Comparing CNN and RNN for Prediction of Judgement in Video Interview Based on Facial Gestures

Nishank Singhal
Department of Computer Science and Engineering
BITS Pilani, Dubai Campus
Email:nishanksinghal20nov@gmail.com

Neetika Singhal
Department of Information Technology
Indira Gandhi Delhi Technical University For Women, New Delhi
Email: neetikasinghal16@gmail.com

Srishti
Department of Computer Science and Engineering
BITS Pilani, Dubai Campus
Email:srishti258@gmail.com

Abstract - **This paper presents a novel technique of judging the performance of a candidate in a video interview. The candidate is judged as confident and attentive or unconfident and inattentive by taking the direction of face and eye into consideration. This corresponds to how many times is the candidate interacting actively, by making a firm eye contact with the interviewer. Image Processing techniques like Haar Cascade, Image filtering, Gamma Correction have been used for the detection of face and eye. Convolutional Neural Network (CNN) and Recurrent Neural Network (RNN) have been used for training and testing the images into right classes.**

Keywords – Image Processing, Haar Cascade, Gamma Correction, CNN, RNN

I. INTRODUCTION

The aim of the paper is to automate the quality of a video interview based on the facial direction, like head movement and eye contact of the candidates. Apart from what a candidate says in an interview, body movement, attentiveness, pupil of the eye and face direction are an integral part of how well humans communicate. Therefore, it is extremely important to understand the role of nonverbal communication in an interview. The nonverbal communication includes eye contact, facial movement, posture and hand gestures that helps in the interpretation of candidate's answer. Eyes are a way to establish a connection with the other person. You feel at ease while talking and communicating with the person when he or she is looking towards you. A big part of eye contact is to build trust as eye contact indicates trueness and confidence of the communication. This paper emphasizes the importance of face direction and eye contact in an online interview and how can it play a major role in selecting a successful candidate.

During recent years, facial expressions have gained the amount of attention. Darwin, in 1872, noticed that humans, use the same facial expressions for a variety of emotions [1]. Later, Ekman and others found out that micro-facial expressions, unconscious and involuntary miniscule movements of facial muscles denoting anger, disgust, fear, joy, sadness, shame, and surprise are same for the entire world [2][3], hence it is easily understood by others however the use of computer for automatically judge during an online interview will ease the process of selection of a candidate.

The paper aims at predicting the candidate's facial and eye movements as suitable or not suitable for the interview. The paper follows an approach of detecting the facial and eye direction with the help of Image Processing primarily, followed by training the Convolutional Neural Networks and Recurrent Neural Networks for multi-layered classification. With its forward-looking and refined ways, image processing has achieved remarkable success when it comes to object detection. In this paper, Image Processing acts as a pre-processing technique responsible for highlighting the region of interest, in this case, the face, eyes and pupil of the eyes of a particular candidate undergoing the interview process. This helps in providing the required features to the classifiers, CNN and RNN.

As for CNN, the last couple of years have seen a drastic change in the field of computer vision with its futuristic, fast and scalable end-to-end learning framework [4]. CNNs have always been known for their excellent application in image related tasks and are very useful where convolutional constraints are a potential premise. Like any other standard multi layered network, CNNs too comprise of one or more hidden layers along with subsampling steps and fully connected layers, but is able to put itself at a higher spot due to its ease of training with fewer parameters without compromising on the number of hidden layers involved [5].

Recurrent Neural Networks (RNNs) are popular models that have shown great application in many of the NLP tasks like Image Classification, Speech Recognition and Machine Translation. RNNs make use of sequential information, output being dependent on the previous computations. The ability to have a "memory" that stores the previous calculations makes RNN unique and different from other neural networks.

A. *Problem Statement*

The three major tasks of the problem involve: estimation of the face angle, the distance of the pupil from its center and the classification as correct/incorrect face/eye directions and the facial expressions classification later. Image processing techniques help in identifying the points of interest for finding out the face angle and pupil distance. This is followed by employing the multi-layered classification by using CNN and RNN for the purpose of training, as explained in section 3.

