

## Face recognition Attendance system using HOG and CNN algorithm

Aditya Kapse<sup>1</sup>, Tejas Kamble<sup>1</sup>, Ashutosh Lohar<sup>1</sup>, Shubham Chaudhari<sup>1</sup>, and Digambar Puri<sup>2</sup>

<sup>1</sup>Student, Department of Electronics and Telecommunication Engineering, Ramrao Adik Institute of Technology, Nerul

<sup>2</sup>Assistant professor, Department of Electronics and Telecommunication Engineering, Ramrao Adik Institute of Technology, Nerul

**Abstract.** Recognition of faces is one of the most useful applications and has a critical role in the technological field. Recognizing the face is a lively concern for authentication, specifically in the context of taking attendance. Attendance system using face recognition is a process of recognizing the profile of the person by using facial features supported by various computing technology and monitoring. The evolution of this process is focused on achieving the digitizing of the orthodox system of taking manual attendance. Current approaches for taking attendance are monotonous and inefficient. Manual records of attendance can be easily manipulated. The orthodox process of checking attendance such as current fingerprint or card scanning systems are susceptible to proxies. To tackle these issues, this paper has been proposed. The proposed system makes the utilization of various algorithms such as histogram of oriented Gradient (HOG), convolutional neural network (CNN) and support vector machine (SVM). After the face is recognized, the reports of attendance are going to be created, maintained and stored in excel format. The system is examined in various situations like illumination, head movements, and the variation of distance between the face and cameras. The proposed system was found to be efficient and reliable for marking attendance during a classroom with negligible time consumption and no manual work. This system is inexpensive as less installation is required.

### 1 Introduction

Attendance is very important for administration purposes, but usually it can become a tedious activity, with lots of inaccuracies [1]. The orthodox procedures for maintaining attendance have many limitations because it can be extremely difficult to take roll calls and maintain records when the total count of students is high [2]. Every organization has developed their own steps to take attendance. Some organizations follow a document based approach and others are using digital methods such as fingerprinting techniques and card swapping techniques. However, these methods also have some limitations because students will have to stand in long queues. If someone fails to bring his identity card, then he won't be marked present. The digital way of taking attendance is usually carried out with the help of biometric features [3]. Recognizing facial features is one such biometric way to improve digital systems for taking attendance. This method was found to be very efficient in terms of checking attendance. The usual face recognition methodologies do not handle challenges like posing, variation in light, face movements, and obstructions. The method proposed is meant to unravel the downside of present systems. Face recognition has been improved a lot till now, but the necessary stages are face detection, feature extraction, and face recognition [4]. First of all, multiple cameras depend upon the necessity and the size of the room has to be installed on the walls of the class so that it covers the entire class. Video captured from these cameras will be used by the system to get the names of the students who are present. There may be a possibility of

getting the image blurred as all the students will be moving. So an enhanced image will be passed to the system for face detection [5, 6]. The system will now first detect the faces in the image, then it will extract the facial features, and at the end it will recognize the students names using those features. Face recognition is accomplished using the CNN (Convolutional neural networks), and SVM (Support Vector Machine) algorithms. After completion of all the steps, the system gives the names of the students who are present in the class. Then attendance of the student is marked in the excel format with respective name and time of arrival. Hardware requirements of this system are less and hence it is a cost-friendly system.

### Keywords

HOG (Histogram of oriented Gradient), FLE (Face landmark estimation), CNN (convolutional neural network), SVM (support vector machine)

### 2 Literature survey

Thida Nyein, et al. [1] proposed a Face recognition attendance system using Facenet and support vector machine this system was divided into three parts first one was pre-processing of raw data in which face alignment is done on the raw data then that data is converted in training dataset in which the images are trained with model and classifier where they used facenet for feature extraction and support vector machine for classification. The flow of the

proposed system is machine starts from taking image as an input and then face detection is done by using opencv and then feature extraction and embedding is done by using Facenet and feature matching is done by using support vector machine by matching with trained dataset. After that, it records the attendance by faces and then generates an excel file for attendance records. The accuracy they were getting was 80%. This system was foundation of idea for the face recognition attendance system but the limitations of the process was that with Facenet if person is wearing any accessories then it becomes difficult for the algorithm to detect faces and the accuracy was also low.

