated based on all those images.



**Fig. Created Dataset**

In the dataset a total of 200 images will be saved because at 200 images the system works perfectly as on 200, time taken is not much and the confidence level is best. If images are increased, then the confidence rate decreases and also the time taken increases a lot. Therefore 200 image is the best possible option for this system:

|  |  |  |
| --- | --- | --- |
| Number of Images | Time Taken (sec) | Confidence Level (%) |
| 100 | 7.52 | 70% |
| 200 | 9.10 | 75% |
| 300 | 13.78 | 67% |
| 500 | 22.34 | 66% |

1. *Training Model*

The next step is to train our model by using the images which we have taken. The model will be trained using LBPH algorithm. The ‘yml’ file contains the histogram generated by training on those 200 images.

Text, letter

Description automatically generated

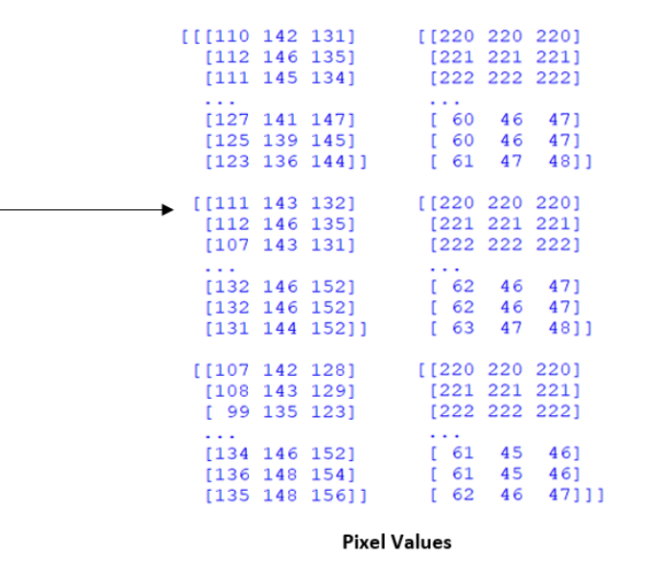
**Fig. YML File**

## Face Recognition

Face recognition is performed using the LBPH algorithm.

Local Binary Pattern (LBP) is a basic yet effective texture operator that labels pixels in an image by thresholding each pixel's neighborhood and treating the result as a binary number.

Using the LBP combined with histograms known as LBPH method, we can represent the face images with a simple data vector.

The LBPH has four parameters, which are as follows:

**Radius:** The radius represents the radius surrounding the center pixel and is used to construct the circular local binary pattern. Normally, it is set to 1.

**Neighbors:** The total number of sample points required to construct the circular local binary pattern. The higher the computational cost, the more sample points you include. Normally, it is set to 8.

**(Grid X):** Number of cells in the horizontal direction the larger the dimensionality of the generated feature vector, the more cells there are in the grid and the finer the grid is. Normally, it is set to 8.

**(Grid Y):** Number of cells in the vertical direction the higher the dimensionality of the final feature, the more cells there are in the grid and the finer the grid is. It is usually set to 8.

LBPH only reviews the local features of the important objects. In face recognition cases, the LPBH algorithm will only review the face, eye, and mouth features.

The LBPH algorithm's first significant step is to build an intermediate image that describes the original image better than the original image by highlighting the major facial features.

To do so, Algorithm uses the sliding window concept, which is entirely reliant on radius and neighbor parameters. Suppose we have a grayscale facial image. We can get a part of that image as a window of three x three pixels. It can also be represented as a three x three matrix which contains the intensity of each and every pixel (0~255).

**Fig. Extracting pixel values from input image**

Then we must take the matrix's central value, which will be utilized as the threshold value. This value will be used to define the new values from the eight neighbors.

Text, letter

Description automatically generated

For each neighbor of central value, we set a new binary value. We set 1 for values equal or higher than the threshold and 0 for the values lower than the threshold value. Every region employs a local binary operator. The LBP operator defined in window of 3x3 is.

Schematic

Description automatically generated

here '(xc,yc)' is central pixel with intensity 'ic'. And 'ip' is the intensity of the neighbor pixel.