II. BACKGROUND

A. *Face and Eye Detection using Haar Cascade*

Paul Viola and Michael Jones have proposed an effective method of object detection using Haar feature-based cascade classifiers [1]. The proposed method uses machine learning by training a cascade function from a huge data set of positive and negative images, which is then used to detect objects in other images. This algorithm has been used for the purpose of face and eye detection.

The algorithm works as follows: Haar features, shown in Fig. 1, are extracted from the images (having faces and not having faces) that are used to train the classifier. Each Haar feature is computed by subtracting sum of pixels under white rectangle from sum of pixels under grey rectangle.

Out of all the features computed, only the best features are selected with the help of an algorithm known as Adaboost. Adaboost works by finding the best threshold for each feature that would decide the faces as being positive or negative. The process begins by assigning equal weights to each of the image. After every classification, weights of wrong classified images are increased until the required accuracy or rate of error is obtained or required correct result is obtained.

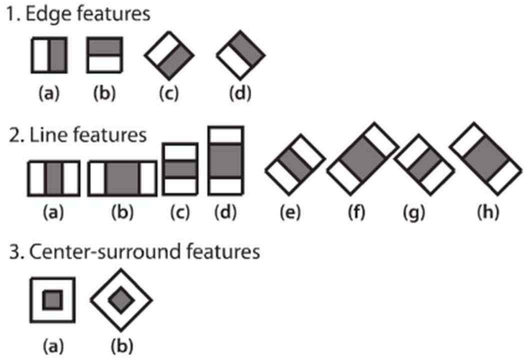

Fig. 1. Haar-like features.

Weighted sum of these weak classifiers is the final classifier for this algorithm. It is not possible for a weak classifier alone to do the classification independently, but when it is used as combination with other classifiers, it becomes a strong and produces excellent results. Concept of 'Cascade of Classifiers' is then applied to get the desired results. Different features are grouped into different stages of the classifiers and then applied one after the other, rather than applying all of the best features on a single window. If a window fails at first stage it is discarded, and the remaining features are not considered, since initial few stages contain very few number of features. And if it passes, it moves to the second stage of features and the process is continued similarly thereafter. The window that passes all the stages is concluded to be a face region. Fig. 2 shows the output of the Haar Cascade algorithm, depicting the detection of face and eyes.

B. *Stages of Image Processing*

After the application of the Haar Cascade algorithm, the highlight of the face and eyes is obtained. However, for the purpose of classifying the expressions into the desired classes, the center of pupil in the eyes and the center of the face have to be detected, following the algorithm that we have developed. Hence, for detecting the center of pupil and center of face, various methods of image processing are applied.

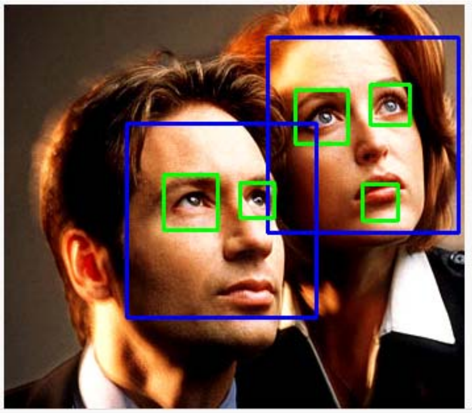

Fig. 2. Pictorial representation of Haar Cascade algorithm

Image Filtering

In order to filter out the pupil of the eyes, image filtering is used. Image filtering works by convolving a kernel with an image. Convolution is defined as an operation between an operator and every part of the image and a kernel is a fixed size array consisting of numerical coefficients along with a *pivot point* in that array, which is typically located at the center [6].