Ketan Mahajan , et al. [2] author proposed two phase attendance system scheme were making use of Probability based face mask Pre-Filtering (PFMPF) and Pixel based hierarchical Feature Ad boosting (PBHFA) . The author were making use of this algorithms to solve the problem in Harr cascade method . The system was divided into two parts first part is training phase which has two steps first is face detection using Viola Jones algorithm and next is feature extraction which is done by using PCA algorithm. The testing part was divided into 2 section training dataset and testing dataset limitations with the system was training time was very slow and it also has high False detection rate and the system was effective when the face is in frontal view . They have not used any face centralization method.

Vidya Patil, et al. [3] proposed an automatic students attendance marking system using Kth nearest neighbour (KNN). Here, they were acquiring images from the camera, then preprocessing of the image is done by using Histogram equalization method and the face is detected using HAAR cascade algorithm and then features are extracted using the LDA algorithm and then face recognition is done by using three algorithm namely LDA, SVM and KNN .The limitations with system was KNN dont work properly when the dataset is too large and also it is sensitive to noisy and missing data. proposed system has covered all the drawbacks of this system.

In Ali Elmahmudi, et al. [4] proposed a deep face recognition system for partial faces. In this system, the methodology adopted by the author uses a VGGF model which is pre-trained and will be used for feature extraction. This system uses cosine similarity and SVM for classification. In this system, when the partial faces were present in the images, the accuracy of SVM starts dropping, but on the other hand, Cosine similarity keeps holding the accuracy.

Yohie Kawaguchi , et al. [6] has proposed a model which is based on face recognition and constant surveillance . The author has proposed a system which uses active student detecting (ASD) which has two cameras on install inside on the walls of the classroom where one was a sensing camera and other camera were used for the face detection. He has proposed a plan in which estimation of seating area is done by using ASD after that capturing student using capturing camera.The present student has been estimated using background frame subtraction and inter frame subtraction. He has resolve the issue of linear sum assignment by giving correlation of seats and students.

### 3 Proposed System

#### 3.1 Architecture

The given system is extremely effortless and manageable with coherent functioning. It has data on pupil’s face images and their name. Multiple cameras are installed on the walls of the classroom such that the entire area is covered. Video of the lecture will be obtained by using these cameras.

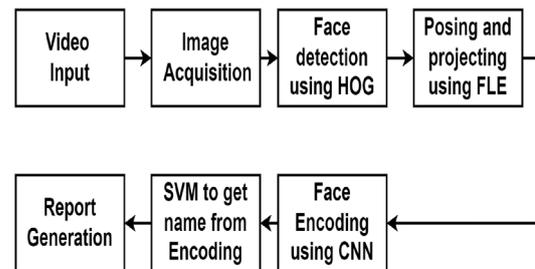


Fig 1. System flowchart

This will increase the effectiveness of our system because if one camera is not able to cover certain students then other cameras will get them. There are numerous poses which a pupil can perform. If in any case the system fails to detect faces due to the pupil’s movements, then those faces can be detected in the next iteration of image acquisition. Now, face detection has to be performed to get the faces from the acquired image. As the system detects the face it will further process it to extract facial features and then it will compare those with the facial features present in the database. Now, the name of the student whose image is matched will be updated in the excel sheet. As there will be cases where a pupil is already marked present but his face is again detected in other acquired images. So this system has a provision to make sure that each pupil will be marked as present only once during a lecture. The proposed system flowchart is shown in figure 1.