If the value of neighbor pixel is greater than or equal to the central value, it is set as 1 otherwise it is set as 0 or

we compare a pixel to its 8 closest pixels using the median pixel value as a threshold value.

Diagram

Description automatically generated

The matrix will now only contain binary values (ignoring central pixel value).

Text

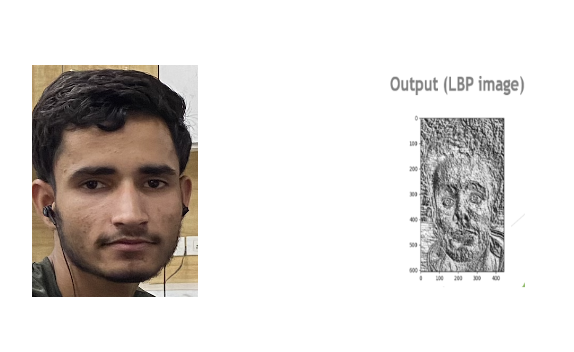
Description automatically generated

Then, we have to convert this binary value to decimal value and will set it to the central value of matrix, which is a pixel from the original image.

Table

Description automatically generated

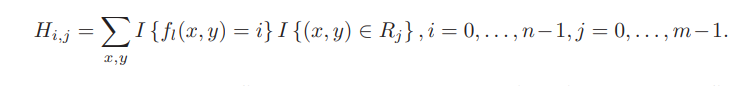
At the end of this procedure, we will have a new image which will represent the characteristics of the original image better than that in the original image.



**Fig. Converting Input image to LPBH image**

Now, if we use the image generated in the very last step, we can use the two parameters which are Grid X and Grid Y, to divide the image into large number of grids. we can extract the histogram of each region now.

As we have image in the grayscale format, each histogram will contain nearly 256 positions, which will represent the occurrences of each pixel intensity. An algorithm is trained on the following data. Each histogram which is created, will be used to represent each image from the training dataset. So, given an input image, we will perform the steps again for that image and create a histogram which represents that image. The LBP histograms extracted from each sub-region are used for calculation and combined into a single, histogram with advanced features defined as:



where:

n is the number of different labels produced by the LBP operator.

m is the number of sub-regions.

and I{} is defined as follows:

Text

Description automatically generated

So, to find the image that perfectly matches the input image we only need to compare both the histograms and return the image with most matching histogram.

**Chart, histogram

Description automatically generated**

**Fig. Histogram of an Image**

We can use a lot of various approaches to compare the histograms like to calculate the distance between the two histograms for example: the Euclidean distance, chi-square, absolute value, etc. In this following example, we can use the Euclidean distance based on the following formula:

Text

Description automatically generated

If the face is detected, the algorithm basically outputs the id and name of the student having the closest histogram.The algorithm will also return the calculated distance, which can be used as a 'confidence' assessment.

Then we may apply a threshold and 'confidence' to see if the algorithm correctly recognised the image. We can assume that the algorithm has successfully recognized if the confidence is higher than the threshold defined as we are subtracting the result from 100 for better understanding. If the confidence value is too low, then the image will be marked as an unknown image and it will be stored inside the Unknown Images folder.

|  |  |
| --- | --- |
| **Person** | **Confidence level (%)** |
| Tejas | 75% |
| Atharv | 77% |
| Pritti | 72% |
| Vishal | 71% |

The system can work in conditions such as low light and can also detect and recognize the face. Here is the table in which some features of the system are mentioned:

|  |  |
| --- | --- |
| **Features** | **Results** |
| Detection of single face | Yes |
| Low light face recognition | Yes |
| Detection of Multiple faces | Yes |
| Recognition in mask/veil | No |
| Recognition in Bright Light | Yes |

## Attendance Record

As soon as we stop the face recognition process, an attendance file will be automatically generated by the system which will contain the id, name, date, and time at which each and every student was detected. Attendance is marked in the csv file for the students whose face were matched successfully with the existing database.

