Proceedings of the International Conference on Intelligent Computing and Control Systems (ICICCS 2020)
IEEE Xplore Part Number:CFP20K74-ART; ISBN: 978-1-7281-4876-2

Convolutional Neural Network based Automated Attendance System by using Facial Recognition Domain

Shailender Kumar
Department of Computer Science & Engineering
Delhi Technological University
Delhi, India
shailenderkumar@dce.ac.in

Dhruv Kathpalia
Department of Computer Science & Engineering
Delhi Technological University
Delhi, India
kathpaliadhruv@gmail.com

Dipen Singh
Department of Computer Science & Engineering
Delhi Technological University
Delhi, India
dipen.singh.19@gmail.com

Mandeep Vats
Department of Computer Science & Engineering
Delhi Technological University
Delhi, India
mandeepvats98@gmail.com

Abstract - **This project aims to recognize faces in an image, video, or via live camera using a deep learning-based Convolutional Neural Network model that is fast as well as accurate. Face recognition is a process of identifying faces in an image and has practical applications in a variety of domains, including information security, biometrics, access control, law enforcement, smart cards, and surveillance system. Deep Learning uses numerous layers to discover interpretations of data at different extraction levels. It has improved the landscape for performing research in facial recognition. The state-of-the-art implementation has been bettered by the introduction of deep learning in face recognition and has stimulated success in practical applications. Convolutional neural networks, a kind of deep neural network model has been proven to achieve success in the face recognition domain. For real-time systems, sampling must be done before using CNNs. On the other hand, complete images (all the pixel values) are passed as the input to Convolutional Neural Networks. The following steps: feature selection, feature extraction, and training are performed in each step. This might lead to the assumption, where convolutional neural network implementation has a chance to get complicated and time-consuming.**

Keywords – Face detection, face recognition, deep learning, convolutional neural networks

I. INTRODUCTION

Face recognition is a unique technique for performing authentication biometrically. It has broad applications in areas of finance, security, and military. Face recognition has gained a lot of interest in the last few years, which has led several researchers to work for developing new techniques and improve the existing ones. Its wide range of applications appeals to researchers and keeps them driven. Face recognition can be performed on a real-time video by considering it as a sequence of frames where each frame is considered to be a single image. Before facial recognition can be carried out, we must first ensure the presence of a face in the frame. This can be done by performing face detection. In this step, the model detects the face and separates it from the image for identification, eliminating redundant data that is not required for facial recognition. This reduces the number of pixels on which the model has to work on and hence increasing the overall efficiency.

However, facial recognition [12] also faces some problems, making it very hard to perform. Various factors like pose variation, facial hair, image illumination, image background, and facial expressions affect the image, and the outcome can differ based on these characteristics. In situations where the face is not visible or hidden from the camera, the face might not even be detected. Thus, the image used as input to the model could be in different conditions as opposed to the image, which is to be examined.

We are attempting to instill this technique in universities, so that common issues like higher time-consumption during attendance, students marking proxy attendance and mass bunks during lectures can be prevented. Marking attendance in classes is an overwhelming task for the professors as it is not only time consuming, but also the students tend to mark proxy attendance, which leads to inaccurate records of attendance. Manual attendance is certainly tough for the professors, as it makes it difficult to maintain a record of the students. The conventional ways often have their difficulties. The majority of these methods lack dependability. It leads to an increasing need for better methods of attendance. This research stresses on using facial recognition as a technique for marking attendance. Real-time automated attendance monitoring without wasting teacher's precious time is the main objective of this project. Not only does this method save time, but it is also more reliable than traditional methods.

In the next section, we discuss some of the facial recognition algorithms, assessing their pros and cons. In Section III, we present our proposed model and briefly explain how it works. The experimental results obtained after testing the model are given in Section IV. Section V concludes the paper with a summary.