#### 3.2 System Design

In our system, there will be cameras installed on ceiling of the classroom which will take video input and pass it to the system where system will perform various operations and algorithms to generate attendance report. This system is composed of the following things:

- Image acquisition
- Face detection
- Posing and Projecting using FLE
- Face encoding Using CNN
- Comparison Using SVM
- Report generation

#### 3.3 Image Acquisition

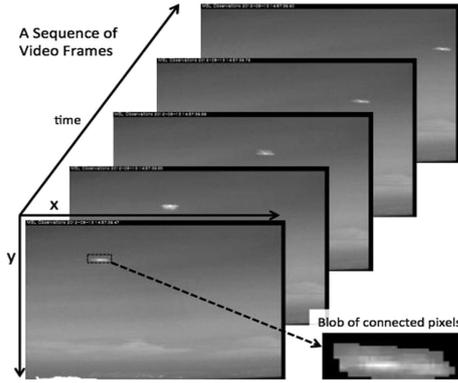


Fig 2. Image Acquisition

We all know that a video is a collection set of multiple images captured in a fraction of seconds. So we will take each frame from the video obtained by the cameras and try to find faces in it. If we found face then we will go ahead for centralization of face else we will go for the next frame of the video [7]-[8]. This process of image acquisition is shown in figure 2.

### 3.4 Face Detection using HOG

Face detection is one of the most demanding technologies in the field of machine learning. we have done face detection with the help of HOG. Here HOG features are extracted from multiple face images to be used as part of the recognition mechanism. The HOG (Histogram of Oriented Gradient) is a feature descriptor used in machine vision for processing of digital images for the purpose of detecting objects. HOG is extensively known for their use in the detection of moving objects [10]-[11].

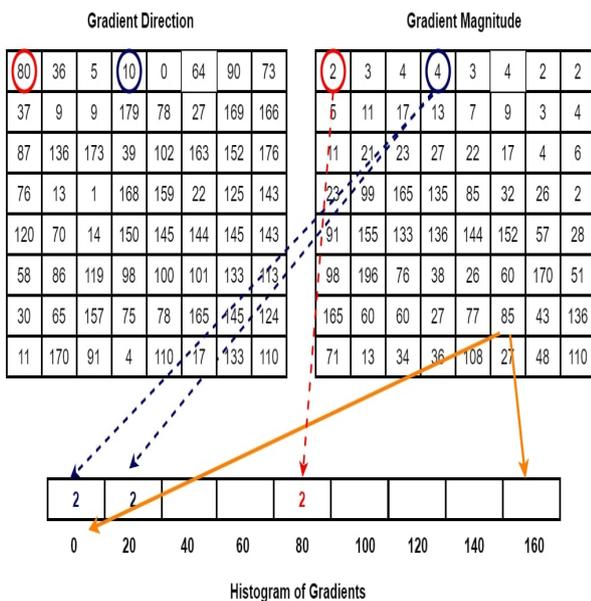


Fig 3. Feature Extraction using HOG

In Figure 3 you can see that HOG is doing feature extraction using two things first one is Gradient direction and

other is Gradient magnitude. The formulas for the calculation of Gradient and Magnitude are given in figure 4.

$$\text{Magnitude : } \|\nabla f\| = \sqrt{\left(\frac{\partial f}{\partial x}\right)^2 + \left(\frac{\partial f}{\partial y}\right)^2}$$

$$\text{Direction : } \theta = \tan^{-1}\left(\frac{\partial f}{\partial y} / \frac{\partial f}{\partial x}\right)$$

Fig 4. Formulas for Calculation of HOG

After successful calculation of Gradient direction and Gradient Magnitude for each pixel, features can be extracted as shown in figure 5.

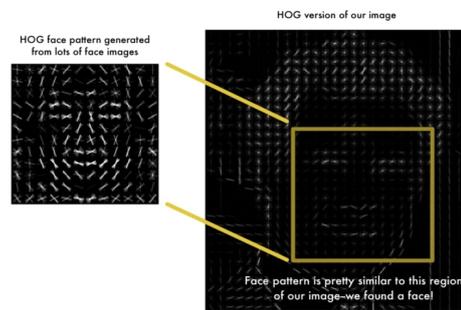


Fig 5. Image Detection using HOG

### 3.5 Posing and projecting using FLE

Head movements of students in different directions are possible during the lecture. These movements can affect the accuracy of our system. Suppose a same person is looking in different direction; then it is a totally different image for the computer as shown in the figure 6.

