2022 13th International Conference on Computing Communication and Networking Technologies (ICCCNT) | 978-1-6654-5262-5/22/$31.00 ©2022 IEEE | DOI: 10.1109/ICCCNT54827.2022.9984382

FACE RECOGNITION ATTENDANCE SYSTEM FOR ONLINE CLASSES

Bhargavi Poyekar*, Rishita Mote†, Jahnvi Shah‡, Surekha Dholay§
Department of Computer Engineering, Sardar Patel Institute Of Technology, Mumbai, India
Email: *bhargavi.poyekar@spit.ac.in, †rishita.mote@gmail.com, ‡jahnvi.shah@spit.ac.in,
§surekha_dholay@spit.ac.in

Abstract—**Due to the increasing need for online lectures due to situations like COVID-19 and various online learning platforms, there is a need for a reliable attendance system for online classes. Our system is developed for deploying an easy and secure way of taking attendance without tedious roll-calls and inaccurate participant lists. The teachers will take screenshots of students in the online meets with their video cameras on. This screenshot will be uploaded to our system which will recognize the students and generate a report of their attendance. Our system will allow students and faculty to view the attendance of each lecture and give feedback if any discrepancy is found. We collected the dataset of face images of 94 students which are then augmented to increase the dataset and then used HOG for face detection. We then applied four algorithms namely VGG-16, MobileNet, InceptionV3, and our own CNN model for face recognition. MobileNet gave us the highest accuracy of 97.14%. We, therefore, deployed this model for our website to recognize faces and generate the attendance report.**

Index Terms—**HOG, VGG Face model, CNN, MobileNet, InceptionV3.**

I. Introduction

Recently, education has changed dramatically, with the distinctive rise of e-learning, whereby teaching is undertaken remotely and on digital platforms. Research suggests that online learning has been shown to increase retention of information, and take less time, meaning the changes coronavirus have caused might be here to stay [1]. Furthermore, according to Mrinal Mohit, the Chief Operating Officer of BYJU, there has seen a 200% increase in the number of new students using its product. This research also suggests that e-learning requires 40-60% less time as compared to traditional classroom lectures. According to survey [2], higher achieving districts maintained higher daily attendance during the COVID-19 pandemic. The survey suggests that there has been an increase in students attendance those who have stable internet connection and laptop as compared to their offline attendance. This increase in attendance of default students leads to a conclusion that students are marking proxy during online lectures.

In June 2021, National Forum on Education statistics did a survey [3] on Attendance, Participation, and classroom engagements of student during online lectures for Iowa University's students. It suggests that because the traditional "present/absent" technique is not an efficient method for online attendance of students, teachers are encouraged to mark attendance based on their classroom participation and engagement. They redefined the "attendance" term as attendance should be based on students's effort put forth during lectures, his or her knowledge rather than just a finite amount of "seat time". However, this cannot be practiced as many students leave to meet in between due to loss of internet connection. Hence, attendance systems for online platforms must be based on a method that checks who is present in the meet actively. Also, students who have joined the meet does not mean that they are actually attending the lecture. There have been cases where students just join the meet and roam freely, or somebody else might be attend their lecture as a proxy to give audio-based attendance. So, there is a need for an attendance system that will take attendance by checking whether the student is actually in front of the device attending the lecture.

We tried to solve this problem using our proposed solution which is a system that allows teachers to upload screenshots taken during the online lecture with students having their camera turned on. This screenshot is then passed to our face detection model, which detects the faces using Histogram Object Gradient(HOG). These detected faces are cropped and then passed to our Face Recognition model. We implemented four algorithms, VGG-16, MobileNet, InceptionV3 and our self-built CNN model. MobileNet model gave highest accuracy of 97.14% and so we deployed it for our website. Our proposed solution gave us positive results, and can be used in online lectures.

In the following sections we have discussed: Section II presents the literature survey related to our study, Section III presents the Proposed Methodology, Section IV elaborates our Implementation, Section V mentions our results and Section V concludes our paper.

