ISSN No:-2456-2165

Survey on Face Detection Algorithms

Anushka Jadhav[1], Surabhi Lone[2], Sagarika Matey[3], Tanvi Madamwar[4], Prof. Sumitra Jakhete[5,]
Information Technology Department
Pune Institute of Computer Technology, Pune, Maharashtra, India

Abstract:- This paper consists survey of some efficient face detection algorithms. Automatic face detection is a very complex problem in image processing and many methods and algorithms have been proposed for the same. Detection of faces is done efficiently from images, video and live video streaming using face detection algorithms.There are many techniques for face detection. This survey paper includes a general review on different face detection techniques. We have tried to explain a few particularly important algorithms like Cascade Classifier, Dlib CNN, Dlib HOG, MTCNN for face detection with their working, characteristics, advantages and disadvantages.

***Keywords*:-** *Face detection, Cascade Classifier, Dlib CNN, Dlib HOG, MTCNN, deep learning.*

I. INTRODUCTION

Face detection algorithms are a very integral part of any facial analysis system depending on the ability to identify the human face part on an image. Due to increasing demand for face recognition systems in the past, face detection algorithms are used in many real - life applications. Till date, many face detection algorithms have been evolved and studied for years to get better results.

Several facial analysis systems, like facial detection, facial expression recognition, can be performed only when a face is well detected in a given image or video. However, face detection is considered as an extremely tough research task due to many challenges such as pose variation, different locations, illumination, expression variation, facial hair, presence of glasses, differences in camera gain, lighting condition, and image resolution . To resolve this problem , different methods have been proposed such as Cascade Classifier [1] , MTCNN [3] , Dlib HOG [5] , Dlib CNN[2] .

This paper aims to compare four most commonly used approaches for face detection. The paper explains all of the above mentioned methods in detail also highlighting their pros and cons followed by the experimental results and analysis performed by us. The paper is divided into three sections, in the first section all the face detection approaches namely, Cascade classifier, MTCNN, Dlib HOG, Dlib CNN are explained thoroughly. In the second section, experimental results of each approach are discussed and the third section contains the conclusion based on the obtained results.

II. ALGORITHMS

A. Cascade Classifier

Boosting which is a special case of ensemble learning states the working of boosted classifier or cascade classifier with haar-like features. It depends on Adaboost classifiers. Training of Cascade Classifier is done on a few hundred images which contain the object which needs to be detected (positive images) and other images that do not contain that object (negative images).

For live face detection, Viola-Jones proposed object detection framework in 2001 [1] which includes following steps:

1. Haar Feature Selection
2. Integral Image
3. Adaboost
4. Cascading Classifier

➢ *Haar Feature Selection*

Most of human faces have some similar features such as:

1. Bright nose bridge region is lighter compared to the eye.
2. Dark eye region is darker than the upper-cheeks.
3. Some particular regions such as the nose, mouth,eyes.

Haar-like features are convolutional kernels. Haar-features are edge features, line features and four rectangle features which are rectangles of black and white pixels. Each feature derives a single value which is obtained using subtraction of the sum of white rectangle pixels by sum of black rectangle pixels.

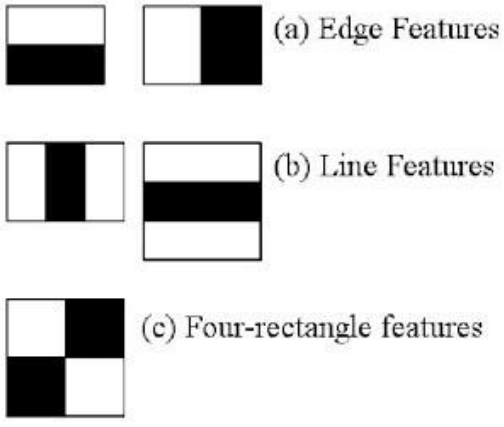

Fig 1:- Haar-like features

Plenty of features are calculated from all possible sizes and locations. So the computation required is so high even the 24*24 window gives the result of 160000 features. Which leads to introduction of integral images.