A 5x5 averaging filter kernel has been used, that can be defined as follows:

$$K = \frac{1}{25}\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \end{bmatrix} \qquad (1)$$

Average of pixel values inside a five by five window is calculated for a particular pixel. The window is centered on that pixel and the result obtained by summing up all pixels falling within this window is divided by 25. The output filtered image is the produced by performing this operation on all the pixels of the image. Expressing the procedure above in the form of an equation:

$$H(x, y) = \sum_{i=0}^{M_i-1} \sum_{j=0}^{M_j-1} I(x + i - a_i, y + j - a_j)K(i, j) \qquad (2)$$

Gamma Correction

In order to brighten the pupil of the eye after image filtering, gamma correction is applied. A non-linear operation which is used to encode and decode the luminance and tristimulus values in a video and/or still images known as Gamma correction or simply 'Gamma' [7], is used to correct the brightness of an image by using a nonlinear transformation function in between the input and the mapped output values:

$$I' = 255 \times \left(\frac{I}{255}\right)^{\gamma} \qquad (3)$$

The effect for all the pixels would not be the same as the relation is non-linear and would depend on the original value of the pixels. When $\gamma<1$, the original dark regions get brighter and the histogram shifts towards right whereas it is opposite for the case when $\gamma>1$.

Contouring

A contour can be defined as a sequence of points that outlines a shape or region having the same intensity or color. These Contours act as an important tool for the purpose of analysis of the shape and detection and recognition of the objects. Contouring has been used for the purpose of finding the center of face and the center of pupil, which are the key elements required in the algorithm.

C. *Convolutional Neural Network*

The abstract description of a CNN can be visualized as follows:

$$x^1 \longrightarrow w^1 \to x^2 \to \cdots \to w^{L-1} \to x^L \to w^L \to z \qquad (4)$$

As Eq (1) suggests, a 'CNN' moves layer after layer in the forward direction. Input value of 'x1', in the very first layer goes through the process, w1 (collectively called as a tensor). This layer maybe a convolutional layer, pooling layer, fully connected layer, etc. The process continues until the (L-1)th layer is reached which is usually a softmax layer (to make x^L a probability mass function) followed by the last layer which is a loss layer.

Finally, we achieve $x^{L} \in R^{C}$, i.e. a column vector with C elements and these estimates the later probability of x^1 representing C categories. Hence the CNN predictions can be used as an output as [8]

arg max i x L (5)

It is worth mentioning here that pooling and subsampling layers helps in reducing the noise in an image by decreasing its resolution. [9]

Loss layers are responsible for minimizing the cost by computing the loss of the forward pass and the gradient descent w.r.t. the loss of backward propagation. The softmax with loss layer just provides a numerically stable gradient.[10]

The object of the convolution operation is nothing but extracting features from the input. The convolutional procedure can be expressed mathematically as:

$$y_{i^{l+1},j^{l+1},d} = \sum_{i=0}^{H}\sum_{j=0}^{W}\sum_{d^l}^{D^l} f_{i,j,d^l,d} \times x^l{}_{i^{l+1}+i,j^{l+1}+j,d^l} \qquad (6)$$

Where 'x' is a tensor of third order in such a way that $x \in R^{H\times W\times D.}$ H, W, D are the elements each one denoted by an index-triplet' (i,j,d). $0 \le d \le D = D^{l+1}$, and for any spatial location (i^{l+1}, j^{l+1}) satisfying $0 \le i^{l+1} < H^l - H + 1 = H^{l+1}$, $0 \le j^{l+1} < W^l - W + 1 = W^{l+1}$. In this equation, " $(x^{l}{}_{i\ l+1+i,jl+1+j,dl})$" refers to the element of x^l indexed by these triplets "$(i^{l+1} + i, j^{l+1} + j, d^l)$" .

The charm of a CNN lies in the fact that all locations in the space has the same filter which leads to less parameters being involved and subsequently reducing the training time. The sharing of parameters promotes effective learning of good features in images. This means that out of all M features, it is possible that only one feature is useful in recognizing all Y object categories or a joint of all M features is required to recognize an object type of Y. Taking advantage of the same, we are implementing our classifications which are discussed in detail in the following section.