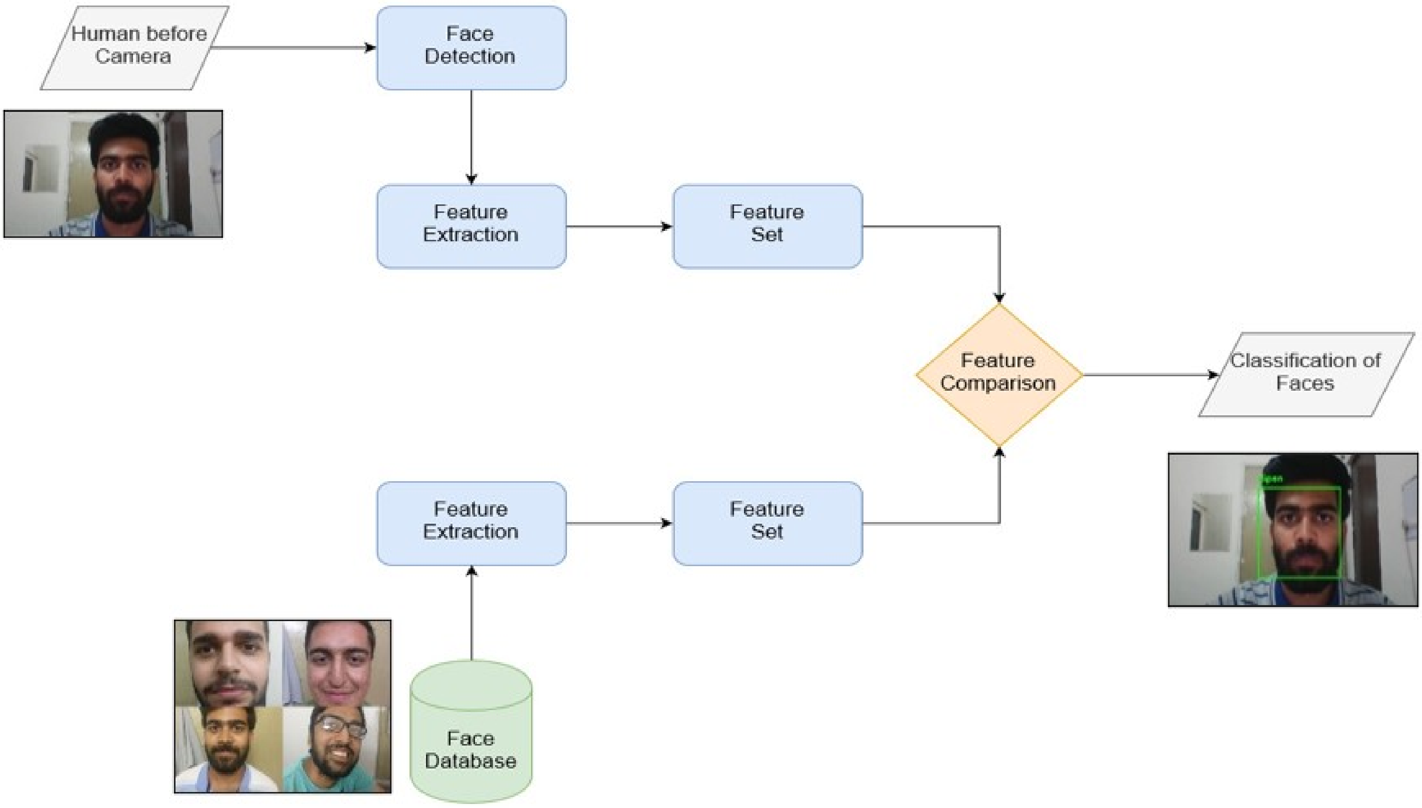

Figure 1: Flow-chart showing the working of the facial recognition aspect of our paper

II. Related Works

Dr. Priya Gupta et al. [1], obtained an accuracy of 97.05% using their proposed method. They performed feature extraction with the help of Haar Cascades, which were, in turn, fed forward to the network instead of raw pixel values. The complexity was greatly reduced as it led to a decrease in the redundant input features. It uses Deep Neural Networks, which makes the model very efficient in terms of using fewer resources and making it faster.

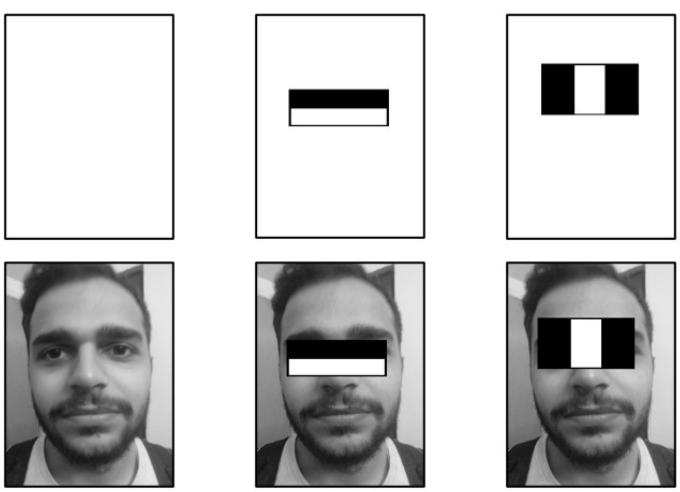

Figure 2: Haar Features applied to a Face

Haar cascades [10] use different features that are trained first with the help of a training set, which consists of both positive as well as negative images. After we have trained the classifier, it can be used to detect if the object is present in any test image or not. These have been immensely useful in tasks related to facial detection as they make the process much faster as it has considerably fewer computations as compared to other methods. Different filters can be used for features such as eyes, nose, mouth, etc. as shown in figure 1.