➢ *Integral Image*

Integral images simplify the calculation and computation of the sum of pixels.For rapid calculation of large sets of scales integral images are introduced. Few operations are performed per pixel for evaluation.

➢ *AdaBoost*

For selection of best features from 160000 Adaboost is used.On all training images, each and every feature is applied and for each individual feature the threshold is applied which will classify the faces to be positive or negative.A feature with minimum error rate is selected to classify its best weather face be positive or negative. Which will reduce the 160000+ features with few hundreds. Which makes Adaboost an efficient classifier which provides an algorithm of effective learning.

Paper says that 200 features provide the accuracy of 95%. Where in the final steps of adaboost 160000+ features are reduced to 6000 features.

Even when 6000 features are applied to an image it is time consuming and inefficient so they introduced the concept of Cascade Classifier.

➢ *Cascade Classifier*

So in an image most of the region is non-face area.So it is best to check whether the window is in which region, either in face region or non-face region.If it is non-face it can be discarded.Concept of Cascade Classifier states that instead of applying 6000 features on a window, grouping of features should be done as per different stages and apply them simultaneously on one-by-one stage. If the window fails at the initial stage, discard it and if it passes then the second stage is applied and the process continues. When a window passes all the stages it is a face region.Detector had 38 stages of features with 1, 10, 25 and 50 features in the first five stages.

Viola-Jones stated the algorithm has simple architecture and can detect faces at various scales which can work on CPU. But it sometimes gives a lot of false predictions and dont work on side faces and it is under occlusion.

B. Dlib CNN

The Dlib CNN face detection algorithm is a combination of Convolutional Neural network(CNN) and Dlib as the name suggests. CNN is a deep learning algorithm that is used for analyzing imagery taken as an input. Dlib is an open source library that has various machine learning algorithms to solve complex problems. The algorithm uses Dlib toolkit in addition to the CNN features to detect faces and overcomes various disadvantages of other face detection algorithms as the CNN features gives the algorithm an edge.

Including the CNN based features, Dlib CNN also uses MMOD i.e. Maximum-Margin Object Detector. The work done manually of selecting the filter in order to extract features from the images in the other approaches is automated in this algorithm. The only thing to be done is to set the number of filters to use here.

Steps of Dlib CNN:

1. Load the face detection model
2. Initialize face detector
3. Apply CNN based detector

➢ *Load the face detection model:*

The pretrained face detection needs to be loaded first here. As mentioned above, this model automates the work of selecting the filters to image inorder to extract the features. The face detection model is in a model weights file which one should get for executing the algorithm.

➢ *Initialize face detector:*

At the time of initialization, the weights file downloaded in the above step is required here. While initializing the Dlib CNN face detector the weights file is required as an input.

➢ *Apply CNN based detector:*

Once the face detector is initialized, we need to apply that detector on the test images and in turn it will give us the detected faces.

The output of CNN is very specific here, in the binary classifier format. It will return 1 if the face is there otherwise 0 for no face.

Dlib CNN works well with not only frontal faces but non-frontal faces and with odd angles whereas HOG based detectors lack. Basically, Dlib CNN detects faces in almost every angle. In this case, Dlib CNN is very much accurate for detecting faces in various orientations.

Dlib CNN is very easy to implement. As mentioned above the steps in this algorithm are very minimal, in turn making it very easy to train. It works very well with GPU rather than CPU. The algorithm works very robustly with different face occlusions.

It does not work well with the real time videos. In this case frames of videos should be provided in order to make Dlib CNN work on it. Dlib CNN has very high computation power. It does not work well with CPU The algorithm cannot detect very small faces. The minimum face size here is 80x80. So the images input having the face size less than that won't be detected.

