

# Real Face Detection and Recognition: The Live Experiment

Shailesh Wadhankar  
Delivery Partner  
Persistent Systems Ltd  
Pune, India

Priya Singh  
Senior Technical Specialist  
Persistent Systems Ltd  
Pune, India

Soumyakant Sahoo  
Software Engineer  
Persistent Systems Ltd  
Pune, India

## ABSTRACT

Face Recognition is one of the most widely researched and challenging fields in computer vision and machine learning. Dynamism of human face poses several challenges in developing a machine learning algorithm or a deep learning model for face recognition across different environments. This mix of human face dynamism and altering environmental factors leads to inaccurate face recognition.

The goal of this paper is to propose a face recognition model, below are the multiple actions taken to finalize the model.

- ✓ **Framework Selection:** Open Face framework and linear SVM classifier to recognize a person's face after comparing with other models or frameworks available with the help of live experimentation on human faces
- ✓ **Live Face Recognition activity:** Two rounds of Crowd testing has been conducted at Persistent Systems Pune & Nagpur offices.
  - **Crowd Test 1 (CT1):** 223(86 and 137 in two batches) candidates, 40 images each. Systems were trained daily with new images collected in the process.
  - **Crowd Test 2 (CT2):** 81 candidates, 80 images each. System has been trained only for last day of testing. Total daily score was higher than CT1, as the system was trained with double the number of images.

## General Terms

Face Recognition, Computer Vision, Machine Learning Algorithm, Deep Learning

## Keywords

Open Face framework, linear SVM classifier, face recognition model, OpenCV, detect eye-blinks, eye aspect ratio, Neural Networks, CNN

## 1. INTRODUCTION

Face recognition is a biometric technique to recognize a given face. There are other biometric ways to identify individuals for their claimed identity through fingerprint scanning, retina scan, DNA analysis etc. So, what makes face recognition different from other biometric techniques?

The surreptitious answer is face recognition does not require an individual's participation for being recognized. This holds the key for considering face recognition as the primary biometric technique for authentication and recognition. Use cases for face recognition are spread across domains from national security, cyber security to smart home security systems. In past 10 to 15 years, the data generated has grown manifolds and so has the privacy and security concerns of an individual's data. Face recognition has been in consideration for long time to address these concerns.

You can divide Face recognition into two steps:

- 1) Face authentication
- 2) Face recognition

Face authentication is the process of matching the queried face with a given dataset of images to authenticate the claimed identity. Face recognition is the process of identifying an individual by comparing live capture or digital image data with the stored record for that person. In this paper, the authors have discussed a face recognition model based on OpenFace framework and a linear SVM classifier for webcam based facial recognition. They have also discussed the results of two rounds of live face recognition activity conducted with two different testing approaches.

## 2. LITERATURE SURVEY

Xiaoyang Tan, Songcan Chen, Zhi-Hua Zhou, Fuyan Zhang (2006) [3] categorize and critically evaluate several prominent recognition algorithms. They describe that the challenges faced by the current face recognition techniques lies in the difficulties in collecting samples. Fewer samples per person mean less laborious collection effort, lower cost for storing and processing them. This situation is called "one sample per person" problem: given a stored database of faces, the goal is to identify a person from one image stored in the database irrespective of varying and unpredictable poses, angle, lighting, etc. Our experience with data capturing phase of Persistent Employees, resonates with the live sample collection issues discussed by the authors and prove how algorithms accuracy changes with number of sample faces in the dataset.

Dominic Asamoah, Peter Amoako-Yirenyki, Stephen Opoku Oppong and Nuku Atta Kordzo Abiew(2017) [7] employ a method, where an individual's face is captured on a camera by receiving video sequence, streamed into frames and transformed into RGB. Haar classifiers are used to detect the eyes region and eyelid feature. Eyes are detected to be either open or closed at a given moment by using threshold and equations on the symmetry of human face. The eye region is processed to ascertain certain attributes of eyelid movement. It is suggested that, Kalman filter when used with the blink cycle would be a strong thresholding application, useful in tracking and predicting the blinking rate of the eyelid. Authors, initially used Haar classifier and later changed to EAR method as it provided better results.

