Recognition and Identification using Deep Learning

Juhi[1], Brij Mohan Singh[2], Himanshu Chauhan[3]

[1]Department of Computer Science, College of Engineering Roorkee, Roorkee, India
[2]Professor, Department of Computer Science, College of Engineering Roorkee, Roorkee, India
[3]Associate Professor, Department of Computer Science, College of Engineering Roorkee, Roorkee, India

---***---

Abstract - *Face identification and Face recognition has been a fastest spreading, extracting and fascinating area of interesting in real time applications. A huge amount of face recognition algorithms have been developed in last 10 years. Face recognition is acknowledged biometric method because of its approach. In this paper an aim is made to review a wide range of techniques used for face recognition comprehensively. This includes CNN, DNN, HOG, OpenFace and FaceNet. Once this space has been generated, tasks such as face identification, recognition, verification and cluster can be easily implemented working excellent techniques with FaceNet embeddings as feature vectors. A new approach on how to select suitable hard-negatives for training using Triplet Loss. We inspect all the procedures with parameters that challenges face recognition and identification. We organized to achieve an accuracy of 0.96 and val accuracy of 0.97 as opposed to loss of 0.24 and val loss 0.19 in the OpenFace model. In another models we archives a good accuracy to recognize the face.*

A facial recognition system is a mechanics have the ability of verifying or identifying a person from a digital pictures or a video frame from a video source. There are different methods in which face recognition systems work, principally, they work by differentiate selected facial features from a specific image with faces within a database. It is also related as a Biometric Artificial Intelligence from application that can uniquely recognize a person by inspecting patterns based on the person's facial textures and shape.

Key Words: **Face Recognition, Dlib, OpenFace, FaceNet, Deep Learning, Neural Network.**

1. INTRODUCTION

Face Recognition is a computer vision method of identifying and verifying a person based on a photograph of their face. There are different methods in which face recognition system works but mainly they work on specific selected face features from given set of images within the database.

- Face Verification is a one-to-one mapping for a given face against a known identity.
- Face Identification is a one-to-many mapping for a given face against a database of known faces.

Facial recognition is occupied in many businesses. Some uses of the face recognition system are to attendance system, unlock mobiles, tagging others on social media, payments, advertise, diagnose diseases, etc. [12]

The face recognition market is look forward to grow to $7.7 billion in 2022 from $4 billion in 2017. That is because face recognition own all kinds of commercialized applications. It can be used for everything from supervision to marketing. But that is where it turns complicated. It may used to track individuals movements out in the word like automated license plate readers track vehicles by plate numbers.

Fig 1: Face Recognition

1.1 FaceNet

It is a deep neural network used for identified features from the given image of the person's face. It was publicized in 2015 by Google Research Schroff et al. The FaceNet system can be used in general thanks to multiple third-party open source execution of the model and the accessibility of pre-trained models. The FaceNet model can be used to pull out the high-quality features from images of faces and this is known as face embeddings that can be used to train a face identification system. A deep neural network is trained by a triplet loss function that inspire vector for the matching identification to enhance more similar, whereas vectors for different identities are expected to enhance less similar. The important innovation in this work is to focus on a training model to create embeddings directly. [4]

1.2 OpenFace

We have to find a method to represent the face in numerical embedding. We can represent with the help of a pre-trained DNN OpenFace. Through the training portion of the OpenFace pipeline, 500,000 images are passed through the neural network. OpenFace instruct these images to produce 128 facial embeddings that represent a generic face. [3]

1.2.1 Triplet Loss Function: Neural networks need to be trained in such a way that embedding of anchor image and positive image should be similar and embedding of anchor image and negative image should be much farther apart. [4]

- Distance b/w anchor embedding & positive embedding = $||F(A)-F(P)||^2$
- Distance b/w anchor embedding & negative embedding =$||F(A)-F(N)||^2$