II. Literature Survey

The system proposed by R. Hartanto and M. N. Adji, [4] incorporates skin color detection technique using Haar Cascade algorithm, and Classification by LBPH algorithm. They achieved an accuracy of 98.2% for high illumination intensity and 94.7% for low intensity. But, it is implemented for offline classroom and can detect upto max face tilt of only 30 degree. S. Dev and T. Patnaik [5] implemented Harr classifiers for face detection, and KNN, CNN, SVM for face recognition by considering head movements and illumination. The accuracy of KNN was 99.27 with precision of 0.99 and recall 0.98. In [6] Bhat, Rustagi, and Purwaha implemented

Multi Task CNN for face detection during offline classes. The detected images were fetched by the database created on their own to match the face captured. They obtained an accuracy of 97% on LFW dataset and 85% on public class dataset. A study by Z. Lin and Y. Li, [7] used their own dataset with images containing 50 students, 150 students and 82 students. They captured image area and segmented and zoomed it to improve recognition. The highest recognition accuracy reached 99%. It was observed that the intervals reduced accuracy.

Database created by ORL with 400 grayscale images is used by Bai, X., Jiang, F., Shi, T., and Wu, Y. [8] They performed face detection using Adaboost Cascade classifier. Using LBP, they extracted features and trained the classifier on images and tags for face recognition. When tested with 5 images of each person, the accuracy of face recognition reached more than 95%. E. Winarno, [9] generated a dataset of 2D images. They performed Facial recognition using an approach to the development of 2D to 3D image reconstruction models using CNN and PCA feature extraction method. They obtained an accuracy of 94% using CNN-PCA for 50 objects.

Samet et. al. [10] implemented Viola-Jones face detection algorithm and Eigenfaces, Fisherfaces and LBP face recognition algorithms based on OpenCV. The accuracy calculated based on the euclidean distance were as follows: Fisherfaces-100%, LBP-100% . The gap of this paper is that students themselves have to upload their image or 2-3 secs video. The model proposed by Damale et. al. [11] present comparison of various face detection algorithms like DNN, SVM and MLP with 98%, 87% and 86.5% accuracies respectively. The feature extraction is done using PCA and LDA. In CNN, images were directly fed to the CNN module as a feature vector. The system does not handle blur image, or illumination difference, no head and face tilting. Zuo et. al. [12] created their own convolutional neural networks to extract facial features obtaining an accuracy of 99.5% on NUAA face detection dataset of China Southern Airlines and highest 99.02% accuracy on self built dataset. The system does handle blur images and cases when face/head are tilted. Salim et al. [13] perform face recognition through LBP algorithm giving an accuracy of 95 % when tested and trained on their own database. Their system does not account for liveness detection.The system used in [14] uses LBPH (Local Binary Pattern Histogram) for facial recognition and detection with an accuracy of 92% on a self generated dataset of 200 images. Their analysis is limited since it requires a regular environment and can only process one student at once. Anzar et al. [15] make use of Dlib open source software library, Linear Discriminant Analysis (LDA) process and Histogram of Oriented Gradients (HOG), achieving an accuracy of 93.38%. They have considered static weights for face recognition and ancillary modalities which makes the system unreliable.

After conducting extensive research, we found some gaps, gathered data sources and gained significant insight to proceed further. We found that some parameters like illumination intensity and low quality images are not taken into consideration. The face and head tilts of a person were not tested. We found various data sources and some authors even created their own datasets. Similarly, we created our own dataset.

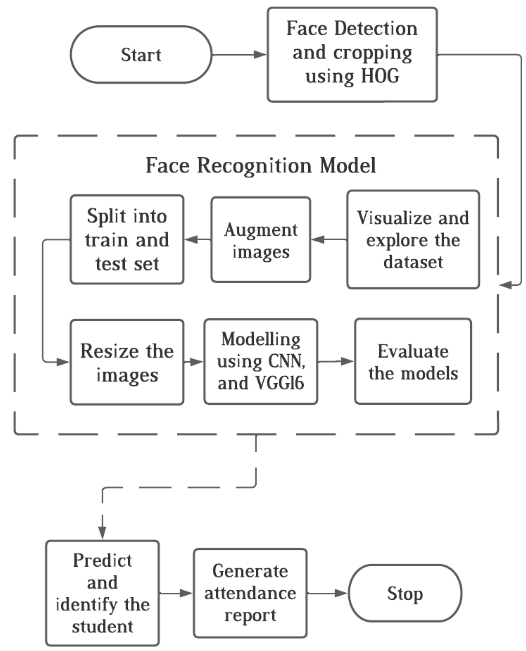

Fig. 1: Flow Diagram

III. Proposed Methodology

In this section, we have described our proposed methodology, which detects and recognizes the students from screenshots of the online lecture, uploaded by the teachers and generates the attendance report. In Fig. 1 the flow for the face detection and recognition is shown.