D. *Recurrent Neural Network (RNN)*

Recurrent Neural Networks are nothing but the Artificial Neural Networks that contains directed cycles in its computation graph. RNNs differ from the Feed Forward Neural Networks (FFNN) in that the current state of these RNN is directly dependent on its earlier states as the information travels in loop from one layer to another, whereas there is dependency of the current state to the previous states in the FFNN as the information moves purely in one direction from one layer to other. The dynamic temporal behavior of RNN is due to its memory that enables it to store information about its past computations. Architecture of a high level RNN is depicted herein under as Fig 3.

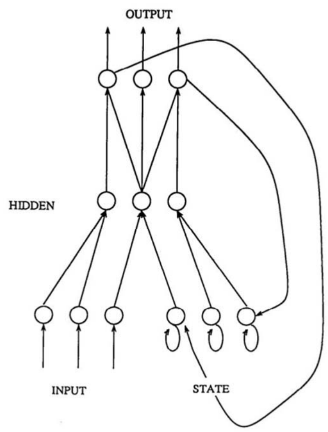

Fig. 3. RNN Architecture

III. Datasets, Implementation and Results

A. *Dataset for evaluation*

The images used in the project are divided into three categories, face-correct, eye-correct images; face-correct, eye-incorrect images and face incorrect images.

The training dataset consists a total of 10,000 images. This raining dataset is divided into nearly 2 equal classes, namely-images with the correct face direction and images with incorrect face direction --with respect to the camera.

The class of images with correct face direction is further categorized as images with correct eye contact and incorrect eye contact with respect to the camera. Both categories have approximately equal number of images.

The dataset of training images has been extracted from a video feed, where face images are of size 200x200 pixels whereas the eye images are of size 100x100 pixels.

Fig. 4 shows the sample of images having correct face and eye direction. Fig. 5 shows the sample of images having correct face but wrong eye direction. Fig. 6 shows the sample of images having wrong face direction.

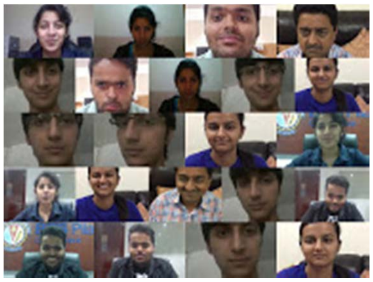

Fig. 4. Correct Face and Eye Direction

For the purpose of testing dataset, three video feeds have been used. The first video feed is near-to perfect video having the correct face as well as eye direction, spanning for about 32 seconds. The second video feed depicts incorrect face and eye direction, spanning for about 39 seconds. The third video feed is a combination of correct and incorrect face and eye directions, spanning for about 40 seconds.

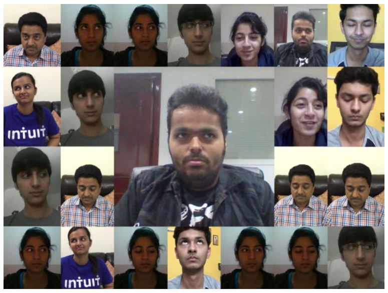

Fig. 5. Correct Face but wrong Eye Direction

Fig. 6. Incorrect Face Direction

B. *Implementation*

1) *Image Processing*

Fig. 7 summarizes the steps of implementation using image processing. First, using a live video feed, the face of the candidate is detected using a haar cascade. The center of this detected face is fixed and is used as a reference in the next stage. Hereupon, any major changes made by the candidate in the direction of their face with respect to the reference, is noted. If the changes are not much noticeable, these faces are further used to detect the eye of the candidate using another haar cascade. The center of this detected eye is fixed and is used as a separate reference at a later stage. To acquire the black spot inside the eye (called the pupil), a number of filters and gamma correctors are applied along with a black threshold. This is followed by finding the center of the identified pupil by drawing contours on it. This is done in order to calculate the distance between the previously calculated reference and the newly obtained center. This is done to check whether the candidate is looking in the correct direction or not, i.e. looking straight into the camera or not. Image processing on the whole is being used to categorize and distinguish between the correct and incorrect face and eye gestures. This acts as a preliminary step to the training method. Fig. 8 demonstrates the output at the various stages of Image Processing.