C. Dlib HOG

This is the most widely used face detection model.it is based on HOG i.e Histogram of Oriented Gradients. It is a feature descriptor with linear SVM Machine Learning to perform face detection. HOG is a simple and powerful feature descriptor. This model built upon 5 filters i.e left looking, back looking, right looking, front looking rotated

left, front looking but rotated right. The idea behind HOG is to extract features into a vector, and suckle it into a classification algorithm Support Vector Machine, that will assess whether a face is present in a region or not. The features extracted are the distribution (histograms) of directions of gradients of the image. Gradients are typically large around edges and corners and allow us to notice those regions.

Following the flow for the Dlib HOG,

- *Normalize gamma and colour*

It is not done always , because it doesn't provide much more accuracy boost.

- *Compute gradient*

Compute the gradient of the image , to reduce dimensionality of the image , gradient captures all of the edges of the image.

- *Weighted vote into spatial and orientation cell*

Distribute the gradient in the weighted vote into spatial and oriented cells , these are the cells which can be looked as a vector containing values of histogram , it is basically separating the input into another format which is of less dimension.

- *Contrast normalize over overlapping spatial blocks*

Normalizing histogram , means normalizing contrast in the block .

- *Collect HOG's over detection window*

The image is divided into 8x8 cells , which gives proper representation of HOG.We build histogram from 64 values of gradient direction and their magnitude.The classification of histogram communicates angles of gradients from 0 to 180 degrees.

Then we find out 2 values,

1. Direction
2. Magnitude

While building HOG,we consider 3 subcases,

1. The angle is < 160 degree , not in the middle of 2 classes , that angle gets added to the right category.
2. The angle is < 160 degree , in the middle of 2 classes , then we consider twin contributions to the 2 closer classes . and split magnitudes .
3. The angle is > 160 degree , then we examine that pixel contributed proportionally to 160 degree and to 0 degree.

- *Linear SVM*

After collecting all data , repeat steps 4 and 5 , and then we can train the SVM classifier .

Dlib HOG works very well with both frontal face and a bit non frontal face , it is a light weight model , and it can work under small obstruction . But it can be really slow for some of the real time detections and sometimes doesn't work under substantial obstruction.

D. MTCNN

MTCNN- Multitask Convolutional Neural Network combines face and face key detection. It is based on cascade framework. Kaipeng Zhang et al. proposed this method in their paper 'Joint Face Detection and Alignment using Multi-task Cascaded Convolutional Networks'. MTCNN works by constructing the image pyramid of the corresponding face image.

The overall structure of MTCNN can be divided into 3 parts:-

1. P-Net (Proposal Network)
2. R-Network (Refine Network)
3. O-Net (Output Network)

- *P-Net*

When the face is detected P-Net returns the coordinates of a bounding box. It will repeat the process section wise by shifting the 12 x 12 kernel 2 pixel right or down at a time. The shift of 2 pixel is known as the stride. The faces in most of the pictures are larger than 2 pixels. So probability of missing a face by the kernel is very low.

- *R-Net*

R-Net includes the co-ordinates of the new and more accurate bounding boxes and also the confidence level of each of these bounding boxes. In this stage we also get rid of the boxes with lower confidence level and perform Non-Maximum Suppression on every box to further eliminate the boxes which are redundant.

- *O-Net*

O-Net is actually different from P-Net and R-Net. This stage proposes more supervision to mark face region. We get 3 outputs from O-Net, the coordinates of the bounding boxes, coordinates of the 5 facial landmarks, and the confidence level of each box.

Then once again the boxes with lower confidence levels have to be removed, and have to standardize both the bounding box coordinates and the facial landmark coordinates. Finally when the Non-Maximum Suppression is applied again we get the facial features of the person which is basically five facial landmark positions.

As compared to all methods of face detection MTCNN has very high accuracy. It may take time for training but we can save time by using pre-trained models and at the same time keep high accuracy. MTCNN even supports real time face detection.