R. Rahim et al. [2], used the Fisher Linear Discriminant(FLD), which was discovered by Robert Fisher in 1936. It is a popular pattern recognition method, which has applications in the face and object recognition. It increases the accuracy of the classification by forming inter-class, and intra-class scatters. The algorithm can recognize the faces even with changes in some attributes of the faces, like expressions or wearing glasses.

Faizan Ahmed et al. [3], shows a comparative study on the different methods for performing facial detection and recognition. They achieved an accuracy of 96.7% using the Adaboost classifier [4] along with Haar features and an accuracy of 90.88% using Support Vector Machine(SVM) [5] classifier for Face detection application. For Facial Recognition, they used the following methods:

i. Principal Component Analysis(PCA) [6] method which uses the concept of Eigenfaces, with an accuracy of 71.15%. The computations required in this method is much lower as compared to the other methods as it only considers the 2D face recognition problem. Thus the complexity is reduced by a substantial amount.
ii. Linear Discriminant Analysis(LDA) [7] method uses Fisherfaces, with an accuracy of 77.90%. It mainly focuses on reducing the number of features being applied on each face.
iii. Local Binary Pattern(LBP) [8] has an accuracy of 82.94%.
iv. Gabor Classifier [9], which takes into consideration the local features, has an accuracy of 92.35%. It is not designed specifically for face recognition, but its filters can recognize various prominent features in an object.

Convolutional Neural Networks [11] are a type of Neural Network that is mostly used in the field of image classification, particularly Face Recognition. Convolutional Neural Networks take an input image and tweak the weights of the network-based of the input image so that it can differentiate it from other images. This allows the network to learn and identify the important characteristics (that are essential for recognizing different faces) on its own. The need for human supervision is thus minimized- it can automatically differentiate the images into separate classes. Convolutional Neural Networks also reduce the need for pre-processing required to train the model, thus it utilizes less computation power. Due to these advantages, deep learning algorithms like convolutional neural networks have become the standard in facial recognition.

III. PROPOSED WORK

We have proposed a method for an automated attendance system using facial recognition. The system should be able to detect faces in each frame of a real-time video. Further, after recognizing the detected faces, it should be able to mark the attendance of students whose faces are recognized by the system. The crux of the system is that it marks the attendance of only those students who have attended more than sixty percent of the total class; the rest of the students are marked absent.

The proposed system requires a video camera in the class to be an initial requirement. The proposed system is designed to work on video footage of students. The underlying idea is to extract features of students' faces from the footage, and compare these features with those which are extracted from the training images used for training the model. If these features match, the student is marked present for that single frame.

- A video camera placed in the classroom would continually record the class and pass the input stream to the attendance system.
- At regular intervals, individual frames are analyzed by the model.

- All the faces of the students for a particular frame would be detected and, in turn, recognized by our model.
- The same process would be followed for each frame for real-time video analysis and facial recognition.
- At the end of a class, after all the frames are done, the model would mark the students as absent or present based on the model's results.
- The student must be recognized in at least 60 percent of the frames to be marked present.
- The database would update the information on marking the attendance of all the students for that particular date.
- The database also stores the attendance record for all dates. Therefore it will also make it easier to manage the attendance records.

Pre-processing

Our model is based on deep learning-based facial recognition. We have used the face_recognition library that is a pre-trained network and uses it to generate 128-d vectors from the training dataset. Pre-processing involves the following steps:

Step 1: Face Detection

- Convert the given image to grayscale.
- Apply Haar features to each image by dividing the image into smaller squares and detect the presence of different features such as edges, corners, etc.
- We obtain a basic structure of the image, representing the obtained features.
- We can compare this structure to a previously extracted pattern of a face. This helps us in identifying the different faces present in the image.

Step 2: Resolving the issue of projecting faces

- Although we have isolated faces, the computer mistakes a face looking in different directions as different faces.
- To resolve this issue, we need to alter the positioning of a face in an image.
- We take some points that lie on a face and using these points, we manage to detect a face, its boundary, the positions of eyes, nose, and lips from an input image.