Ranjana Sikarwar and Pradeep Yadav (2017) [4] present a hybrid approach for face detection and feature extraction. They present a combination of three well-known algorithms; Viola- Jones face detection framework, Neural Networks and Canny edge detection method to detect face in static images. The proposed work emphasizes on the face detection and identification using Viola-Jones algorithm - a real-time face

detection system. Neural Networks are used as classifiers between faces and non-faces while Canny edge detection method is used for detecting face boundaries. The Canny edge detector is primarily useful to locate sharp intensity changes and to find object boundaries in an image. The authors use hybrid approach for face detection and feature extraction with a set of algorithms and convolutional neural networks for classification.

### 3. PROPOSED METHODOLOGY

Authors propose a two-phase process for facial recognition. **First phase** is the detection phase, which comprises the following two steps:

1. Face detection as a webcam input and capture frames using OpenCV.
2. Detect eye-blinks using eye aspect ratio method on captured frames to identify if it is a live person's face or not.

In the first step, authors propose detecting faces within a pre-determined middle boundary on screen having 200\*100 pixels dimension. In case of multiple faces within the determined boundary, the face closer to camera is focused on. If a focused face is in determined boundary, up to 20 frames are captured. Authors reasoned their decision to capture only 20 frames in contrast to recommended 60 frames per face is to reduce the overall face detection time.



**Fig 1: Application Screen with 2 faces, 1<sup>st</sup> in and 2<sup>nd</sup> out of the boundary**

After the completion of first step, frames that satisfy the following two conditions are saved for the second step:

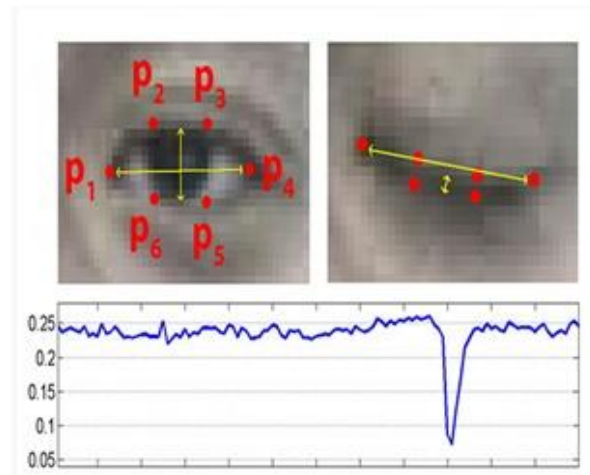
1. Face Coverage  $\geq 40\%$
2. Blurriness  $\geq 100$  Cycle per pixel(C/P)

Frames that satisfy the above-mentioned condition proceed to second step for detecting of eye blinks using the eye aspect ratio method. Authors propose facial landmark detection method to localize the important regions of the face, comprising eyes, eyebrows, nose, ears, and mouth.



**Fig 2: Eye blinks not detected in the boundary are discarded**

Stressing on detecting eye blinks is to ensure the frames of face being captured is a live human face instead of an image. EAR (Eye Aspect Ratio) is determined using following formula:



**Fig 3: Eye blink detection with EAR method**

$$EAR = (\|P_2 - P_6\| + \|P_3 - P_5\|) / 2\|P_1 - P_4\|$$

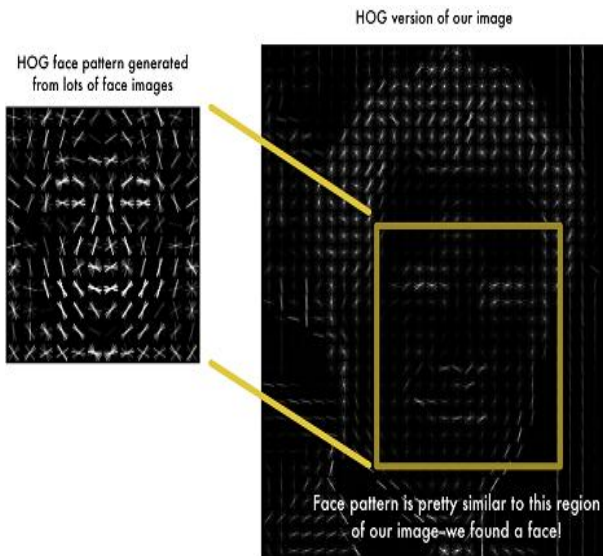
Author's results show that the eye blinks are consistently well captured in high-resolution close-up image using EAR than OpenCV Haar-cascade method. The EAR value with eyes remains constant and rapidly falls to zero when the eyes close between the blinks. This makes it an effective method for detecting eye blinks ensuring the face being captured is a live human face. The captured frames from the first phase is observed for both eye blinks before being sent for the second phase of facial recognition.