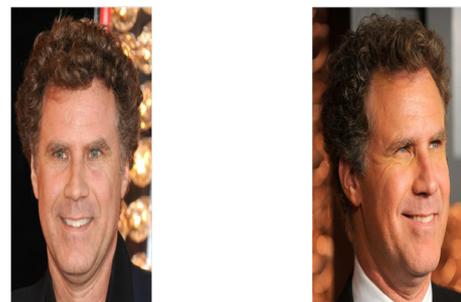


Fig 6. Image of Person Looking in Different Direction

So to overcome this issue, we will use Face Landmark Estimation Algorithm (FLE) [12]. The idea of FLE is it will compute 68 landmark points on all over the the faces like edge of both eyes, top of the chin, edge both eyebrows and then it will compute the distance between eyes, nose and the chin. It is shown in the figure 7. After putting 68 landmark points on the face, the final centralized image will be obtained as shown in figure 8. So this image will be used by CNN to generate 128 encoding out of it.

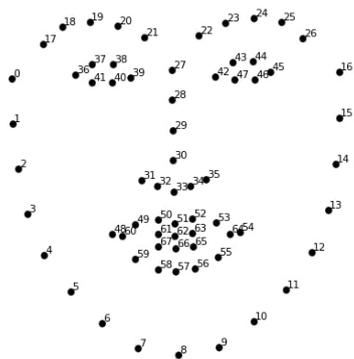


Fig 7. 68 landmark points on Face

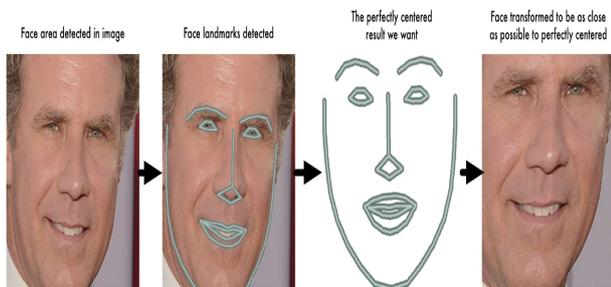


Fig 8. Centralize image using FLE

### 3.6 Face Encoding using CNN

The data set training steps are shown in figure 9 . First we will take image of a person and generate 128 encoding out of it. Afterwards we will take test image of another person and also generate encoding out of it and same with the another image; then we will compare the result and tweak the neural network slightly according to the comparison. CNN stands for Convolutional Neural Network [13]-[15]. It is a deep learning technique which is a Mask-based ap-

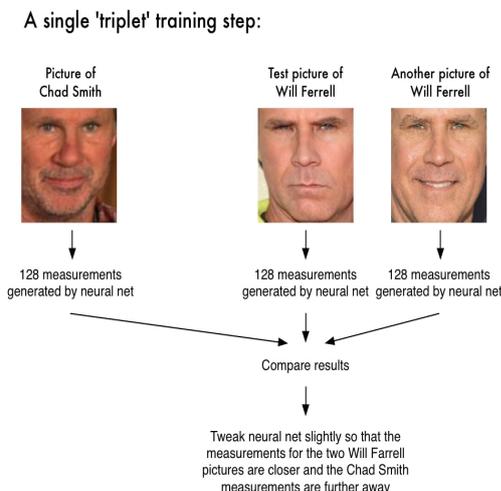


Fig 9. Training steps of Neural Network

proach. CNN is composed of many layers of neuron; the first layer is used to extract basic features, namely horizontal and vertical edges; the next layer is used to detect more

complex features such as corners; as we again move to downwards, we get more complex things such as objects, faces . We will use convolutional neural network model to extract few basic measurements from each face. We will train the model in order to generate 128 measurements for each face as shown in figure 10.

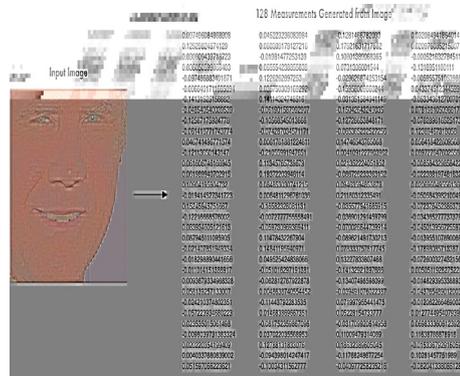


Fig 10. 128 Encoding generated by CNN

### 3.7 Comparison using SVM

Support Vector Machine (SVM) is a linear classifier. Our system will use this classifier in such a way that it can take all the measurements of unrecognized faces and it will compare them with the measurements of all the images present in our database [16]-[18]. After comparing, it will give us the name of that person whose measurements were closest match. This classifier takes milliseconds to execute. Basically, it introduces the imaginary plane between the two encoding and tries to find the closest match.