A. Data Collection

We collected the data of students from our online google meet lectures and laboratories. The parameters taken into consideration were various head and face tilts, different illumination intensities and different quality images. Our collected data set consists of images of 94 students each having 2-10 images in their folders with their name_surname as the folder name. In total, we have 359 images. We can see the examples of 3 students in our data set folder in Fig. 2. We divided those images into training and validation.

B. Data Augmentation

To improve the training of our model, we augmented our dataset. we performed different types of augmentation which consisted of:

1) Rotation: Rotating the images by values between -15° to 15°.
2) Adding Gaussian Noise.
3) Cropping and Padding the image.
4) Blurring image with mean over neighbour values.
5) Changing Hue and Saturation: Converting the RGB image to HSV, changing the H and S values between

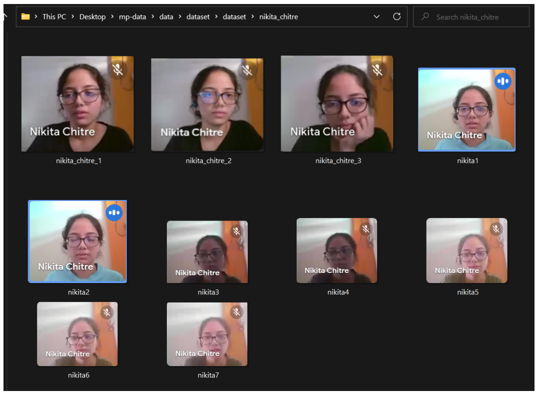

Fig. 2: Data set Example with different Illumination intensity images

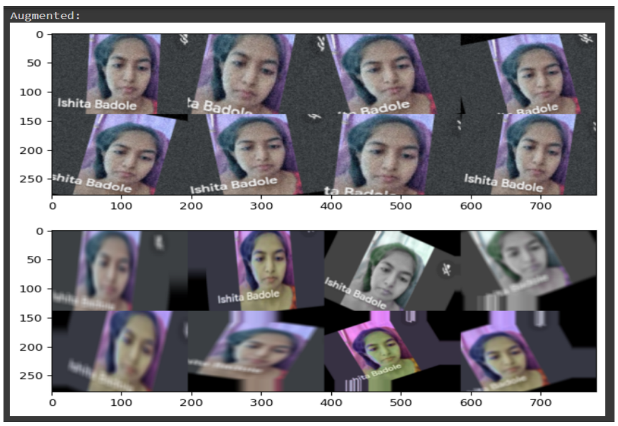

Fig. 3: Example for Augmented images

range of -60 to 60 and then converting the image back to RGB.

After augmenting the images, our dataset includes 1763 training images and 140 testing images. Fig. 3 shows the augmented images for an example image from our dataset.

C. Face Detection

In computer vision there are many techniques applied to detect faces in an image. One such powerful feature extraction method is Histogram of object Gradient (HOG). HOG is used not only for face detection but also to detect objects such as car, vegetables, fruits, etc. Step 1 in HOG is that it divides the image into small number of cells. These cells are connected to each other. In step 2 we compute histogram of each cell. Histogram is method of categorizing the data to evaluate the frequency of each features of data. In image processing, histogram represents the pixel intensity values of the image. There are 256 pixel intensities possible for an 8 bit gray scale image. The histogram displays all the 256 pixel numbers and their distribution over an image. To do so, an image is scanned and a count of of the number of pixels found at each intensity value is stored at each single pass. Step 3 is to combine all the small histograms of the connected cells and form a feature vector. This feature vector is unique for each face. This final feature vector is then used for face recognition.

D. Face Recognition

Each face detected is then recognised using our two models. We first defined a function to load the images from the extracted folder and map each image with a person id. The following sub-section elaborates on each of the models use, their architecture and accuracy and loss achieved.

