Face Recognition based Attendance Management System using Deep Learning

Akshay Dhomse
Software Developer, Infogen-Labs

Kirti Bihade
Studio partner ML/DL, Infogen-Labs

----------***----------

Now a days Face Recognition has become an in is becoming a new trending and popular when it comes to user authentication and Monitoring. I'm sure Everyone got surprised when Facebook implemented the Image auto-tagging technique. It Identify the person in the image so accurately even that picture is taken in darkness. All these successful Face Recognition systems are result of Computer vision with advancement in Deep learning models like Convolutional Neural Network(CNN). By using Face Recognition model we can uniquely identify person by comparing and analyzing face patterns based on person's unique face features. Here our goal is to track the Attendance and working hour of each employee by replacing manual method of attendance management system without human interference.

Key Words: **CNN, Face Recognition, Deep Learning, Python, Tensorflow, OpenCV, Face Detection.**

Face recognition is a technique which involves determining if the face image of any given person matches with any of the face images from a labelled stored database. Among the different biometric techniques facial recognition may not be the most reliable but it has several advantages over the others. It is widely used in various areas such as security and access control, forensic medicine, police controls and in attendance management system. The various techniques for marking attendance are:

- Signature based System.
- Fingerprint based System
- Iris Recognition
- RFID based System
- Face Recognition

Amongst the above techniques, Face Recognition is natural, easy to use and monitor, also does not require any special input from the person in front of camera.

The best way of doing Face recognition is by opting one-shot learning technique. One-shot learning aims to learn information about object categories from one, or only a few, training images. The model still needs to be trained on millions of data, but the dataset can be any, but of the same domain. In one shot way of learning, you can train a model with any face datasets and use it for your own data which is very less in number. Here we have used Facenet as a one-shot model. Face recognition system compose of two parts Face Detection and alignment using Multi-Task Cascaded Convolutional Networks which detects all faces in an image and put a bounding box, facial landmark to it which is followed FaceNet CNN model used for face Recognition(person Identification) on detected face which further matched against embeddings of the training faces of people in the database.

FaceNet is a one-shot model that transforms and learn face mapping into compact 128D Euclidean space similar to word embedding. Once this space has been produced, face recognition, verification and clustering can be easily implemented using standard techniques with FaceNet embeddings as feature vectors. To train the model, FaceNet used triplets of face patches with or without aligned. A triplet mainly consists of one anchor image, one exact same image to the anchor image and one non-similar image to the anchor image. So the triplet loss minimises the distance between an anchor and a positive, both of which have the same identity, and maximises the distance between the anchor and a negative of a different identity. where distances directly correspond to a measure of face similarity. Distance would be closer for similar faces and further away for non-similar faces.

Fig No 1.1 Learning image feature using Triplet Loss

At Infogen Labs we have build a Face Recognition system which will capture real time images of employees when they comes in front of camera with the help of computer vision techniques like openCV and then model detect position of face using MTCNN(Multi task cascad-convolutional neural network) then it will check whether that face is present inside the bounding box, if face is present inside the bounding box only then It recognize that face based using Face Recognition model i.e., Facenet. After we Recognize the face we get the Name and the present time and send it to the Employee management system.

Image dataset creation and pre-processing:

We captured images of Employees and stored in their respective name folder as a class containing Transformed images. We Transformed the image taken from webcam and performed different operation on it like- Image resize, Rotation, cropping, Image Enhancement etc.

Fig no 1.2: Image pre-processing and Enhancement

Then we create classes of different people under which we stored multiple pre-processed images of each person.

Following is the dataset Structure:

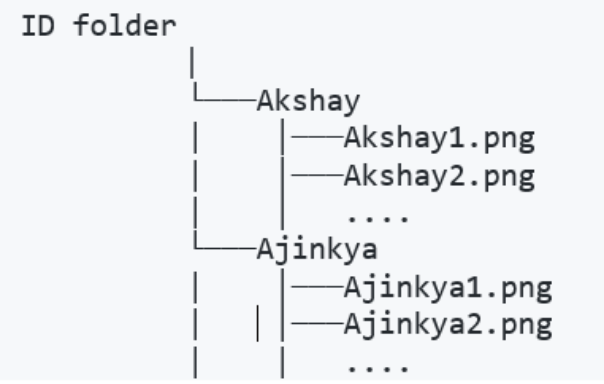

Fig no 1.3: Dataset structure

Project Model Flow:

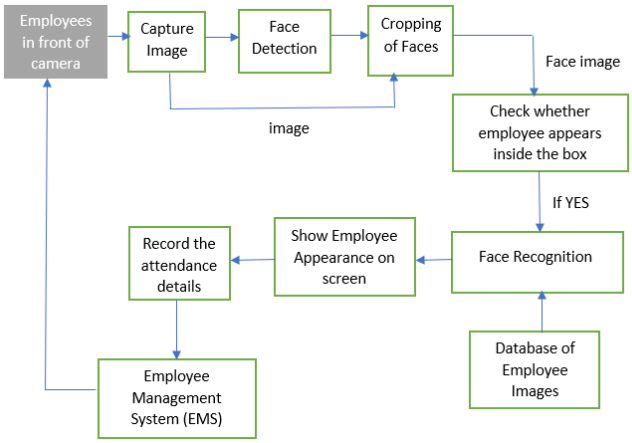

Fig no 1.4: Project model flow architecture

In face recognition it consists of two part: Face Verification and Face Recognition.

Face Verification:

It mainly deal with whether the person is present in the detected face. We need to verify the person by comparing the detected face with stored face images dataset. We used MTCNN model to detect and align the image taken from the camera. Now Verification is implemented using threshold score(an empirical value), if the score is below threshold then it is considered positive otherwise negative. A score is calculated as Euclidean distance between vector embeddings of two faces:

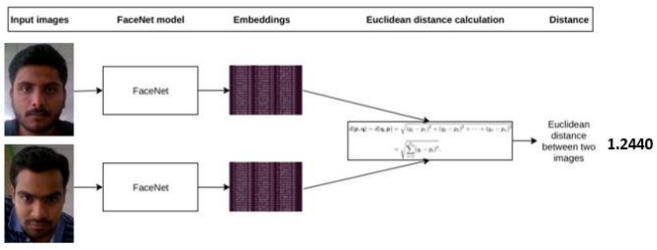

Fig no 1.5: Facenet calculating Euclidean distance between 2 distinct images.

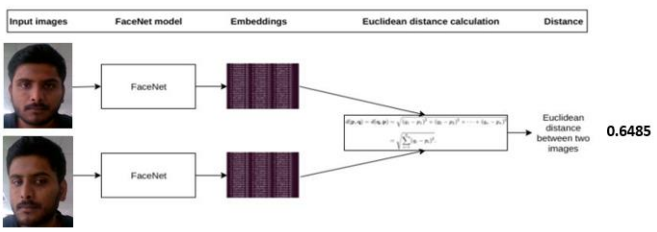

Fig no 1.6: FaceNet Calculating Euclidean distance between 2 similar images.

From fig 1.5 and fig 1.6 we understood that, a low score(0.6485) means the detected face is close to the stored historical face of person (and hence verified). Likewise, a high score(1.2440) means both the faces are different.

Face Identification:

It Identify the person on detected face in the image. Here we may need to identify a person in the detected face against an Image database of Employees.
After we Identified the face we check the appearance of that face inside the bounding box. We used OpenCV computer vision techniques to generate the bounding box and verified the appearance of that person.

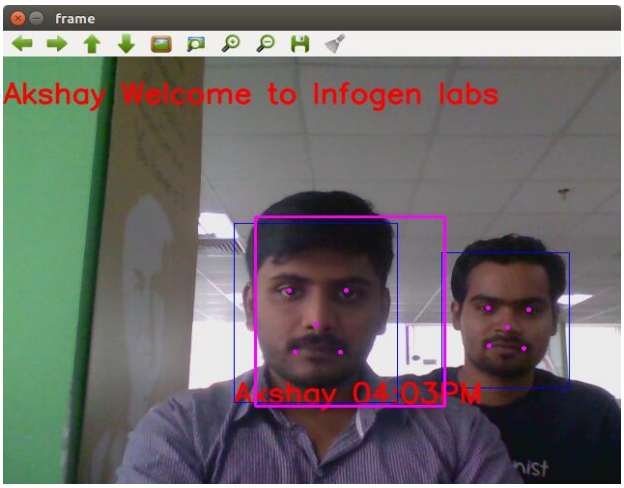

Fig no 1.7: Identify person only inside the bounding box

As we can see in above image, We Identify the person only after he/she appears inside the bounding box. By using this technique we can increase the accuracy of the identification of person.
Identification can be implemented by training a multi-class classifier over the embedding of face generated by the FaceNet model.