Equation, $||F(A)-F(P)||^2 — ||F(A)-F(N)||^2 \leq 0$

Now equation boils down to , $||F(A)-F(P)||^2 — ||F(A)-F(N)||^2 \leq -\alpha$

$||F(A)-F(P)||^2 — ||F(A)-F(N)||^2 \leq -\alpha$

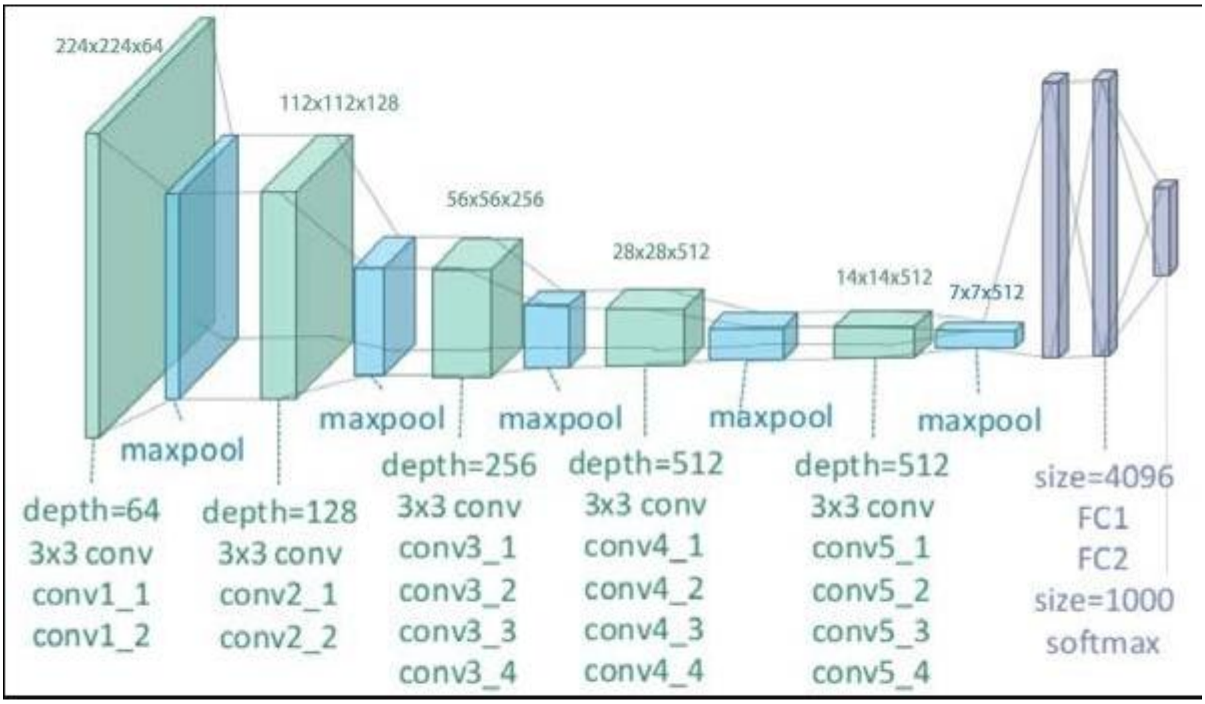
For N set of triplet images,

$$Loss = \sum_{i=1}^{N} \left[\|f_i^a - f_i^p\|_2^2 - \|f_i^a - f_i^n\|_2^2 + \alpha \right]_+$$

2. LITERATURE REVIEW

In this section we have reviewed several face recognition methods which we consider while preparing our model for detection of human face. As time passes technology advancement at its peak we have so many methods with which we can recognize face such as PCA, LDA, ICA, SVM and ANN for recognition. We are working on face recognition because of general security purpose such as using some sensor and our model we can control surveillance activity in many ways using AI techniques.[10][11][12]

Fig 2: Transfer Learning CNN



2.1 Face Recognition Using Modified OpenFace:

In this research paper by kevin et al. [1] represented a method of deep learning network using Triplet loss which becomes a common framework for person identification and verification. In this work they work on hard-negatives for training using triplet loss. They also used Adaptive movement Estimation algorithm to mitigate the risk of early convergence due to additional hard-negatives pairs. They attain accuracy of 0.955 and AUC of 0.989 as opposed to 0.929 and 0.973 in the original OpenFace.

2.2 A Review Paper on Face Recognition Techniques:

In this review paper by Sujata G. Bhele et al. [2] represented various techniques and algorithms used for face recognition. In which they focus on techniques such as PCS, LDA, ICA, SVM, Gabor wavelet soft computing tool like ANN for recognition and various hybrid combination of this techniques.