### 3.8 Report Generation

After the face recognition process is done, the attendance report will be updated and the name of the recognized student is inserted into the report along with the time at which the face was recognized. The attendance of all the students is updated only once. It means there will be only 1 occurrence of each student in the report.

### 3.9 Implementation Details

To implement the proposed system there are some hardware and software requirements.

#### 3.9.1 Hardware

This system needs:

- A high quality camera of resolution 1600x1200 or more to capture the videos [19].
- A device having operating system(Windows, Linux, MacOS), operating voltage of 19V, input voltage of 20-40V, 1 USB port, i7 11th gen or equivalent processor, minimum 8GB RAM, clock speed of 2.4GHz and storage of atleast 100GB.

### 3.9.2 software

To run this system we need :

- An IDE such as pycharm, vscode, etc to make the development process convenient.
- Python version 3.0 and higher.
- Some modules such as OpenCV (computer vision library), numpy, CSV, datetime, os and time.

### 3.9.3 Training images

- STEP1 : create a folder /train-images/ inside the open-face folder.  
`mkdir train-images.`
- STEP 2: create a subfolder for every single person you want to recognize. For example,  
`mkdir ./train-images/Aditya/`  
`mkdir ./train-images/Tejas/`  
`mkdir ./train-images/Ashutosh/`
- STEP 3: make sure all images of each person are moved into the correct sub-folders. Also make sure only one face appears in each image. The images don't need to be cropped. It is done by OpenFace automatically.
- STEP 4: From inside the openFace root directory run the openFace scripts:
  - First, do pose detection and alignment:  
`./util/align-dlib.py ./train-images/ align outer Eyes And Nose ./aligned-images/ -size 96`  
 This will create a new ./aligned-images/ subfolder with a cropped and aligned version of each of your test images.
  - Second, generate the representations from the aligned images:  
`./batch-represent/main.lua -outDir ./generated-embeddings/ -data ./aligned-images/`  
 After you run this, the ./generated-embeddings/ sub-folder will contain a csv file with the embeddings for each image.
  - Third, train your face detection model:  
`./demos/classifier.py train ./generated-embeddings/`  
 This will generate a new file called ./generated-embeddings/classifier.pkl.
- STEP 5: Recognize faces Get a new picture with an unknown face. Pass it to the classifier script like this:  
`./demos/classifier.py infer ./generated-embeddings / classifier.pkl your_test_image.jpg`  
 You should get a prediction that looks like this:  
`=== /test-images/Aditya-1.jpg === Predict Aditya with 0.73 confidence.`

### 3.9.4 Implementing system

To implement the system, a camera of recommended specifications should be connected to the computer where all

the videos will be stored and all the computations will be done. The computer should have one of the specified operating system with an installed IDE, a specified python interpreter and all the required modules [20].

Once all the requirements are fulfilled then we will first build a program to create an excel file for each lecture on a particular day. Name of this excel file will be subject\_date.csv so that every day for the same subject new file will be created. This file will be stored in the corresponding teacher's folder.

Now, we have to create a program which stores the captured video and updates the names of the recognized faces into the newly created excel file. This program will first acquire images from the video then it will detect all the faces in it using HOG. If any faces are found then it will use FLE so that the faces are centralised. Now, we will pass these faces one by one in the CNN model to get encoding for each of them. Now, it will compare these encoding with the encoding of images of all the students using SVM. This will give the face distance for each comparison and then it will record the attendance of the student whose face distance was least amongst the all. It will keep repeating the whole process during the lecture and then for next lecture the previous program will generate new excel file.