- Next, the image is rotated such that the transformed face in the rotated image is as close as possible to a perfectly centered face.
- Using this method, the computer does not categorize the projections of a face as different faces.

Step 3: Encoding Faces

- The faces that the model detected cannot be compared to each face in the database, as this method would require a lot of time.
- Instead, we have to take an approach that uses only a few metrics for each face. Metrics such as distance between the eyes or the shape of the nose aren't the most accurate in differentiating between faces.
- The metrics that are used are measurements that the network defines on its own, using Deep learning.
- Here the CNN will be trained to create 128 vectors for each face.
- Now, we take a triplet: 2 images of the same known person and one of a different person. The measurements for each image are generated.
- The network learns by adjusting so that the first two images are closer, and the measurements of the third image are farther.
- The neural network does this repeatedly for thousands of images so that for the same face, two different images give nearly the same 128 measurements.
- Huge amounts of input data and processing power is required to train a CNN using this process. However, once trained, the network can be used to generate accurate measurements, even for new faces.
- Hence, now the network (that is already trained) only requires our images to calculate the 128 measurements accurately.

Step 4: Finding a Match

- First, we detect the faces and match each of them to the encodings that we have from our training set.
- The Euclidean distance is calculated with each of the faces in our database:
  - The distance must be below some threshold value to predict that the face matched.
  - If the distance is higher than the threshold, then we consider the face to be unknown.

Processing the Input Data

The input data is the video stream from the video camera from the classroom. As the video stream is a collection of frames, we apply the face detection module to each of the frames. Once all the faces in a frame are detected, they are then compared with the generated pickle file from the training dataset. The faces that are matched are classified with the name of the student, whereas the faces that have no match are classified as unknown. The initial step of processing the data, i.e., to detect and recognize the faces in the input data, is shown in figure 3. The only step left is to mark the attendance of students whose faces are classified in the previous step.

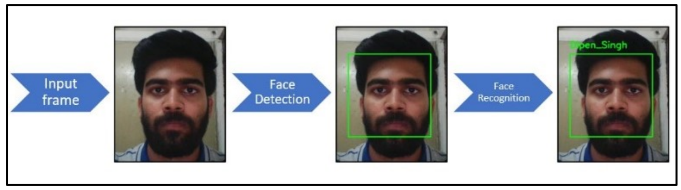

Figure 3: Steps involved in Classification of a face

The facial recognition is performed on each frame of the real-time video. So, each frame contributes to the whole attendance. Being present in only a single frame doesn't mean that the student would be marked present. This is due to the algorithm not being 100% accurate, as we have to consider situations where the students' faces are not visible. It is almost impossible to receive such high accuracy. Thus, we try and tackle the problem in a way that ensures that the students that are present in that class are only marked as present and not the ones that may be incorrectly recognized in a single frame. This also ensures that if a student's face isn't recognized in one or two frames due to any issue, they are still marked present.

The following steps are taken to ensure the above:

- We create a variable count that stores the count of the number of frames that each student's face has appeared in. We initialize this variable to 0 for each student.
- We also create a total count variable that stores the total number of frames.
- For each frame, we do the following:
  - Detect the faces in the frame.
  - Recognize the faces by matching each of them to the encodings that we have from our training set.
  - Increment the count corresponding to the student.
  - If a face is unknown, we skip that face
  - These steps are followed until we reach the last frame, i.e., the end of the class.
- Now we compare the count for each of the students to the total count. If the count for a student is greater than 60% of the total count, then we would mark the student as present in the database, else we would mark him/her as absent.
- This ensures that the student will only be marked present if he has attended at least 60% of the class.

Also, this helps in improving the accuracy of the system as even if a face is recognized incorrectly, its effect would almost be negligible on the final result.

We have proposed a method here in which the model has been trained and tested on our customized database that contains images of 16 students as subjects. To find an effective number of images per person required for our model to provide the best result, we trained and tested the model with a different number of images per person. Starting from a single image of each person, we performed the testing with as many as 25 images per person. The images of every subject differ in their facial expressions, contrast, exposure, configuration, etc. We implemented the model on a 64-bit system using Python 3.6.9. For the pre-processing of input images, we have used the OpenCV package using Haar cascade and its frontal face feature.