2.3 **OpenFace:** A general-purpose face recognition library with mobile applications:

In this research work done by Brandon Amos et al. [3] they focus on large accuracy gap between publicly available face recognition systems and private face recognition systems. This paper presents OpenFace face recognition library that bridges accuracy gap and uses deep neural network techniques.

2.4 **FaceNet:** A Unified Embedding for Face Recognition and Clustering:

In this research paper by Florian Schroff et al. [4] present a system called FaceNetthat directly learns a mapping from face images to a compact Euclidean space where distances directly correspond to a measure of face similarity. In this method they uses a deep convolutional neural network. For training they purpose use triplets of roughly aligned matching / non-matching face patches generated using a novel online triplet mining method. On the widely used Labeled Faces in the Wild (LFW)dataset, their system achieves a new record accuracy of 99.63%. On YouTube Faces DB it achieves 95.12%. their system cuts the error rate in comparison to the best published result [15] by 30% on both datasets.

2.5 VGG16: We can see face recognition using VGG16 by using this feature of transfer learning. Transfer learning is all about "transferring" the learnt weights to another problem. We are using the model developed for one task in another similar task.VGG16 is an advance CNN algorithm. We will load the pre-trained VGG16 model. VGG works on the data images of 224 * 224 pixels. Here we will be using the weights of imagenet. [18] [19] [20]

2.6 HOG: HOG features are labeled together for faces and support vector machine is trained to predict faces from the given images which fed into the system. Histogram of Oriented Gradients is a feature description of object detection. [14][23]

3. METHODOLOGY

In this section we describe the proposed solution as selected convolutional network architecture and discuss associated design choices, evaluation methods and implementation aspects.

3.1 DATASET:

In this work, we have collected data from different sources such as GitHub. We used a dataset from GitHub[4] respiratory by Jason Brownlee in Deep Learning for Computer Vision. In our model we used 171 images to train from different models like FaceNet, OpenFace, and many others.

Table 1: dataset

No of Images	Split Dataset	
171	136	35

3.1.1 Identification of faces from images

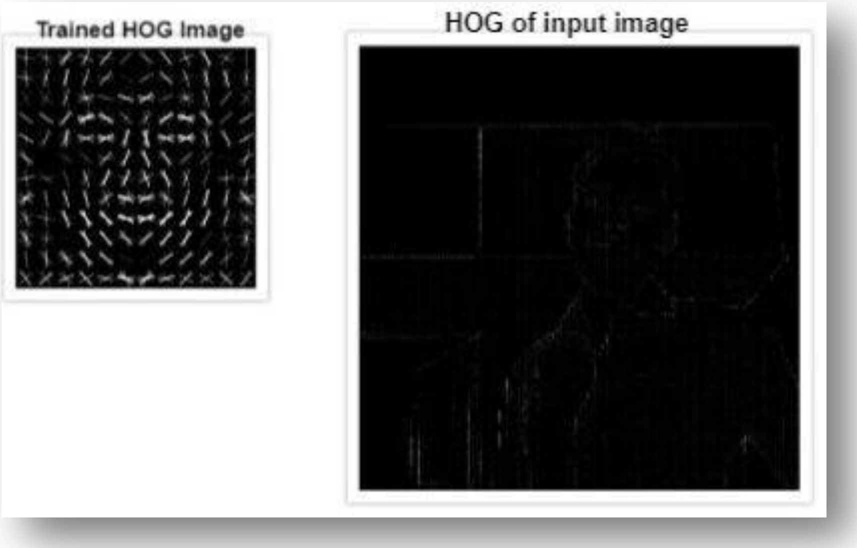
In face recognition software is to separate the actual face from the background of the image with integrate each face from others found in the image. Face identification algorithms also must be able to deal with bad and disparity lighting and various facial positions such as titled or rotated faces. We have different techniques in face identification using dlib library and haar cascade classifier.

We implemented dlib for face identification, which uses a fusion of HOG and SVM or OpenCV's or Haar Cascade Classifier. All are trained on positive and negative images. For embedding for extract face, we use OpenFace implementation which uses Google's FaceNet planning which gives better output using the dlib library.

3.1.2 Identification of face using Dlib library

Dlib for face detection and identification uses a combination of HOG and SVM which was trained on positive or negative images. The gradient for every single pixel gives us detail.