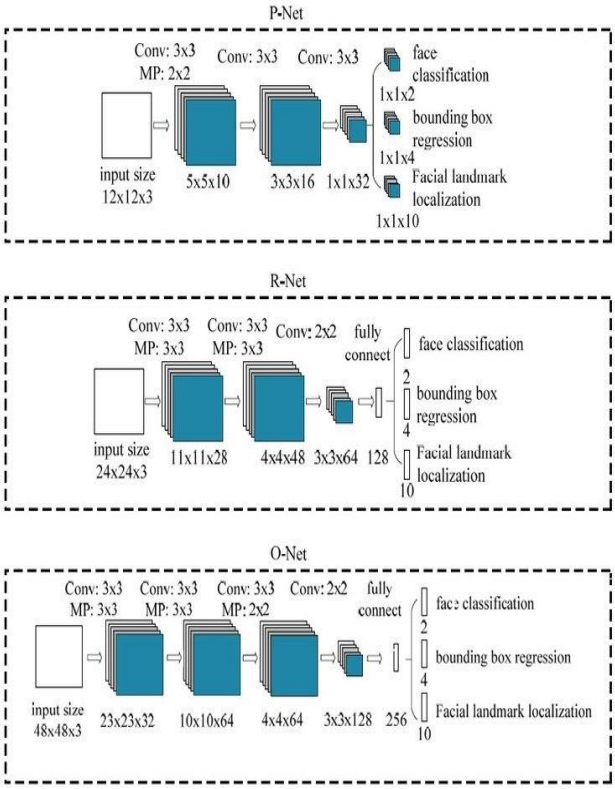

Fig 2:- Overall Structure of MTCNN

III. EXPERIMENTAL RESULTS

➢ *Experiment on Cascade Classifier with Inference Time*

Following are the images which shows output for frontal , low light , multiple and side face detection.

Fig 3:- Result for Frontal Face Detection

Fig 4:- Result for Low Light Face Detection

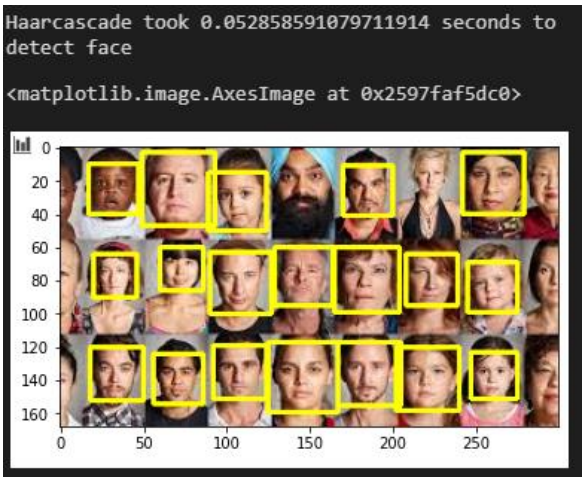

Fig 5:- Result for Multiple Face Detection

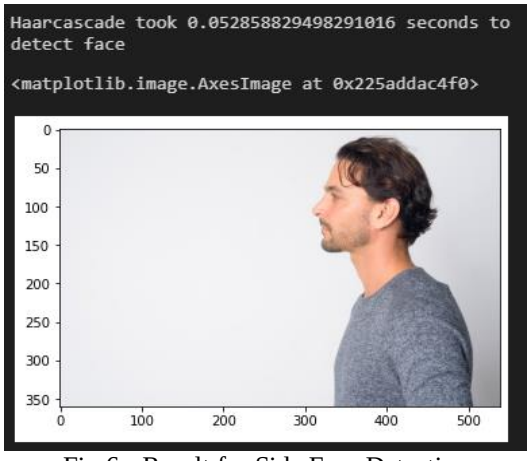

Fig 6:- Result for Side Face Detection

➢ *Experiment on Dlib CNN with Inference Time*

Following are the images which shows output for frontal , low light , multiple and side face detection