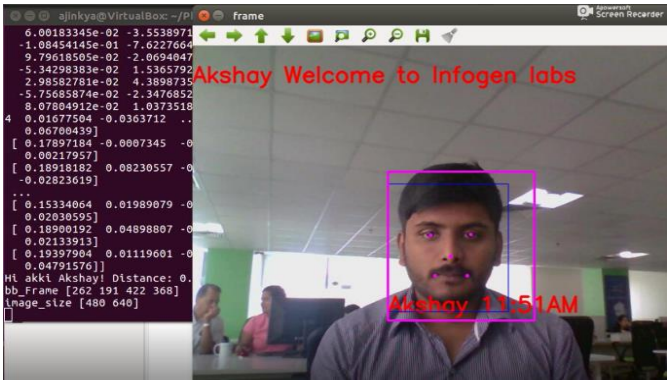

Fig no 1.8 Identifying person and displaying reporting time.

Face Verification based on Euclidean distance:

0: dataset/images/Kirti/ Kirti.png

1: dataset/images/Diksha/ Diksha.png

2: dataset/images/Akshay/ Akshay.png

3: dataset/images/Akshay/ Akshay1.png

Distance Matrix:

	0	1	2	3
0	0.0000	1.3274	1.4118	1.4822
1	1.3274	0.0000	1.5075	1.5120
2	1.4118	1.5075	0.0000	0.6942
3	1.4822	1.5120	0.6942	0.0000

Table no 1.1: Distance Matrix for comparison between 4 images

This above Table explains how the Euclidean distance between embeddings are low for two faces of the same person(image: 2&3),Higher for faces of different person(Image: 0&1).For our model we have set the Threshold of 1.1 also verified the accuracy for multiple people. In our model, the distance between extracted embeddings (FaceNet CNN model is utilized) from the bounded face are calculated.

Model connectivity to EMS:

After the model detect and recognize the person it will create the key-value pair data of the employee name along with unique employee ID and date-time will send to the EMS system.
Also we can maintain the Employee record in the form of Excel sheet:

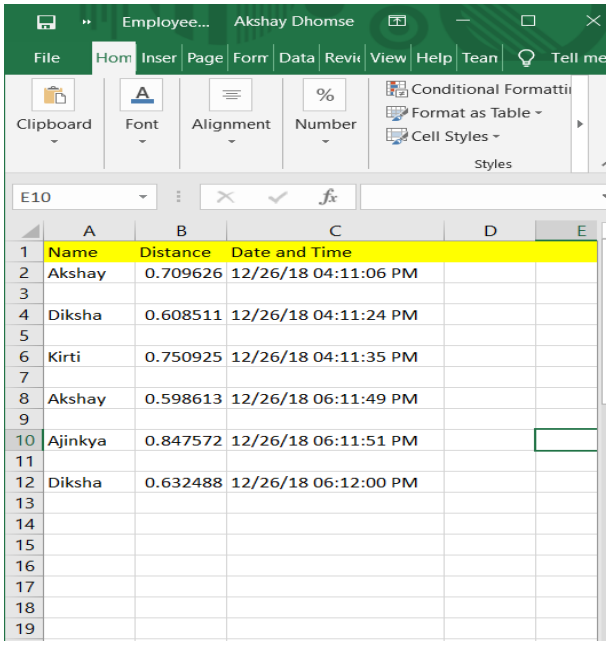

Fig no 1.9: Generation of Employee detail Report.

Conclusion:

Both MTCNN and FaceNet models are using mathematical computations behind it. As number of pixels captured in bounded face affects the recognition, So high resolution image performs better than low resolution images. To increase
theperformance of the model we can train our model for multiple frontal face images
The FaceNet is trained for extracting the features, that is to represent the image by a embedding i.,e fix length vector, so it is important to train the model for more number for images so that model performs phenomenal for real time detection of face. Eventhough we achived almost 100% accuracy during testing the model for 75 employees, but still Poor lighting conditions may affect image quality which indirectly degrades system performance.

References:

[1] FaceNet: https://arxiv.org/pdf/1503.03832.pdf

[2] MTCNN: https://arxiv.org/pdf/1604.02878.pdf

[3] https://medium.freecodecamp.org/making-your-own-face-recognition-system-29a8e728107c

[4] K.SenthamilSelvi, P.Chitrakala, A.AntonyJenitha, “Face recognition based Attendance marking system” IJCSMC, Vol. 3, Issue. 2, February 2014, pg.337 – 342