Table

Description automatically generated

**Fig. Attendance record**

# Limitations

The major issue with face recognition-based systems is that its accuracy is not 100%. There is always a chance of an error. Data processing in the system may take time which will eventually be an issue during the class. Depending on the quality of input data, the system would need an amount of storage. This could be very trouble causing, if the data collected is of very high quality and requires a very large amount of storage space especially for the events with a very large, expected attendance. If the resolution of the image is not high enough, it can basically cause the cameras to be tricked into believing that the person who is being scanned is not the same one as in the picture.

# Future Scope

The future scope of the work can be, capturing a number of detailed images of students and using any cloud technology to store these images. The system can be configured and can be used in Atm machines to detect the frauds.

The system can also be used at the time of elections where the voters can be identified by facial recognition.

# Conclusion

This system introduces a very efficient method of attendance management system in the classroom environment that can totally replace the old manual methods used. This method is very secure, a lot reliable, very accurate and efficient. There is no need for the specialized hardware for installing the system in the classroom. It is really easy to construct by using a camera and computer. There is a need to use some of the algorithms that can recognize the faces in the veil to improve the efficiency of the system. The system will also save a lot of time, decrease the amount of the work for the administration and will also replace the traditional stationery material with electronic apparatus and reduce the amount of human resources which are required for the purpose. Hence a system with expected results will be developed but there will be some room for improvement.

1. Acknowledgment

We would like to thank the faculty of college for giving us the opportunity to make a project on the topic of face recognition. It was a great experience and we learned a lot from it. It helped us to explore our skills and increased our interest in the field of face recognition. Special thanks to Deepali Joshi mam for being so accommodating and understanding. She is very supportive and promotes areas of our interest. She gave us a lot of encouragement, and we are very grateful to her. The environment was excellent in this group and we tried to do our best.

# References

[1] R. C. Damale and B. V. Pathak, "Face Recognition Based Attendance System Using Machine Learning Algorithms," 2018 Second International Conference on Intelligent Computing and Control Systems (ICICCS), 2018, pp. 414-419, doi: 10.1109/ICCONS.2018.8662938.

[2] A. Arjun Raj, M. Shoheb, K. Arvind and K. S. Chethan, "Face Recognition Based Smart Attendance System," 2020 International Conference on Intelligent Engineering and Management (ICIEM), 2020, pp. 354-357, doi: 10.1109/ICIEM48762.2020.9160184.

[3] S. Dev and T. Patnaik, "Student Attendance System using Face Recognition," 2020 International Conference on Smart Electronics and Communication (ICOSEC), 2020, pp. 90-96, doi: 10.1109/ICOSEC49089.2020.9215441.

[4] S. S. Pawaskar and A. M. Chavan, "Face Recognition based Class Management and Attendance System," 2020 IEEE Bombay Section Signature Conference (IBSSC), 2020, pp. 180-185, doi: 10.1109/IBSSC51096.2020.9332212.

[5] R. Hartanto and M. N. Adji, "Face Recognition for Attendance System Detection," 2018 10th International Conference on Information Technology and Electrical Engineering (ICITEE), 2018, pp. 376-381, doi: 10.1109/ICITEED.2018.8534942.

[6] T. A. Kiran, N. D. K. Reddy, A. I. Ninan, P. Krishnan, D. J. Aravindhar and A. Geetha, "PCA based Facial Recognition for Attendance System," 2020 International Conference on Smart Electronics and Communication (ICOSEC), 2020, pp. 248-252, doi: 10.1109/ICOSEC49089.2020.9215326.

[7] O. A. R. Salim, R. F. Olanrewaju and W. A. Balogun, "Class Attendance Management System Using Face Recognition," 2018 7th International Conference on Computer and Communication Engineering (ICCCE), 2018, pp. 93-98, doi: 10.1109/ICCCE.2018.8539274.

[8] R. Mehta, S. Satam, M. Ansari and S. Samantaray, "Real-Time Image Processing: Face Recognition based Automated Attendance System in-built with “Two-Tier Authentication” Method," 2020 International Conference on Data Science and Engineering (ICDSE), 2020, pp. 1-6, doi: 10.1109/ICDSE50459.2020.9310090.