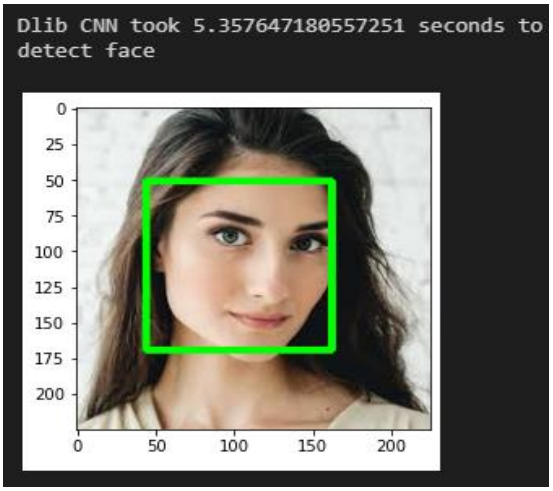

Fig 7:- Result for Frontal Face Detection

Fig 8:- Result for Low Light Face Detection

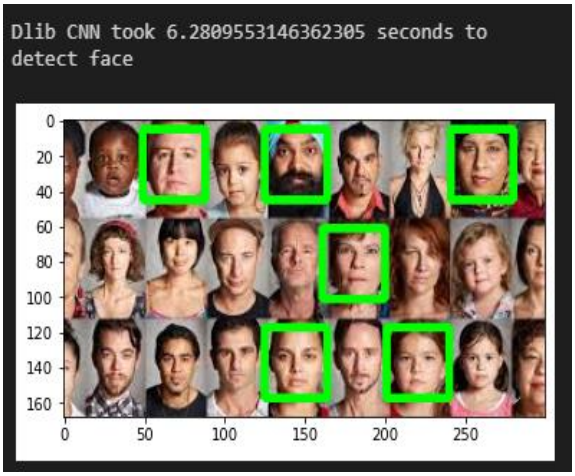

Fig 9:- Result for Multiple Face Detection

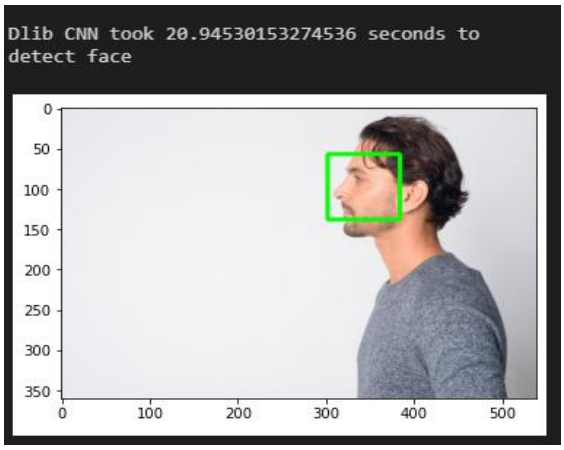

Fig 10:- Result for Side Face Detection

➢ *Experiment on Dlib HOG with Inference Time*

Following are the images which shows output for frontal , low light , multiple and side face detection