## 4 Result Discussion

**Table 1.** Comparison of different algorithms

Algorithm	SVM [14]	KNN [3]	Proposed System
Overall Accuracy	80.15 %	97%	99.2 %
Overall Time Complexity (S-seconds)	480 S	124 S	120 S
Precision	0.78	0.96	0.99
Recall	0.75	0.97	0.98
F1 Score	0.764	0.962	0.992

In Table 1, we have compared different algorithms and found that SVM has the lowest overall accuracy which is 80.15% followed by KNN which has 97% accuracy while CNN has the highest accuracy among them which is 99.2%. The overall time complexity for SVM was 480 seconds which was slowest amongst all followed by KNN which was able to give time complexity of 124 seconds while CNN was fastest amongst all which was 120 seconds. precision score for svm was 0.78 out of 1 while for KNN is 0.96 and for CNN its 0.99 which is highest amongst the all. From table 1, we have observed that our proposed system is better in terms of accuracy, overall time complexity, precision and F1 score.

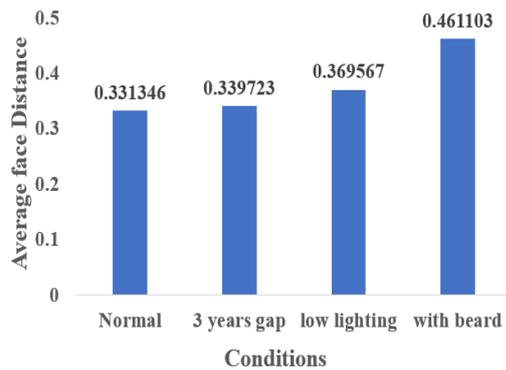


Fig 11. Variation of face distance under various conditions

The graph presented in figure 11 describes the changes in face distance in different conditions. Face distance is basically difference between the facial features of two faces. So face distance will increase, if there is some difference in the facial features of the two faces. From the graph in figure 11, we can see that there is slight increase in face distance if the lighting condition is bad. Also there is a significant increase in face distance if two faces of same person but one face with beard and other without beard is compared.

**Table 2.** Classification report

Measure	value
Accuracy	0.9920
Sensitivity	0.9930
Specificity	0.9908
Precision	0.9918
Negative Predictive Value	0.9921
False Positive Rate	0.0092
False Discovery Rate	0.0082
False Negative Rate	0.0070
F1 Score	0.9924
Mathews Correlation Coefficient	0.9838

The classification report of our proposed system is shown in the Table 2. Here we get the sensitivity score 0.9930 and the specificity score is 0.9908. Our system was able to get the F1 score of 0.9924.

## 5 Conclusion

The proposed system is able to achieve high precision and more accuracy for recognizing faces and marking attendance in less computational complexity. The economical and physical interference of the system is also less. We have successfully used HOG and CNN algorithms. Using these advanced algorithms, we were able to achieve accuracy of 99.20% and F1 score of 0.9924. We have also observed that our systems accuracy varies with various conditions such as lightning but still we were able to achieve accuracy of 99.20% . We have observed that our proposed system is efficient than any other system using different algorithms such as KNN and SVM. Our system can be further improvised and can be used for different purposes

such as for tracking person by including features such as instead of using a single camera, we can use multiple cameras at different locations and interconnect them to form a network which can track a person. This system can also be configured to be used in ATM machines to keep track of the people who are using the ATM so that at the time of fraud or robbery the bank will be able to take immediate action. Also, at the time of elections this system can be used to identify the voter by recognizing his face.