Fig 2 : Training dataset with HOG Algorithm



3.1.3 MTCNN

MTCNN model for face identification, the FaceNet model can be used to create a face embedding for each detected face, then we will modify SVM classifier model to predict the identity of a given face.

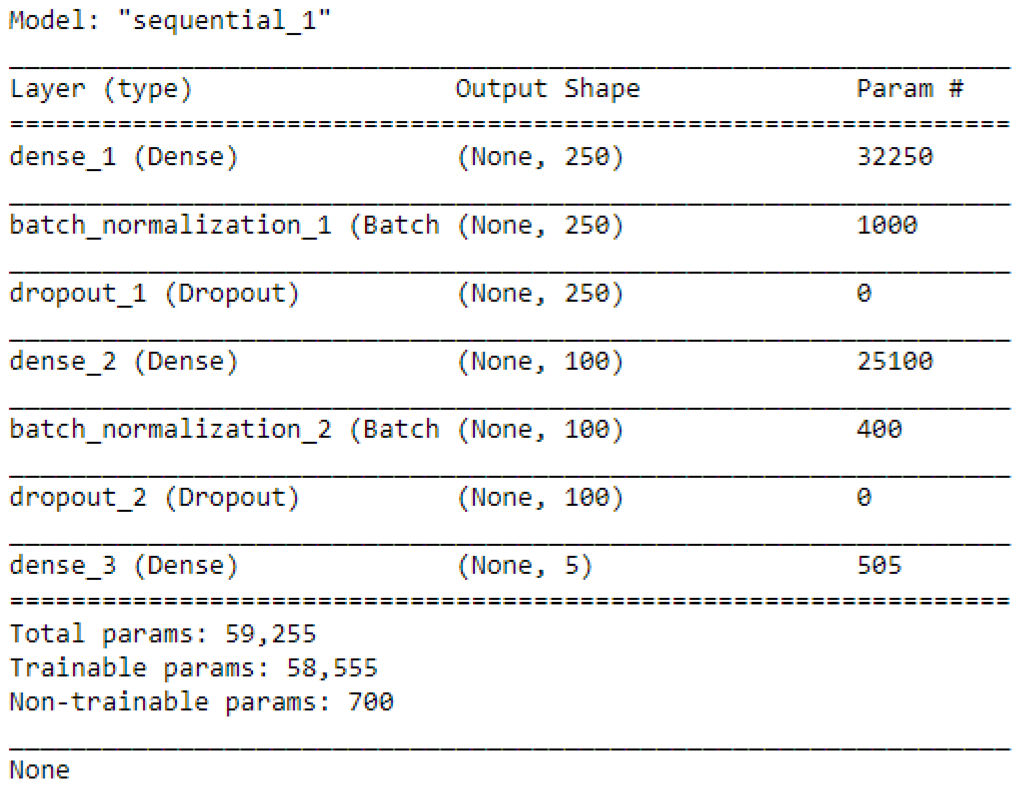
3.1.4 Training a Classifier

We have to implant each of the images in the dataset, we can train a classifier pickings face embedding as train data and name the class labels of training data. Train a model using different techniques and hyperparameter tune each of the models for the best accuracy.

3.2 Neural Network Architecture

A consecutive Neural Network model is generated using "relu" as an activation unit and implementation of weights with the normal modify. Output dimension is the number of specifiec faces of the person. Create a 2-Layer NN model adding batch management and dropout. Training batch_size of 128 and the number of epochs of 500. Input is 128-D embedding vector.