For the second phase, authors propose using OpenFace along with dlib. After comparing OpenFaceframework with TensorFlow CNN, Inception-v3 model and IBM-VR, they found OpenFace delivered better results in their test set-up. The test details and set-up criteria are mentioned in Experimentation section.

**The second phase**, Face recognition for recognizing the face captured from phase one, comprises of the below steps:

1. Finding face in the frame captured by the camera
2. Handling faces looking in different directions
3. Generating 128-embeddings for faces
4. Recognition of the person

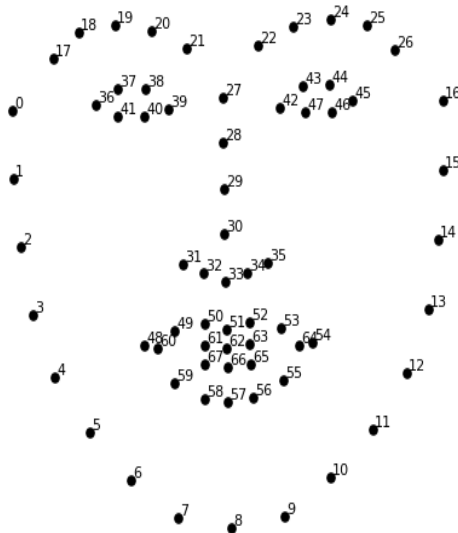
In step 1, to find a face in the captured frame, the image is converted into a grayscale image. The grayscale image is further processed to find the basic pattern of flow of light i.e. the direction in which the image is getting darker or lighter, represented by an arrow. The image is broken into squares of dimension 16\*16 and then replacing each square with a single arrow that holds the majority within the square. Replacing all the squares within the image produces a basic facial structure as shown in the following image:



**Fig 4: Frontal Face Detection using HOG Face Pattern**

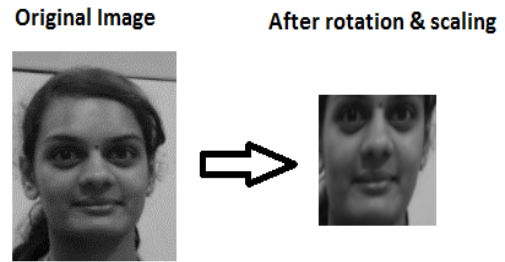
This process is done by dlib's frontal face detection using HOG face pattern.

In step 2, the authors propose to use the face landmark estimation algorithm for handling faces looking in different directions. The core of this algorithm is to detect 68 specific points called landmarks that exist on every face. These landmarks comprise of points such as the top of chin, outside edge of each eye, inner edge of each eyebrow etc. Once these 68 landmarks are detected for a face, a trained machine-learning algorithm can detect these 68 specific landmarks on any face. This process is accomplished by dlib's 68\_face\_landmarks.



**Fig 5: 68 Facial Landmarks Estimation**

Providing information about the step 3, the authors propose affine transformation of image with minimum distortion.



**Fig 6: Transformed image with minimum distortion**

Following the affine transformation of image, is the extraction of few basic measurements for each face such as size of each ear, the spacing between the eyes, the length of the nose, etc. After extracting these basic measurements for each face, CNN (Convolutional Neural Network) is used to find the facial features best suited for recognizing the face. CNN, when trained, generates close to 128 measurements for different images of a single person. However, these 128 measurements are different for different people. This process requires large number of images and computing prowess. For simplifying this task, authors recommend using OpenFace framework, which has a ready trained model to generate 128 measurements for any dataset of human face images.



**Fig 7: 128 measurements generated by OpenFace framework**

Step 4 (last step) involves verifying which person from the training dataset is the closest match with the test image. With the help of 128 measurements generated from step 3 for every face in training dataset, any of the machine learning classification algorithms can be used. A comparison of 5 common classifiers was performed with the available dataset of captured images. The authors propose using linear SVM classifier citing better prediction accuracy among all classifiers:

**Table 1. Comparison results of common classifier**

Classifier	Training Data*	Test Data*	Correct Prediction	Incorrect Prediction

LSVM	3240	1400	35	0
DBN	3240	1400	0	35
Decision Tree	3240	1400	7	28
Guassian	3240	1400	12	26
RSVM	3240	1400	33	2

\*Training Data  $\rightarrow (81 \times 40 = 3240)$  81 employees with 40 images each, Test Data  $\rightarrow (35 \times 40 = 1400)$  35 employees with 40 images each

#### 4. EXPERIMENT

Authors conducted experiments to find the most accurate face recognition model for a given use case. They propose using OpenFace algorithm with dlib in contrast to earlier approaches of using TensorFlow Inception-v3 model and IBM-VR. Experiments were done with the help of 14 employees with 150 images each, covering 24 different variations such as straight face, tilted face, varied facial expressions, holding phone, with headsets, different hair styles etc. in the data set. OpenFace with dlib gave the most accurate results with good confidence distribution scores for top 5 predictions.