## References

- [1] Nyein, Thida and Aung Nway Oo. "University Classroom Attendance System Using FaceNet and Support Vector Machine." *2019 International Conference on Advanced Information Technologies (ICAIT)* (2019): 171-176.
- [2] Ketan N. Mahajan and Nagaraj V. Dharwadkar, "Classroom Attendance System Using Surveillance Camera", *Int. j. innov. eng. res. technol.*, pp. 1-4, Mar. 2021.
- [3] V. Patil, A. Narayan, V. Ausekar and A. Dinesh, "Automatic Students Attendance Marking System Using Image Processing And Machine Learning," *2020 International Conference on Smart Electronics and Communication (ICOSEC)*, 2020, pp. 542-546, doi: 10.1109/ICOSEC49089.2020.9215305.
- [4] A. Elmahmudi and H. Ugail, "Experiments on Deep Face Recognition Using Partial Faces," *2018 International Conference on Cyberworlds (CW)*, 2018, pp. 357-362, doi: 10.1109/CW.2018.00071.
- [5] 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.
- [6] Yohei Kawaguchi, Tetsuo Shoji, "Face Recognition-based Lecture Attendance System", 2005.
- [7] Hubballi, Kirti Bhat, Vandana, "Attendance System for Face Recognition using GSM module", *International Journal of Computer Sciences and Engineering*, (2018), 6. 1180-1184. 10.26438/ijcse/v6i7.11801184.
- [8] S. Bhattacharya, G. S. Nainala, P. Das and A. Routray, "Smart Attendance Monitoring System (SAMS): A Face Recognition Based Attendance System for Classroom Environment," *2018 IEEE 18th International Conference on Advanced Learning Technologies (ICALT)*, 2018, pp. 358-360, doi: 10.1109/ICALT.2018.00090.
- [9] E. Varadharajan, R. Dharani, S. Jeevitha, B. Kavimathi and S. Hemalatha, "Automatic attendance management system using face detection," *2016 Online International Conference on Green Engineering and Technologies (IC-GET)*, 2016, pp. 1-3, doi: 10.1109/GET.2016.7916753.
- [10] J. -M. Guo, C. -C. Lin, M. -F. Wu, C. -H. Chang and H. Lee, "Complexity Reduced Face Detection Using Probability-Based Face Mask Prefiltering and Pixel-Based Hierarchical-Feature Adaboosting," in *IEEE Sig-*

- nal Processing Letters*, vol. 18, no. 8, pp. 447-450, Aug. 2011, doi: 10.1109/LSP.2011.2146772.
- [11] S. Z. Li, R. Chu, S. Liao and L. Zhang, "Illumination Invariant Face Recognition Using Near-Infrared Images," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 4, pp. 627-639, April 2007, doi: 10.1109/TPAMI.2007.1014.
- [12] Samal, Ashok and Prasana A. Iyengar. "Automatic recognition and analysis of human faces and facial expressions: a survey." *Pattern Recognition*. 25 (1992): 65-77.
- [13] R. Brunelli and T. Poggio, "Face recognition: features versus templates," in *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 10, pp. 1042-1052, Oct. 1993, doi: 10.1109/34.254061.
- [14] Guodong Guo, S. Z. Li and Kapluk Chan, "Face recognition by support vector machines," *Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580)*, 2000, pp. 196-201, doi: 10.1109/AFGR.2000.840634.
- [15] Puri, D., Ingle, R., Kachare, P., Awale, R., "Wavelet Packet Sub-band Based Classification of Alcoholic and Controlled State EEG Signals", *International Conference on Communication and Signal Processing (ICCASP)*, Atlantis Press, 2016, pp. 562-567, DOI:10.2991/iccasp- 16.2017.82.
- [16] Kehri V., Puri D., Awale R.N., "Entropy-Based Facial Movements Recognition Using CPVM", *Applied Computer Vision and Image Processing. Advances in Intelligent Systems and Computing*, vol 1155, 2020, Springer, Singapore. <https://doi.org/10.1007/97898115402952>.
- [17] Puri D., Nalbalwar S., Nandgaonkar A., "EEG Based Diagnosis of Alzheimer's Disease Using Kolmogorov Complexity", *Applied Information Processing Systems. Advances in Intelligent Systems and Computing*, vol 1354. Springer, Singapore. <https://doi.org/10.1007/978-981-16-2008-9-15>.
- [18] M. Swapna, Y. K. Sharma, and B. Prasad, "A survey on face recognition using convolutional neural network" in *Data Engineering and Communication Technology*. Springer, 2020, pp. 649–661.
- [19] A. El-Sawy, E.-B. Hazem, and M. Loey, "Cnn for handwritten arabic digits recognition based on lenet-5" in *International conference on advanced intelligent systems and informatics*. Springer, 2016, pp. 566– 575.
- [20] Z. Zhiyao and B. M. Di, "Multiplet selection: a practical study of multi- faces recognition for student attendance system" in *Proceedings of the 2nd International Conference on Image and Graphics Processing*, 2019, pp. 42–46.