Fig 3: Neural Network Layers

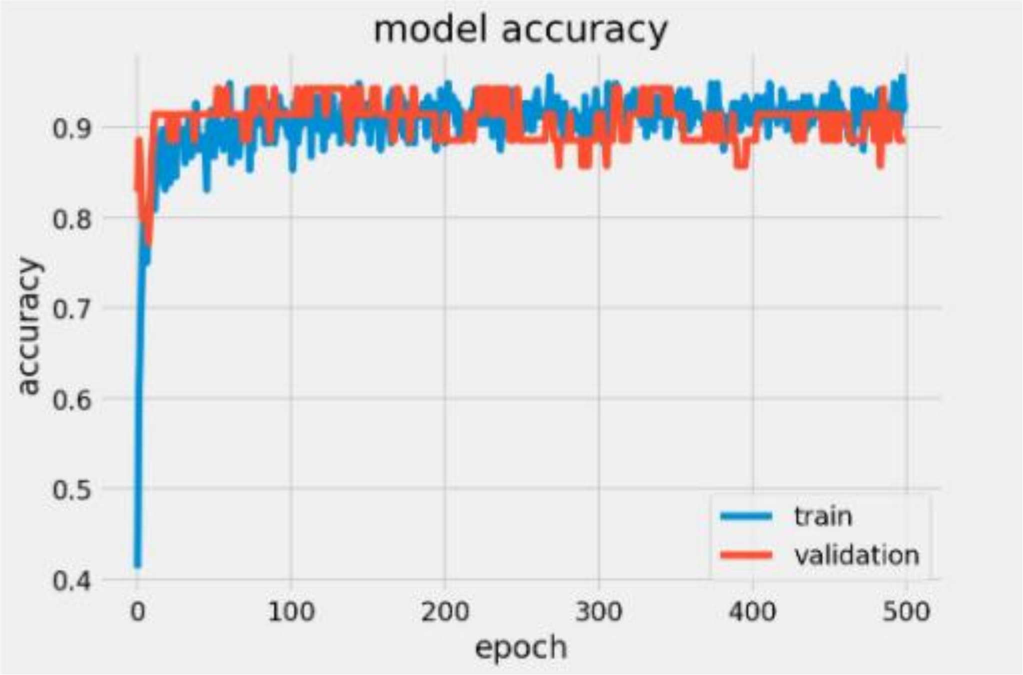


```
Model: "sequential_1"
_________________________________________________________________
Layer (type)                 Output Shape              Param #   
=================================================================
dense_1 (Dense)              (None, 250)               32250     
_________________________________________________________________
batch_normalization_1 (Batch (None, 250)               1000      
_________________________________________________________________
dropout_1 (Dropout)          (None, 250)               0         
_________________________________________________________________
dense_2 (Dense)              (None, 100)               25100     
_________________________________________________________________
batch_normalization_2 (Batch (None, 100)               400       
_________________________________________________________________
dropout_2 (Dropout)          (None, 100)               0         
_________________________________________________________________
dense_3 (Dense)              (None, 5)                 505       
=================================================================
Total params: 59,255
Trainable params: 58,555
Non-trainable params: 700
_________________________________________________________________
None
```

4. RESULT

In our studies OpenFace's accuracy and performance in comparison to different types of face recognition techniques gives a good result. In all experiments we used different types of models to evaluate. [2]