1) CNN: A convolutional neural network is a deep learning model that is widely used in image classification for a large number of image dataset . It has four basic layers that are stacked over each other and an activation function to reduce the computational cost and ensuring that the model works better in complex tasks. Fig 4 shows the architecture of our CNN model. The details of each layer is as follows:

1) Convolutional Layer: This is the first layer of CNN used to extract various features of images. A filter size of M*M is defined and the input image' s pixel values are convoluted with it by dot product. Generally the size of the filter is of odd number. The more the number of filters we use the more the model extracts the features of the image. We define a stride number and based upon that the filter matrix is slided over the image. The output of this layer is termed as feature map.
2) Pooling Layer: This is a bridge between the convolutional layer and the fully connected layer. It's primary goal is to reduce the size of the feature map. There are various techniques to do so such as, average pooling, max pooling, etc. Max Pooling has shown great results in reducing the size and hence it is widely used. A filter size of two-dimensional is slided over the feature map and the max pooling layer takes the maximum value from the feature map covered within the filter size. Thus pooling layer helps in summarizing the feature map.
3) Fully Connected Layer: This layer is placed just before the output. Here the classification takes place. After getting a summarized feature map, it is then passed for flattening. The flattening layer converts the feature map into a single array known as input array for classification. Using this input array the image is classified.
4) Dropout Layer: The output of the model might overfit the data set. Hence a dropout layer is used where some of the values are randomly dropped from the network.

We have implemented to CNN model, the first one with five layers and then with three layers. For both layers we have defined a adam optimization function, categorical cross entropy as loss function and also used our own images weights. We performed 70 epochs and got a highest accuracy of 92.97%.

2) VGGNet: VGG is an advance CNN model. The details of the algorithm is explained below. We performed 50 epochs for VGG model and achieved an accuracy of 92.86%. Each convolutional layer has a filter size 3*3, stride as 1 and padding is the same. The advantage of the VGGNet over other models is that VGG has many layers and after each layer

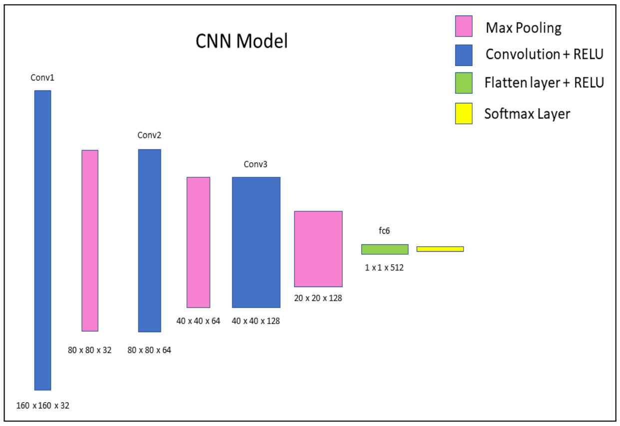

Fig. 4: Architecture of our CNN model

the image size becomes smaller that is exactly half. We have done transfer learning by freezing the weights of first four convolution blocks and trained the remaining layers i.e. three convolution layers and one max pooling layer on our dataset.

E. MobileNet

Mobilenet uses depthwise separable convolution. For this model, we have set the input shape of image as 160x160x3. We unfreezed all the layers of mobilenet for training. An average pooling layer, and dropout layer is added with output dense layer of softmax activation. While compiling, we have used categorical crossentropy as the loss function and adam optimizer with learning rate of 0.0001 which later reduces on getting a plateau by factor of 0.1 and patience of 2 epochs with final learning rate as 1e-4.

F. InceptionV3

InceptionV3 has 42 layers. We have freezed the weights of all the convolutions blocks except for the last block. The input shape of image is set as 160x160x3. The learning rate used is 0.0001.

IV. IMPLEMENTATION

Using the above proposed methodology we have created a website for Face Recognition Attendance System for online classes. Fig. 5 shows the system diagram of our proposed solution. Our System consists of 3 types of users: Admin (HOD), Teacher/Staff, and students. These 3 users have the following features:

A. Admin(Hod)

1) Login and register using the college email id.
2) Add Staff and Students details to the system.
3) View attendance report of students.
4) Add Courses and Subjects.