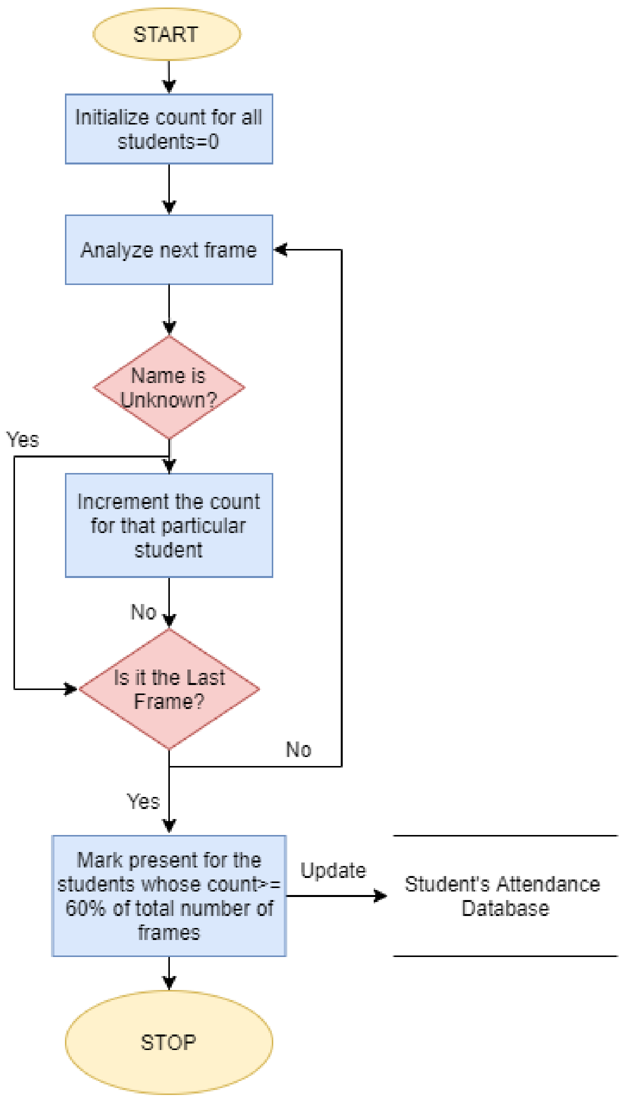

Figure 4: Flow Chart for Data Processing

IV. Experimental Results

All the images are then compressed, using a compression algorithm, to a suitable size that it contains all the important features required for recognition of faces and does not take too much time to extract the classifiers from the training images, converting them to 128-d vector and creating a pickle file. Further, the pickle file generated is used in the model to recognize the faces, instead of training the model every time the model is given an input.

We analyzed our model both quantitatively as well as quantitatively. We first assessed the number of faces the model was able to detect, i.e., quantitative analysis, when given real-time video input, and then measured the accuracy of the model by calculating the number of faces that were recognized correctly, i.e., qualitative analysis.

Also, the result of our framework, when stored in the database is sorted according to the dates on which the lecture has taken place. Initially, all the students for a lecture are marked absent and on recognizing their faces, and if they follow certain criteria mentioned above, the model updates their attendance in the database marking them present for the lecture. A snapshot of a part of a database, showing the names of only five students, is shown in Fig. 5. We have used WampServer for our MySQL database.

Date	Day	Dhruv_Kathpalia	Dipen_Singh	Mandeep_Vats	Dhruv_Ramdev	Deepanshu_Varshney
2019-10-07	Mon	A	A	A	P	A
2019-10-15	Tue	A	A	A	A	P
2019-10-23	Wed	P	P	A	A	A
2019-10-31	Thu	A	A	P	A	P
2019-11-08	Fri	P	P	P	A	P
2019-11-13	Wed	A	P	P	P	P
2019-11-20	Wed	P	P	P	P	P

Figure 5: A snapshot of the Attendance marked in the Database

Once we get the number of faces recognized in a frame, we store the list of such students. For each student, we maintain a count that contains the number of frames in which a student's face is present. And at the end of the session, if a student's face is present in 60 percent of the total number of frames, the student is marked present. Failing to fulfill the required criterion, the student is marked absent.

Table 1: Images used for Model Training and Testing

Test Case	Images Per Person in the Training set	Total students present	Students marked present in 4 classes (average)	Accuracy (%)
Case 1	1	13	7.25	55.77
Case 2	3	13	9	69.23
Case 3	5	13	10	76.92
Case 4	10	13	10.75	82.69
Case 5	15	13	11.5	88.46
Case 6	20	13	12	92.31
Case 7	25	13	12.5	96.15

On performing training and testing with a different number of images per person, we noticed that on increasing the number of images of a person, the number of important features in the pickle file from the compressed images also increases. This further increases the probability of the system comparing and deciding the detected face correctly with the features extracted from the pickle file.