Experiments were also performed to find the accuracy of the selected (OpenFace with dlib) model. Accuracy of the face recognition model proposed by the authors was computed by live face recognition activity conducted with CT1 – 223 employees, 40 images each & CT2 – 81 employees, 80 images each; of Persistent Systems Ltd at Pune and Nagpur locations respectively. This activity was conducted in following phases:

1. Data collection and training of the model
2. Testing the trained model with the images captured.

Data collection was conducted over a period of two weeks at both the locations. For each individual, 40 images for CT1 and 80 images for CT2, were captured and fed to the model for training. For individuals with & without spectacles were captured. All the images for an individual were frontal or straight face and slightly tilted images. Total number of images used to train models for CT1 - 8920 and CT2 - 6480.

For CT1, the model was retrained daily, with captured images for that day of the individuals who were incorrectly recognized. For CT2, the model was retrained only for the last day of testing, as this round already had 80 images per person in the dataset.

After data collection, the images were processed as per the steps mentioned in the paper and a python script was employed to train the model with the processed images.

The second phase of the activity was spread over a period of three weeks. During this period, every individual who was part of data collection in phase one was requested to come and test the model by standing in front of the webcam identifying the individual with their name. Depending on whether the name returned by the model was correct or not, the individual was asked to provide a feedback in 'Yes' or 'No' through a popup dialog with a question "Were you identified correctly?" The python script also logs the user response in a log file. The model generates a response with top five predicted names and their respective confidence scores, with the highest confidence score on top. The data was analyzed and the model was re-trained for images with low confidence and negative recognition for CT1 and only

on the last day for CT2. Reports generated were also analyzed to derive observations from the experiment and state the accuracy of the model.

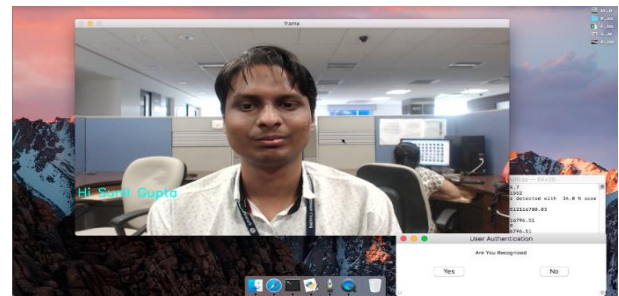


Fig 8: An individual with correct prediction. Pop-up dialog requesting user input for crowd testing

```

01-12-2017log.txt
=== ./images/Detected/1512116165.94.jpg ===|
Predict Sunil_Gupta with 0.45 confidence.
Predict Sangamesh_Somawar with 0.37 confidence.
Predict Hrushikesh_Vaze with 0.05 confidence.
Predict Ayush_Pathak with 0.04 confidence.
Predict Mohit_Gupta with 0.02 confidence.
Time taken for detection:::11.6977419853

=== ./images/Detected/1512116763.49.jpg ===
Predict Sangamesh_Somawar with 0.34 confidence.
Predict Sunil_Gupta with 0.33 confidence.
Predict Hrushikesh_Vaze with 0.14 confidence.
Predict Ayush_Pathak with 0.10 confidence.
Predict Mohit_Gupta with 0.03 confidence.
Time taken for detection:::25.6872839928
    
```

Fig 9: Logs for top 5 predictions with confidence score and other details

#### 5. OBSERVATIONS

##### Crowd Testing 1

1. Retraining the model everyday showed the difference in average increase of model confidence, increasing from 10-20% to 40-50%.