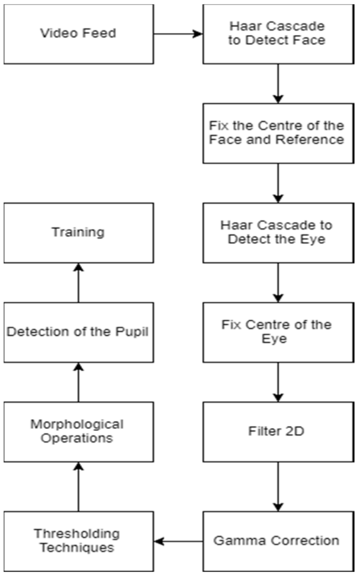

Fig. 7. Steps Image Processing

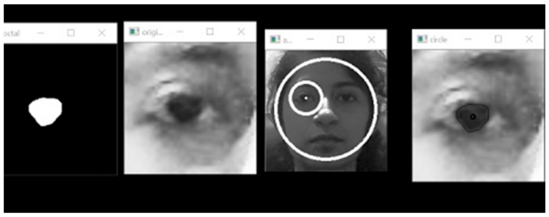

Fig. 8. Output at various stages of Image Processing

2) *Training Details*

With reference to Fig.9 and Fig.10, and image of size 120x120 is fed as input into the CNN and RNN model. CNN model consists of two hidden layers, two Batch Normalization and two fully connected layers with a ReLU and softmax as last layer. RNN model consists of two LTSM layers and a Batch normalization with a softmax as last fully connected layer. Each convolutional layer uses a filter of 5x5 and a stride of 2. Each convolutional layer is followed by a max-pooling layer of filter size 5x5 and a stride of 2. The CNN and RNN models are trained with 10,000 images against a validation set of 500. As per Fig.9, using a batch size of 120, validation is performed for 10.500 iterations. A constant learning rate of 0.001 is used throughout.

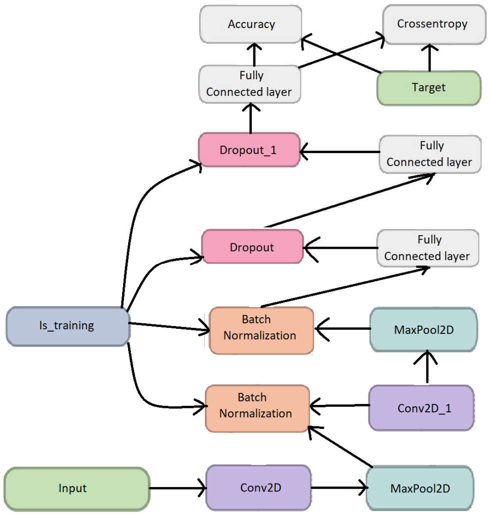

Fig. 9. CNN Model for video interview

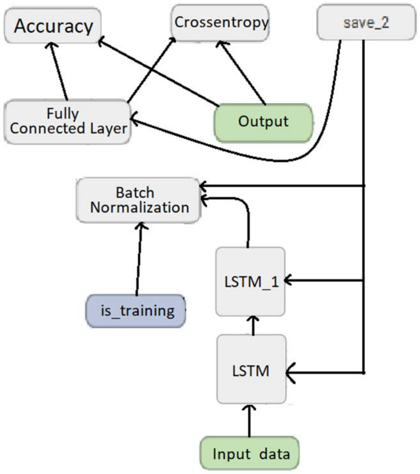

Fig. 10. RNN Model for video interview

3) *Testing Method*

The testing phase is designed to act as a multi-layered binary classification system which is shown in Fig.11.

C. *Results and Discussion*

CNN and RNN, both, are used as models for training and testing of the images. The accuracy of the models is compared in order to figure out the best model between the two. Fig. 12 shows the simulation results of accuracy vs the number of iterations, when RNN is used as a model for training and testing.