In our case for automatic attendance based on facial recognition, the system admin can take multiple images of the student at the time of registration, which would be used for the training dataset for high accuracy. This makes the proposed attendance system reliable for marking attendance of students during lectures in the universities. The trend analysis of a different number of images per person on the accuracy of the system, as observed during the testing of the system is given in Fig. 6.

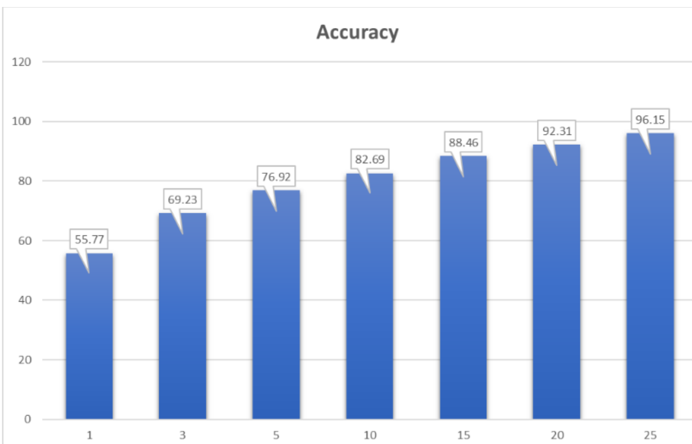

Figure 6: Graph depicting dependency on the accuracy of the model on the size of the training set

V. CONCLUSION

Using Convolutional Neural Network for facial recognition helps in reducing time and the processing power used as compared to other conventional methods. The model has great accuracy. For 25 images per subject, we achieve an accuracy of 96.15%. Although the accuracy is assumed to be very low for fewer images, it is compensated by the extra step that ensures that the student is marked present only if the number of frames their faces are identified is greater than the predefined threshold of 60%. This results in accuracy that is much higher than expected. This model can also be applied to online classes. During online lectures too, the conventional methods would waste precious time. They cannot be considered very reliable either, as anybody can log in as a student if they have the login credentials. Instead of manually taking attendance, which might be tedious for large groups of students, attendance will be taken automatically in the background. Facial recognition would ensure that attendance is reliable.

REFERENCES

[1] P. Gupta, J. Tripathi, M. Sharma, and N. Saxena, "Deep Neural Network for Human Face Recognition," *MECS,* p. 9, 2018.

[2] R. Rahim, "Research of Face Recognition with Fisher Linear Discriminant," *IOP Conf. Ser.,* 2018.

[3] A. Najam, Z. Ahmed and F. Ahmed, "Image-based Face Detection and Recognition," *IJCSI,* 2012.

[4] A. Tikare, S. Kadam, and K. T. Talele, "Efficient Face Detection using Adaboost," in *IJCA Proc on International Conference in Computational Intelligence*, 2012.

[5] B. McCane and I. Kukenys, "Support Vector Machines for Human Face Detection," in *Proceedings of the New Zealand Computer Science Research Student Conference*, 2008.

[6] M. A. Turk and A. P. Pentland, "Face recognition using Eigenfaces," in *Proceedings of the IEEE*, 1991.

[7] J. Lu, N. Plataniotis, and A. N. Venetsanopoulos, "Face recognition using LDA-based algorithms," in *IEEE Neural Networks Transaction*, 2003.

[8] T. Ahonen, A. Hadid, and M. Peitikainen, "Face recognition with local binary patterns," in *"In Proc. of European Conference of Computer Vision*, 2004.

[9] L. Wiskott, M. Fellous, N. Krger, and C. Malsburg, "Face recognition by elastic bunch graph matching," in *IEEE Trans*, 1997.

[10] P. Viola and M. Jones, "Rapid Object Detection using a Boosted Cascade of Simple Features," *IEEE Computer Society Conference on Computer Vision,* 2001.

[11] Yamashita, R., Nishio, M., Do, R.K.G. *et al.* Convolutional neural networks: an overview and application in radiology. *Insights Imaging* **9,** 611–629 (2018).

[12] "A Summary of literature review: Face Recognition" by Kittikhun Meethongjan & Dzulkifli Mohamad, August 2007, Postgraduate Annual Research Seminar 2007

[13] King, Davis E. "Dlib-ml: A machine learning toolkit." *Journal of Machine Learning Research* 10.Jul (2009): 1755-1758.