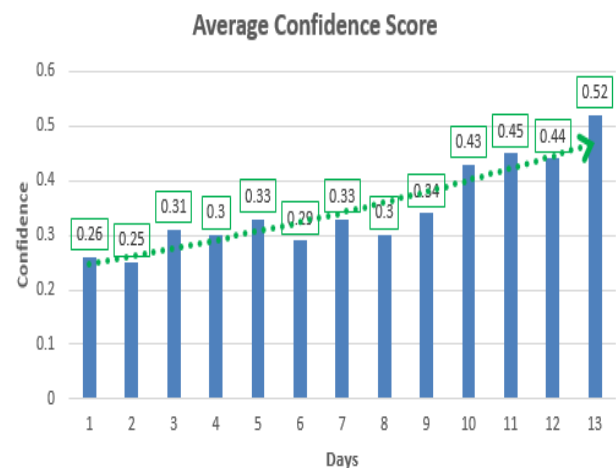
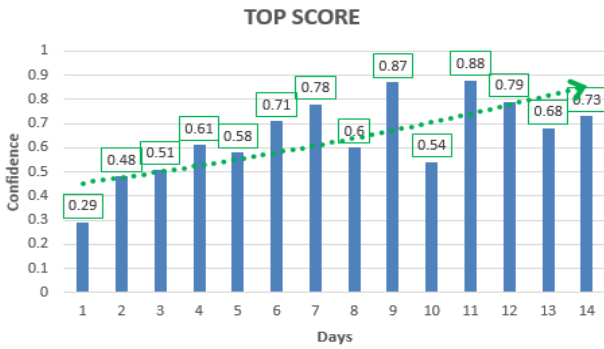


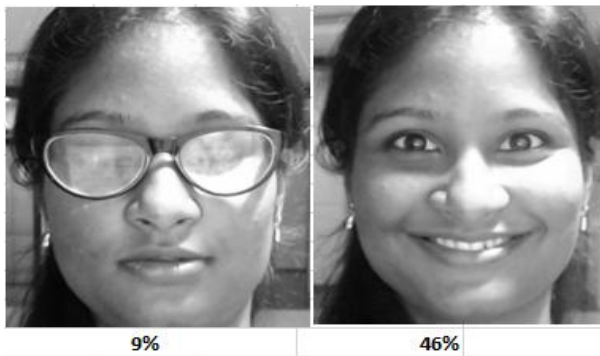
Fig 10: Graph showing average confidence score per day for persons in dataset (CT1)

2. Highest confidence score for an individual increased with retraining the model with more images.



**Fig 11: Graph showing top score per day for persons in Dataset (CT1)**

- Variations in facial images were fed to the model for retraining, resulting in increased prediction confidence.
- Over exposure or extremely less exposure to light were observed as the major hurdles affecting the prediction confidence of the model.
- It was also observed that if the facial features are not visible distinctly for reasons such as over or little exposure to light, the model produced negative recognition.



**Fig 12: Light variation effect on recognition**

- For individuals with spectacles, the model gave better predictions when dataset consists of almost equal number of images with and without spectacles.
- For individuals with beard, the model gave better predictions when dataset consisted almost equal number of images with and without beard.

Both the models were tested with 10-15 employees outside the dataset daily to determine the threshold. Access or privilege related applications based on face authentication deny access if the confidence score is low. Low score means a wrong match and hence threshold is important for such applications. Average confidence score for individuals not in dataset were used to determine the threshold as 15.19%.

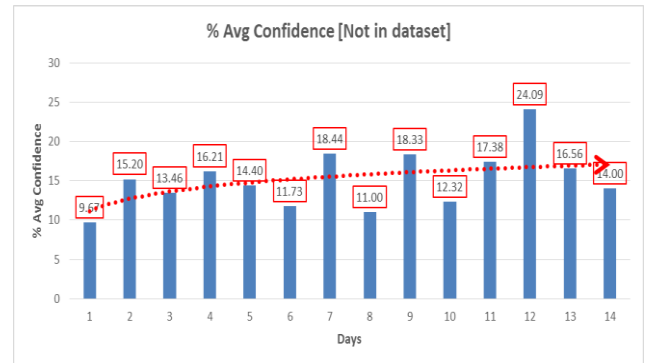
**Table 2. List of few Accuracy & Threshold combinations**

Location	Accuracy of the model	Threshold
Pune	93.32% *	None
Nagpur	92.47% *	None
Pune	78.79% **	15.19%

Pune	78.415% **	10%
------	------------	-----

\* (total positive cases \*100)/total tested cases

\*\* (for total positive and negative cases)



**Fig 13: Graph showing average confidence score for individuals not in dataset (CT1)**

**Crowd Testing 2**

- Model has been trained with 80 images per candidate, resulted in higher Confidence Level than CT1.