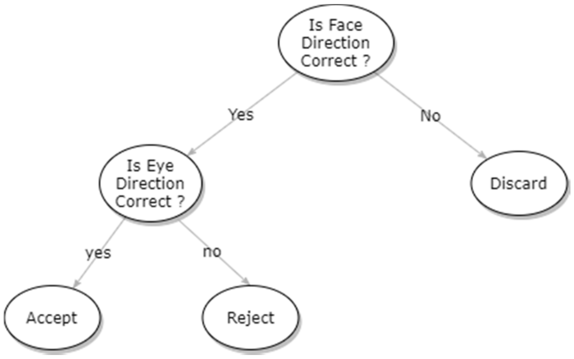

Fig. 11. Testing Phase

Maximum accuracy of 94.12% is obtained for the detection of face direction. This accuracy is obtained after iterations of approximately 10,800. It is also observed that, the accuracy is not so high for the detection of eye direction.

Fig. 13 shows the simulation results of accuracy vs the number of iterations, when CNN is used as a model for training and testing. Maximum accuracy of 91.16% is obtained for the detection of eye direction. This accuracy is obtained after iterations of approximately 5,700. It is also observed that, the accuracy is not so high for the detection of face direction.

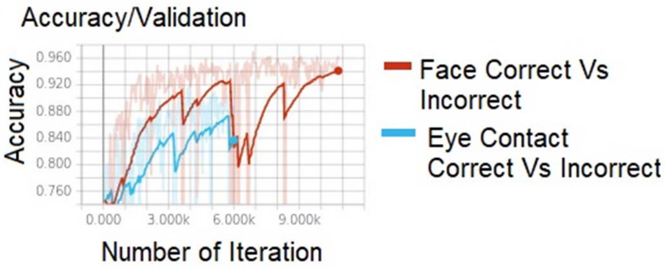

Fig. 12. Result for Training Accuracy Vs Number of iteration using RNN Model

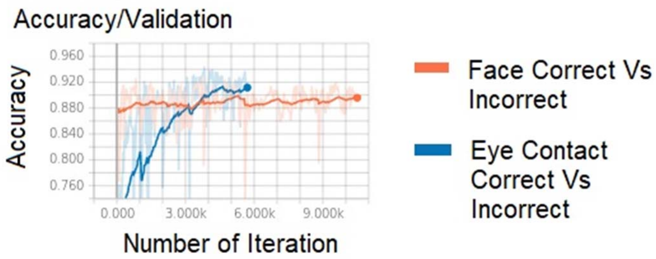

Fig. 13. Result for Training Accuracy Vs Number of iteration using CNN Model

From the above graphs, we find that the accuracy for detecting the face direction is maximum when RNN is applied whereas accuracy for detecting the eye direction is maximum when CNN is applied. Hence, we combine the RNN and CNN models in order to maximize the training accuracy.

Fig. 14 shows the simulation results of accuracy vs the number of iterations, when the combined model of CNN and RNN is used for training and testing. Maximum accuracy of 94.12 % and 91.16% is obtained for the detection of face and eye direction, respectively. Hence, the combined model is observed to show the maximum training accuracy.

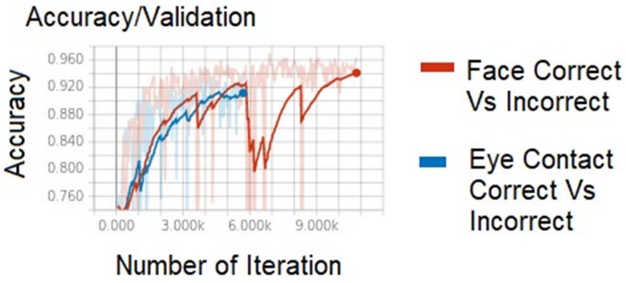

Fig. 14. Result for Training Accuracy Vs Number of iteration using CNN-RNN Model

In addition, testing accuracies are compared for a number of combinations of training models. These combinations include CNN-CNN model, CNN-RNN model, and RNN-RNN model. It is to be noted here that the individual models used in each combination are independent of each other. Fig. 15 shows the simulation results of Testing Accuracy Vs Training-models used. Maximum accuracy is observed in the case of CNN-RNN model is 92.2%, followed by the RNN-RNN model and CNN-CNN model with accuracies of 87.2% and 83.6% respectively.