Fig 5: Model Accuracy and Loss



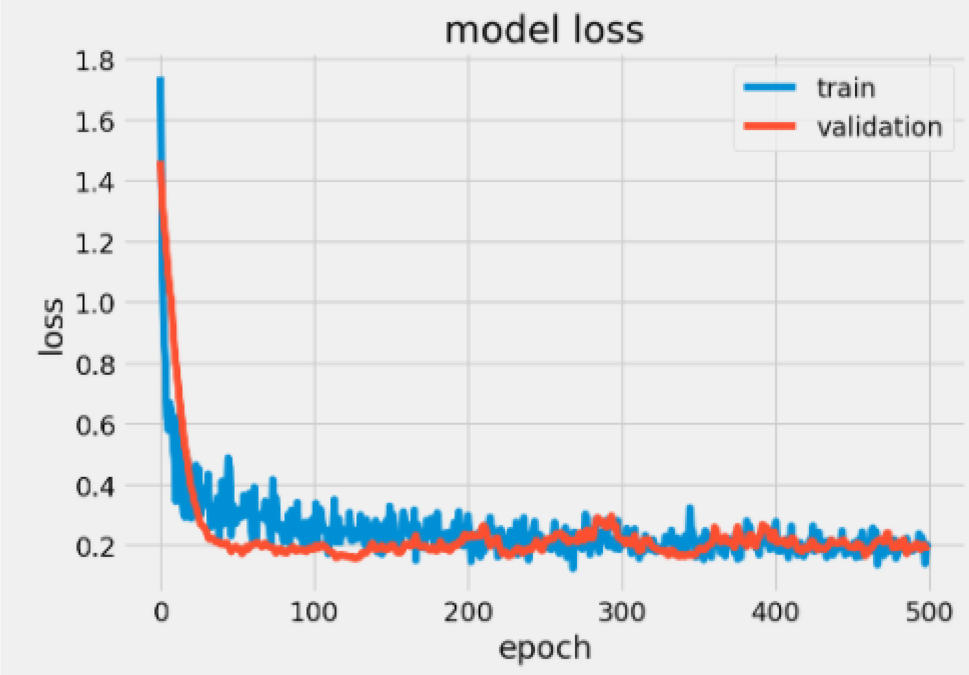
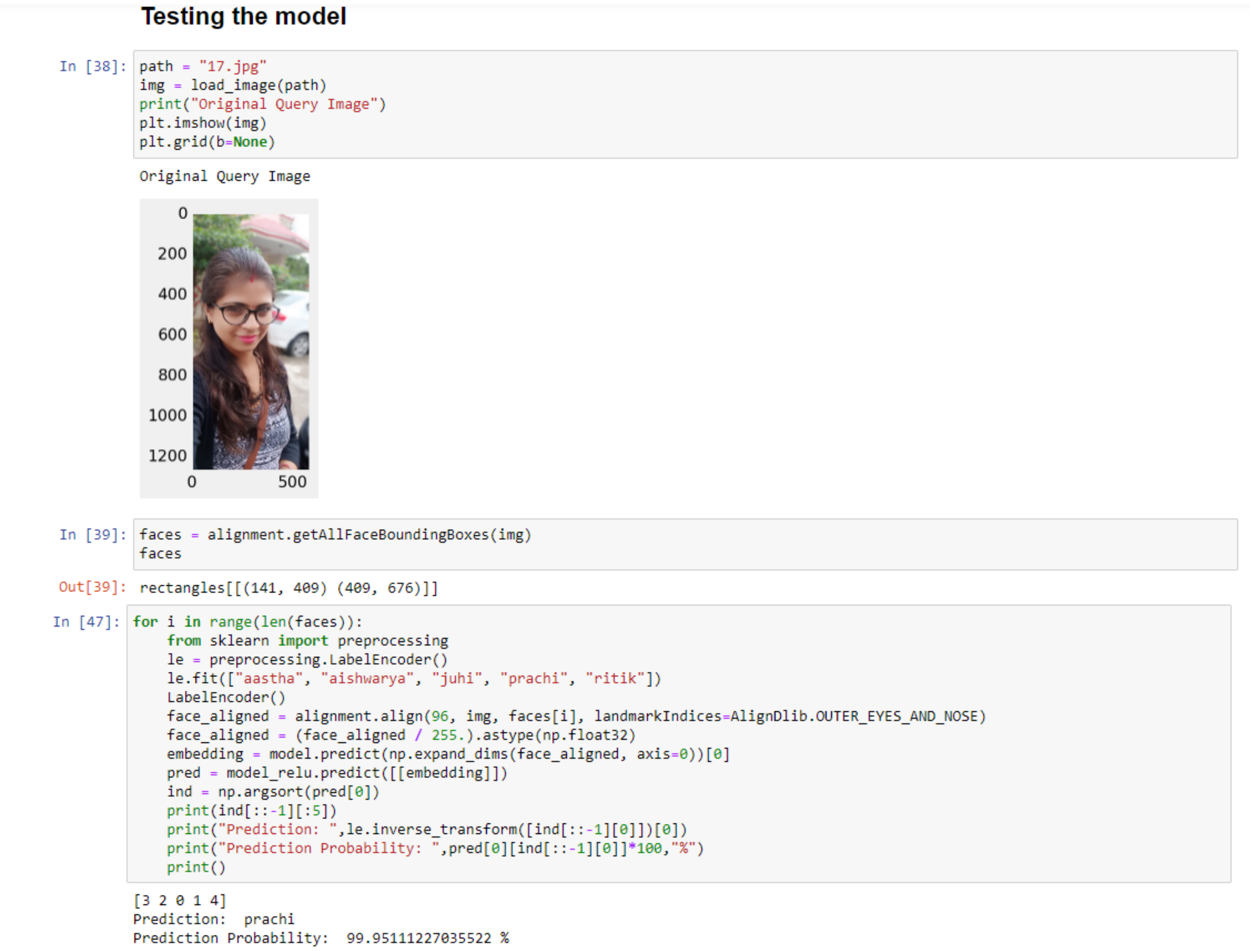