Fig 11:- Result for Frontal Face Detection

Fig 12:- Result for Low Light Face Detection

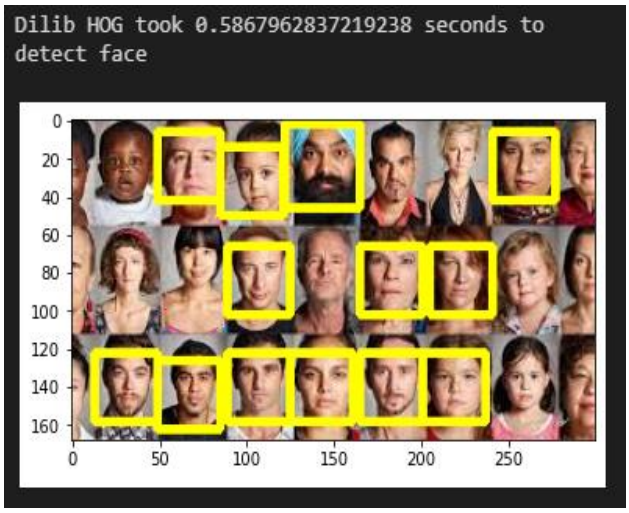

Fig 13:- Result for Multiple Face Detection

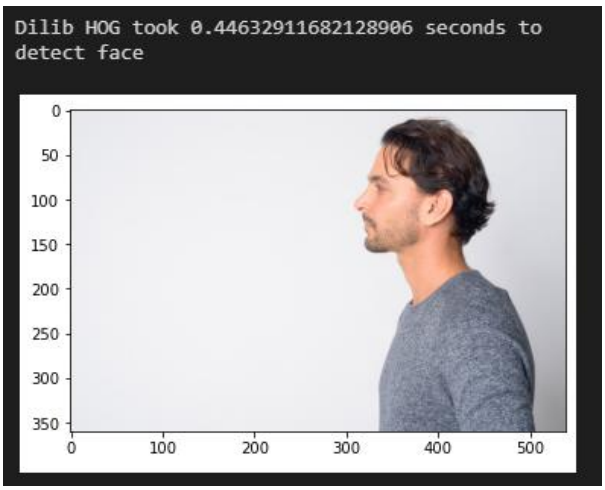

Fig 14:- Result for Side Face Detection

➢ *Experiment on MTCNN with Inference Time*

Following are the images which shows output for frontal , low light , multiple and side face detection