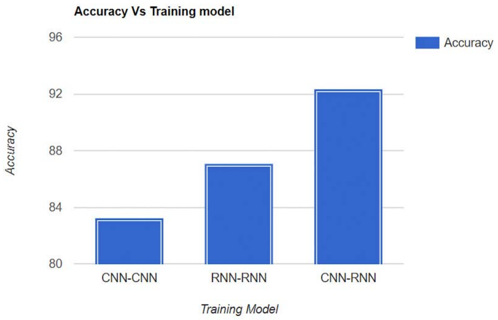

Fig. 15. Testing Accuracy

IV. Conclusions And Further works

The categorization of testing video feeds into right classes, between correct/incorrect face direction and correct/incorrect eye direction is hence successfully performed. The method above proposed produces an error rate of 7.8%, signifying 92.2% of success rate. Hence, it is experimentally proven that a combination of CNN-RNN model is the best method out of all the proposed methods for performing the right categorization of face and eye gestures with the maximum training and testing accuracy.

Future work involves increasing the testing dataset and ensuring the same efficiency across the large testing dataset. Also, it is aimed to include more features like hand gestures and voice tones in the training dataset, thus making the judgement of candidate more real time. Additionally, it is required to increase the number of personality traits such as openness, emotional stability on which the individual in an interview would be evaluated.

V. Acknowledgement

The authors would like to acknowledge the authorities of BITS Pilani, Dubai Campus for their constant support and motivation. The authors would also like to thank Mr. Atul Jain, Deputy General Manager – Human Resources, HCL America Inc for his valuable time, suggestions, support and helping in review, authentication of the dataset. The author would also like to thank Mr. Nalin Singhal, Director CEAF Bangalore, MES, MOD INDIA for his valuable time, support and helping in obtaining the dataset.

References

[1] Darwin, C., The expression of the emotions in man and animals.1872, London,: J. Murray. vi, 374 p.

[2] Ekman, P., A methodological discussion of nonverbal behavior. Journal of Psychology, 1957(43): p. 9.

[3] Ekman, P., Body Position, Facial Expression, and Verbal Behavior during Interviews. J AbnormPsychol, 1964. 68: p. 295-301.E. Rublee, V. Rabaud, K. Konolige, and G. Bradski, "ORB: An Efficient Alternative to SIFT or SURF," in IEEE International Conference on
Computer Vision, November 2011, pp. 2564–2571.

[4] Max Jaderberg, Karen Simonyan, Andrew Zisserman and KorayKavukcuoglu, "Spatial Transformer Networks," 28th InternationalConference on Neural Information Processing Systemss (NIPS'15), pp.2017-2025, 2015.

[5] Andrej Karpathy, "Convolutional Neural Networks for Visual Recognition,"in Lecture Notes in Stanford CS class CS213n 2016.K. Alsabti, S. Ranka, and V. Singh, An Efficient k-means Clustering Algorithm, Proc. First Workshop High Performance Data Mining, Mar. 1998.

[6] https://docs.opencv.org/2.4/doc/tutorials/imgproc/imgtrans/filter_2d/filter_2d.html

[7] Charles A. Poynton (2003). Digital Video and HDTV: Algorithms and Interfaces. Morgan Kaufmann. pp. 260, 630. ISBN 1-55860-792-7.

[8] Jianxin Wu, Introduction to Convolutional Neural Networks, LAMDAGroup National Key Lab for Novel Software Technology NanjingUniversity, China, 2017.

[9] Samer Hijazi, Rishi Kumar, and Chris Rowen, Using ConvolutionalNeural Networks for Image Recognition, IP Group, Cadence, 2015.

[10] YangqingJia and Evan Shelhamer, Caffe - Deep Learning Framework Notes, Berkeley AI Research(BAIR).