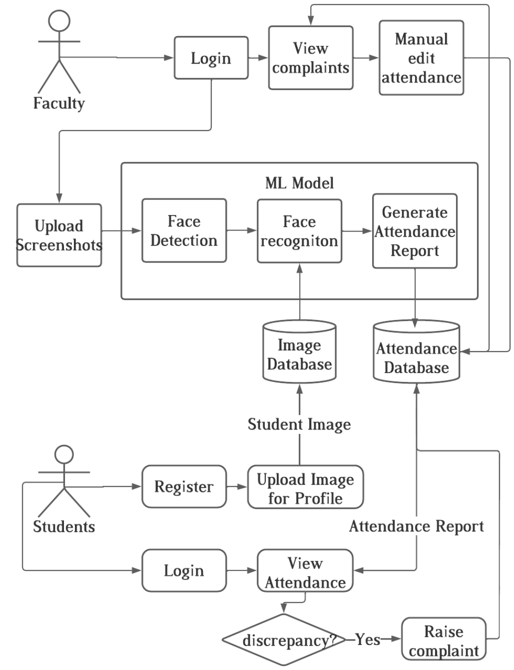

Fig. 5: System Diagram

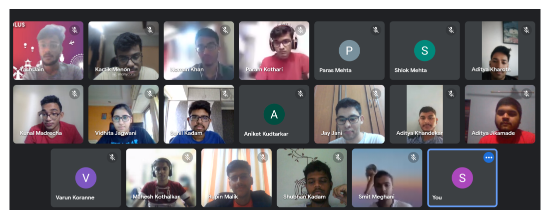

Fig. 6: Online lecture screenshot example

B. Teachers/ Staff

1) Login and register using college email id.
2) Upload multiple screenshots of online lectures. Fig. 6 shows an example of screenshot of online lecture.
3) Get attendance report of student as well as entire class. Fig. 7 shows the attendance calculated after uploading the screenshot.
4) View complaints/ feedbacks of students.
5) Manually edit the attendance of any student.

C. Students

1) Login and register using college email id.
2) View attendance during certain period of time for a particular subject
3) View attendance according to subjects for entire semester.
4) Raise complaints or give feedback about the attendance.

V. RESULTS

The data is tested on 4 different algorithms out of which MobileNet model gave the best accuracy of 97.14%. Our

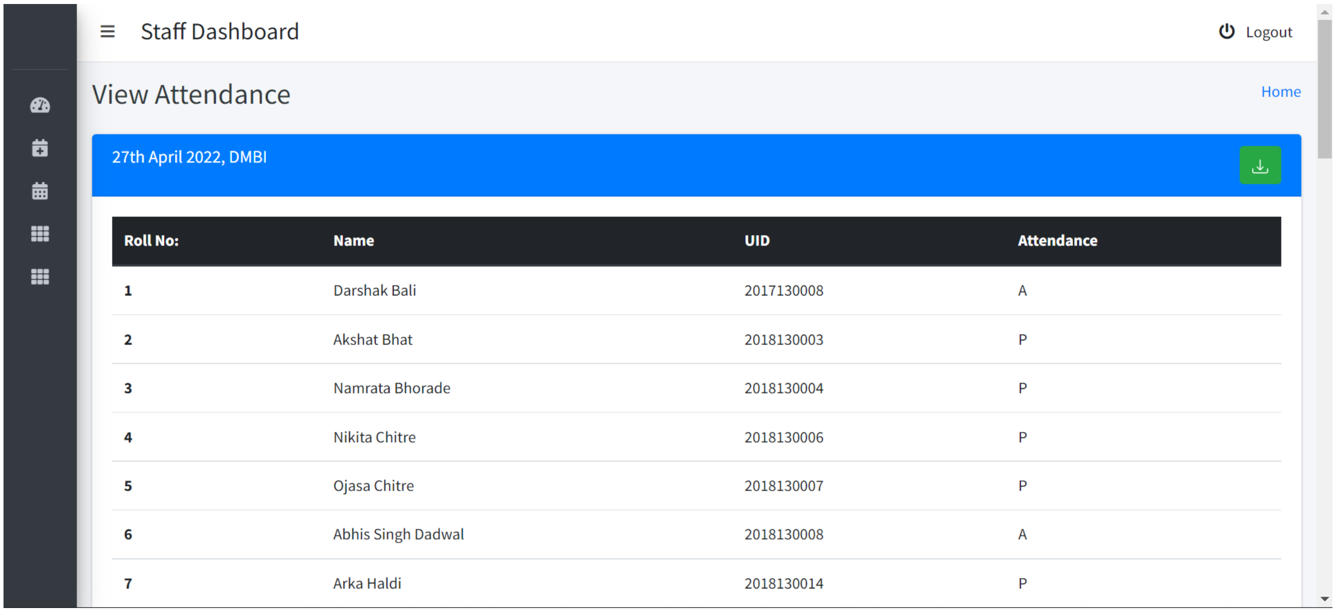

Fig. 7: Teacher Dashboard View Attendance

Model	Epochs	Learning Rate	Optimizer
Our CNN Model	70	0.001	Adam
VGG16 Model	34	0.0001	SGD
MobileNet Model	70	0.000001	Adam
InceptionV3 Model	30	0.0001	RMSprop