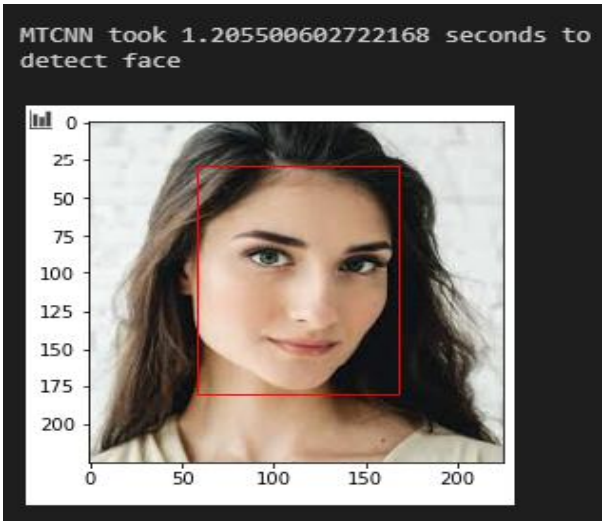

Fig 15:- Result for Frontal Face Detection

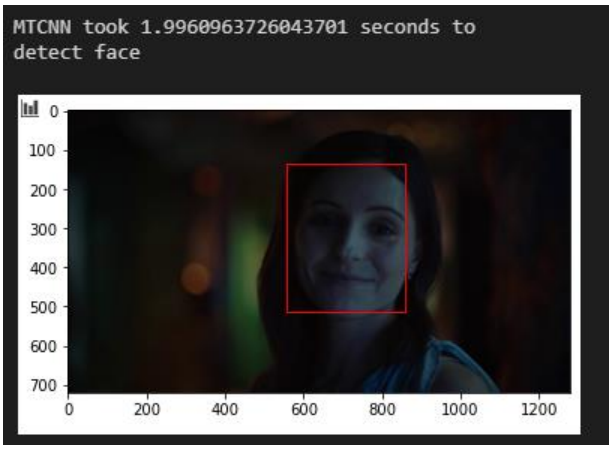

Fig 16:- Result for Low Light Face Detection

Fig 17:- Result for Multiple Face Detection

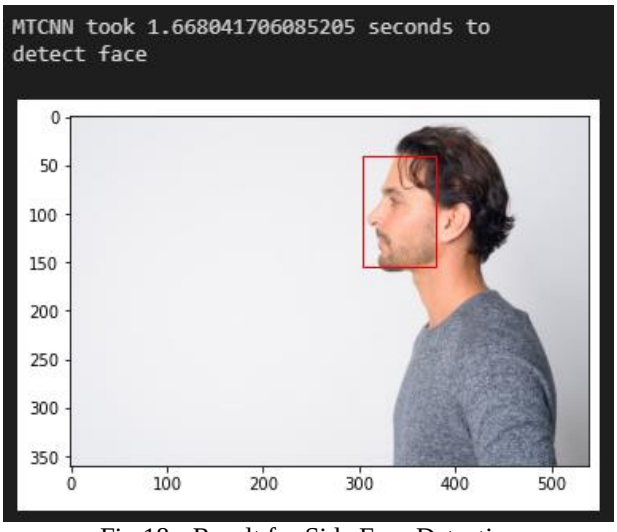

Fig 18:- Result for Side Face Detection

ISSN No:-2456-2165

IV. COMPARISON BETWEEN DIFFERENT FACE DETECTION ALGORITHM

Algorithm	Pros	Cons
Cascade Classifier	• Simple architecture • Detect faces at various scales • Real-time on CPU	• Lot of False Predictions • Don't work on side faces. • Don't work under occlusion.
Dlib CNN	• Easy to implement • Works with odd angles • Robust to different face occlusions. • Works on GPU	• Does not work well on real-time images • Works slow with CPU • Cannot detect faces below the minimum size
Dlib HOG	• Can work bit frontal face • Light weight model • Can work under different obstruction	• Really slow for real time detection • Does not work for side face • Does not work well under substantial obstruction.
MTCNN	• High accuracy. • Supports real time face detection. • It is efficient.	• It may take more time for training.

Table 1:- Comparison of different face detection Algorithms

V. CONCLUSION

We have done a survey of all the above algorithms and provided you with the most simple and easy way to understand. We have introduced each algorithm with its working and characteristics in an abstract way. For each algorithm we have mentioned pros and cons. They are easily understandable and one can have a quick idea about each algorithm according to their application of use.

REFERENCES

[1]. Paul Viola Michael Jones,Rapid Object Detection using a Boosted Cascade of Simple Features, ACCEPTED CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION 2001

[2]. FAREC - CNN Based Efficient Face Recognition Technique using Dlib, Sharma S, Karthikeyan Shanmugasundaram, Sathees Kumar Ramasamy, International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), 2016.

[3]. Analysis on Face Recognition based on five different viewpoint of face images using MTCNN and FaceNet By Al-imran 15201007 Baniamin Shams 15301030 Faysal Islam Nasim 15201051. 2019.

[4]. A Deep Learning Approach for Face Detection and Location on Highway by Yang Zhang 1,2, Peihua Lv1,2and Xiaobo Lu,2018.

[5]. Comparison of Haar-like, HOG and LBP approaches for face detection in video sequences ,Amal Adouani , Wiem Mimoun Ben Henia , Zied Lachiri ,16th International Multi-Conference on Systems, Signals & Devices (SSD'19),2019

[6]. A Convolutional Neural Network Cascade for Face Detection Haoxiang Li†, Zhe Lin‡, Xiaohui Shen‡, Jonathan Brandt‡, Gang Hua† †Stevens Institute of Technology Hoboken, NJ 07030 {hli18, ghua}@stevens.edu ‡Adobe Research San Jose, CA 95110 {zlin, xshen, jbrandt}@adobe.com

[7]. M. Turk; A. Pentland (1991). "Face recognition using eigenfaces" (PDF). Proc. IEEE Conference on Computer Vision and Pattern Recognition. pp. 586–591.

[8]. R. Lienhart and J. Maydt. An extended set of Haar-like features for rapid object detection. In Proc. of ICIP, 2002