Fig 6: Result Accuracy of Model



Testing the model

```
In [38]: path = "17.jpg"
         img = load_image(path)
         print("Original Query Image")
         plt.imshow(img)
         plt.grid(b=None)

         Original Query Image
```

```
In [39]: faces = alignment.getAllFaceBoundingBoxes(img)
         faces

Out[39]: rectangles[[(141, 409) (409, 676)]]

In [47]: for i in range(len(faces)):
             from sklearn import preprocessing
             le = preprocessing.LabelEncoder()
             le.fit(["aastha", "aishwarya", "juhi", "prachi", "ritik"])
             LabelEncoder()
             face_aligned = alignment.align(96, img, faces[i], landmarkIndices=AlignDlib.OUTER_EYES_AND_NOSE)
             face_aligned = (face_aligned / 255.).astype(np.float32)
             embedding = model.predict(np.expand_dims(face_aligned, axis=0))[0]
             pred = model_relu.predict([[embedding]])
             ind = np.argsort(pred[0])
             print(ind[::-1][:5])
             print("Prediction: ",le.inverse_transform([ind[::-1][0]])[0])
             print("Prediction Probability: ",pred[0][ind[::-1][0]]*100,"%")
             print()

         [3 2 0 1 4]
         Prediction:  prachi
         Prediction Probability:  99.95111227035522 %
```

5. CONCLUSIONS

This paper presents a modification to OpenFace face recognition library. We show that the margin parameter in triplet loss function may not be considered as the absolute threshold upon which a triplet pair should be discarded. To mitigate the risk of collapsed training due to more hard-negative pairs being used, we apply the Adam algorithm.

This combination of Adam and our modification results in better accuracy and performance than existing OpenFace implementation on LFW verification benchmark. Furthermore, we also achieve a competitive accuracy and performance despite our smaller training dataset to existing method (Deepface). [8]

ACKNOWLEDGEMENT

I would like to express my deep and sincere gratitude to my guides, **Dr**. **Brij Mohan Singh**, Prof., Department of Computer Science & Engineering, College of Engineering Roorkee, and **Dr. Himanshu Chauhan**, Assoc. Prof., Department of Computer Science & Engineering, College of Engineering Roorkee, fortheir dedicated and sincere efforts and guidance towards the completion of this work providing all the possible and proper guidance.

I feel pleasure to express my thanks to all faculty members and staff of the Computer Science & Engineering Department for their valuable encouragement and timely suggestions. I also express my thanks to my batchmates, juniors, friends for their help rendered.

I would also like to express my heartfelt thanks to Miss Divya Mishra and Mr. Neeraj kumar Pandey for supporting me and also providing their valuable time throughout my MTech .

Finally, yet importantly, I would like to express my heartfelt thanks to my belovedparents for their blessings for the successful completion of this dissertation work.

Juhi

(180060705005)