TABLE I: Model Parameters

Model	Train Acc	Train Loss	Val Acc	Val Loss
Our CNN Model	96.82%	0.1126	92.97%	0.5403
VGG16 Model	99.94%	0.0160	92.86%	0.4118
MobileNet Model	98.64%	0.0668	97.14%	0.2027
InceptionV3 Model	96.77%	0.0997	89.29%	0.4906

TABLE II: Comparison of Accuracy and Loss.

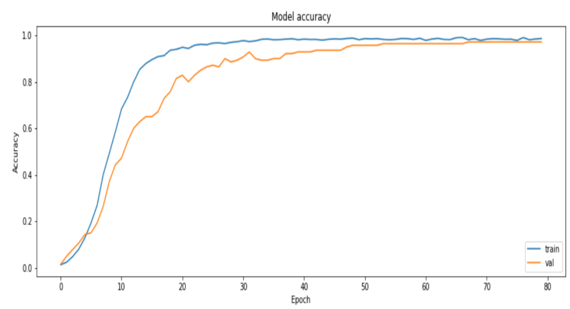

Fig. 8: Accuracy plot for MobileNet model

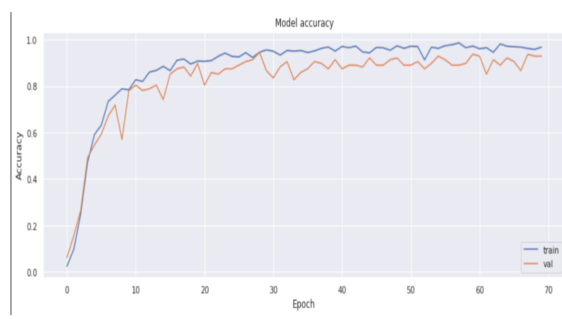

Fig. 9: Accuracy plot for our CNN model

dataset consisted of different illumination conditions, different image blur qualities, many face and head tilts. The augmentation of images rotated the images by 15 to 25 degrees. Table I shows the parameters of the models used. The accuracy and loss of all the algorithms is given in table II. The accuracy plot of the models namely MobileNet model and our CNN model is shown in Fig 8, and Fig 9 respectively.

The embedding vectors from VGG are of significance because using these vectors, the similarity between the faces can be found out. We use the method of calculating ”Squared L2 distance” to match 2 pairs of images. We can use this to check if this distance is beyond a certain threshold then the two images are of different people. It is seen in Fig. 10 that the distance between two different pictures of the same person is quite as low as 0.09 where as when compared to a different person, the distance increases to 0.14.

As shown in Fig. 11 the names of the people in the picture are correctly identified by the model(mentioned in the title). The classification was done using our self built CNN model.