REFERENCES

1. https://medium.com/analytics-vidhya/face-recognition-using-openface-92f02045ca2a
2. https://machinelearningmastery.com/how-to-develop-a-face-recognition-system-using-facenet-in-keras-and-an-svm-classifier/
3. https://medium.com/datadriveninvestor/face-spoof-detection-e0d08fb246ea
4. https://towardsdatascience.com/anti-spoofing-techniques-for-face-recognition-solutions-4257c5b1dfc9
5. https://github.com/Gradiant/bob.paper.icb2019.gradgpad
6. https://github.com/krsatyam1996/Face-recognition-and-identification
7. https://www.geeksforgeeks.org/ml-implement-face-recognition-using-k-nn-with-scikit-learn/
8. OpenFace: A general-purpose face recognition library with mobile applications Brandon Amos, Bartosz Ludwiczuk,† Mahadev Satyanarayanan June 2016 CMU-CS-16-118
9. https://www.researchgate.net/publication/270583005_A_Review_Paper_on_Face_Recognition_Techniques
10. FaceNet: A Unified Embedding for Face Recognition and Clustering Florian Schroff fschroff@google.com Google Inc. Dmitry Kalenichenko dkalenichenko@google.com Google Inc. James Philbin jphilbin@google.com Google Inc. IEEE xplore
11. 3rd International Conference on Computer Science and Computational Intelligence 2018 Face Recognition Using Modified OpenFace
12. https://towardsdatascience.com/face-detection-for-beginners-e58e8f21aad9
13. https://github.com/nyoki-mtl/keras-facenet
14. http://elijah.cs.cmu.edu/DOCS/CMU-CS-16-118.pdf
15. https://www.google.com/search?q=openFace&source=lnms&tbm=isch&sa=X&ved=2ahUKEwjL0puesOjrAhW_wzgGHfMyB6IQ_AUoA3oECA0QBQ&biw=1366&bih=657#imgrc=jhj2QB7Zfo8ZDM
16. Deep Image Similarity Measurement Based on the Improved Triplet Network with Spatial Pyramid Pooling
17. https://www.google.com/search?q=face+recognition&source=lnms&tbm=isch&sa=X&ved=2ahUKEwjysYnws-jrAhWdxjgGHXEVBFwQ_AUoAXoECA4QAw&biw=1366&bih=608
18. FaceNet2ExpNet: Regularizing a Deep Face Recognition Net for Expression Recognition Hui Ding1 , Shaohua Kevin Zhou2 and Rama Chellappa1 1 University of Maryland, College Park 2 Siemens Healthcare Technology Center, Princeton, New Jersey
19. https://medium.com/@shubhamner21/face-recognition-using-vgg16-83490759c398
20. https://www.google.com/url?sa=i&url=https%3A%2F%2Fmedium.com%2F%40tiwari.deepti262%2Fface-recognition-using-vgg16-5bcdddf3dc94&psig=AOvVaw3HmPb3hhB977MDBs3x-Bf-&ust=1600165941530000&source=images&cd=vfe&ved=0CAkQjhxqFwoTCIDmt8KW6usCFQAAAAAdAAAAABAD
21. https://medium.com/analytics-vidhya/a-take-on-h-o-g-feature-descriptor-e839ebba1e52#:~:text=These%20HOG%20features%20are%20then,the%20purpose%20of%20object%20detection.
22. https://medium.com/analytics-vidhya/a-take-on-h-o-g-feature-descriptor-e839ebba1e52#:~:text=These%20HOG%20features%20are%20then,the%20purpose%20of%20object%20detection.
23. Face Detection in real time based HOG http://ijarcet.org/wp-content/uploads/IJARCET-VOL-3-ISSUE-4-1345-1352.pdf
24. https://www.sciencedirect.com/science/article/abs/pii/S0957417417304426
25. Sensitive deep convolutional neural network for face recognition at large standoffs with small dataset Amin Jalali, Rammohan Mallipeddi, Minho Lee
26. Garg, D., Goel, P., Pandya, S., Ganatra, A., & Kotecha, K. (2018). A Deep Learning Approach for Face Detection using YOLO. 2018 IEEE Punecon.
27. https://ieeexplore.ieee.org/document/8745376
28. R. Vaillant, C. Monrocq, and Y. Lecun, “Original approach for the localisation of objects in images,” IEEE Proceedings on Vision, Image, and Signal Processing, vol. 4, 1994.
29. H.A. Rowley, S. Baluja, T. Kanade, “Neural network-based face detection”, IEEE Trans. Pattern Anal. Mach. Intell., vol. 20, no. 1, pp. 23–38, 1998.
30. C. Garcia and M. Delakis, "A neural architecture for fast and robust face detection," IEEE Trans. Pattern Anal. Mach. Intell., vol. 26, no. 11, pp. 1408–1423, 2004.
31. https://www.intechopen.com/books/reviews-refinements-and-new-ideas-in-face-recognition/dimensionality-reduction-techniques-for-face-recognition
32. https://www.google.com/search?q=facenet+face+recognition+with+webcam&source=lnms&tbm=isch&sa=X&ved=2ahUKEwjhn8icx-rrAhWFyzgGHcvWApoQ_AUoA3oECA0QBQ&biw=1366&bih=608#imgrc=lHsWBiumlPw8JM