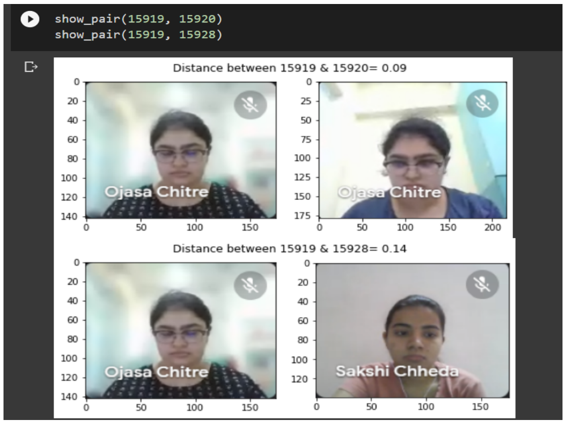

Fig. 10: Squared L2 distance between 2 pair of images

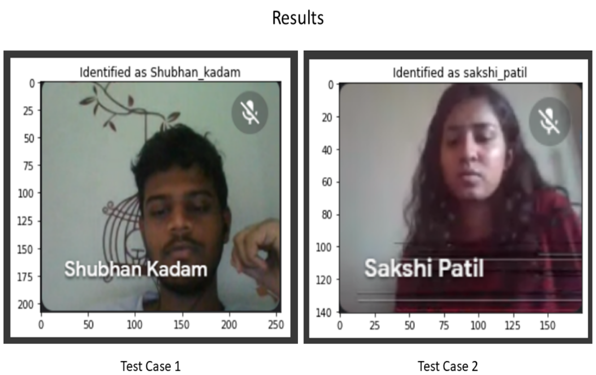

Fig. 11: Face Recognition Testing

VI. Conclusion

After implementing HOG algorithm face detection and CNN and VGG for face recognition recognition, we observed that it worked best for providing a solution to our problem statement. There are a few limitations to this project. The data we worked on was from our college's online lectures. Our system will help professors to take attendance of their online classes in a much easier way with lesser or almost none proxies. It in ways, ensures that the students are attending the classes themselves. The system is ready and can be incorporated to the current pandemic e-learning arrangement. In future we would like to work on increasing our accuracy by increasing the dataset. We would also focus on reducing the time taken to display the attendance sheet as well.

References

[1] C. Li and F. Lalani, "The COVID-19 pandemic has changed education forever. This is how" *weforum.org,https://www.weforum.org/agenda/2020/04/coronavirus-education-global-covid19-online-digital-learning/* (accessed Apr. 29, 2020).

[2] Joanne Carminucci, Sarah Hodgman, Jordan Rickles, Mike Garet,"Student Attendance and Enrollment Loss in 2020–21", *National Survey of Public Education's Response to COVID-19* June 2021.

[3] National Forum on Education Statistics. (2021)."Forum Guide to Attendance, Participation, and Engagement Data in Virtual and Hybrid Learning Models" (NFES2021058). U.S. Department of Education. Washington, DC: National Center for Education Statistics.

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

[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] A. Bhat, S. Rustagi, S. R. Purwaha and S. Singhal,"Deep-learning based group-photo Attendance System using One Shot Learning," *2020 International Conference on Electronics and Sustainable Communication Systems (ICESC)*, 2020, pp. 546-551, doi: 10.1109/ICESC48915.2020.9155755.

[7] Z. Lin and Y. Li, "Design and Implementation of Classroom Attendance System Based on Video Face Recognition," *2019 International Conference on Intelligent Transportation, Big Data Smart City (ICITBS)*, 2019, pp. 385-388, doi: 10.1109/ICITBS.2019.00101.

[8] Bai, X., Jiang, F., Shi, T., & Wu, Y. (2020). "Design of Attendance System Based on Face Recognition and Android Platform." *2020 International Conference on Computer Network, Electronic and Automation (ICCNEA)*. doi:10.1109/iccnea50255.2020.0003.

[9] E. Winarno, I. Husni Al Amin, H. Februariyanti, P. W. Adi, W. Hadikurniawati and M. T. Anwar, "Attendance System Based on Face Recognition System Using CNN-PCA Method and Real-time Camera," *2019 International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, 2019, pp. 301-304, doi: 10.1109/ISRITI48646.2019.9034596.

[10] R. Samet and M. Tanriverdi, "Face Recognition-Based Mobile Automatic Classroom Attendance Management System," *2017 International Conference on Cyberworlds (CW)*, 2017, pp. 253-256, doi: 10.1109/CW.2017.34.

[11] 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.

[12] Y. Zuo, W. Gao and J. Wang, "Face Liveness Detection Algorithm based on Livenesslight Network," *2020 International Conference on High Performance Big Data and Intelligent Systems (HPBDIS)*, 2020, pp. 1-5, doi: 10.1109/HPBDIS49115.2020.9130568.

[13] 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.

[14] K. Preethi and S. Vodithala, "Automated Smart Attendance System Using Face Recognition," *2021 5th International Conference on Intelligent Computing and Control Systems (ICICCS)*, pp. 301-304, doi: 10.1109/ICICCS51141.2021.9432140.

[15] S. M. Anzar, N. P. Subheesh, A. Panthakkan, S. Malayil and H. A. Ahmad, "Random Interval Attendance Management System (RIAMS): A Novel Multimodal Approach for Post-COVID Virtual Learning," *in IEEE Access*, vol. 9, pp. 91001-91016, 2021, doi: 10.1109/ACCESS.2021.3092260.