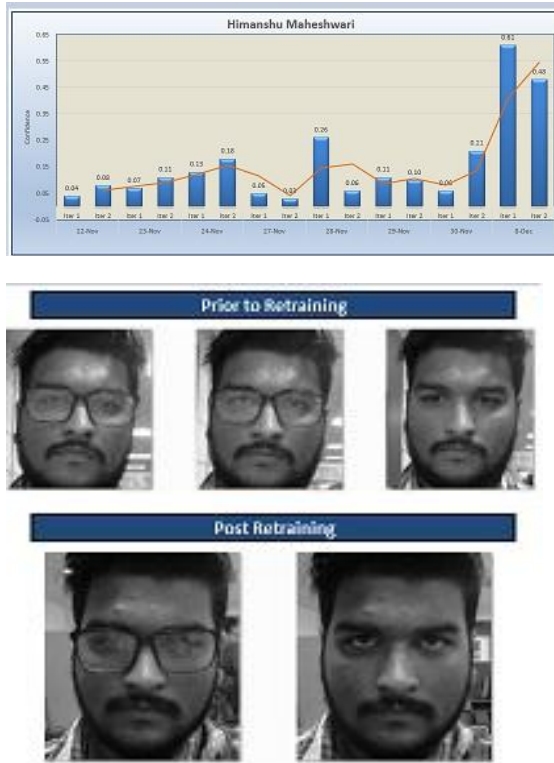
**Fig 14: Graph showing top confidence score per day for persons in dataset (CT2)**

- Average Daily score has been observed as lower than CT1, due to less number of participants and more variations in capturing the image – With/Without specs, Facial expressions, Closed Eyes, Not enough light.

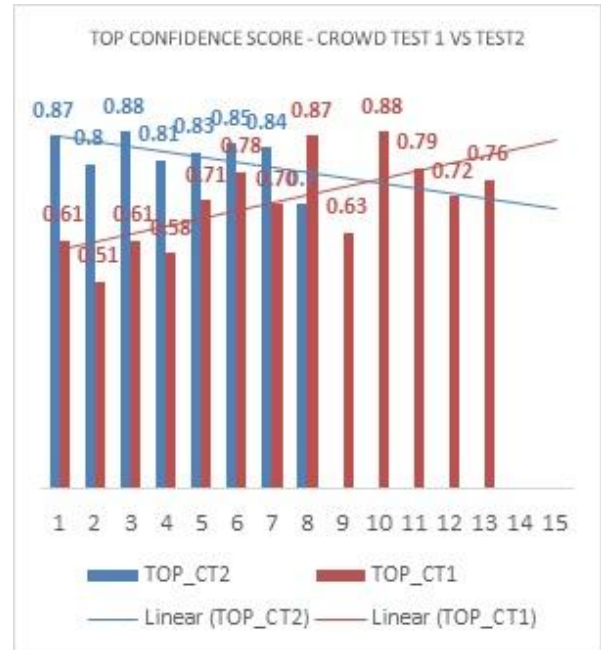


**Fig 15: Graph showing average confidence score per day for persons in dataset (CT2)**

- Retraining the model for the last day of crowd test, has not shown major difference for prior & post test results.
- Retraining the model for with/without specs has shown major differences, where model was not able to recognize or recognizing with low confidence score prior to retraining. But post retraining has shown significant difference.



**Fig 16: Graph showing top confidence scores comparison (CT1 vs CT2)**



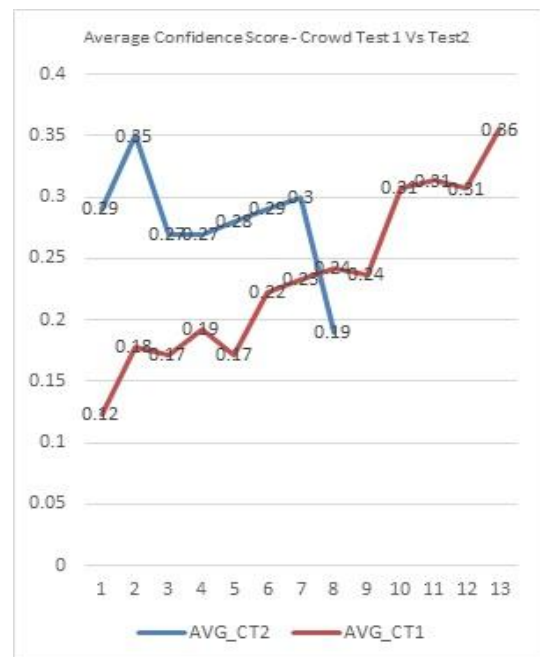
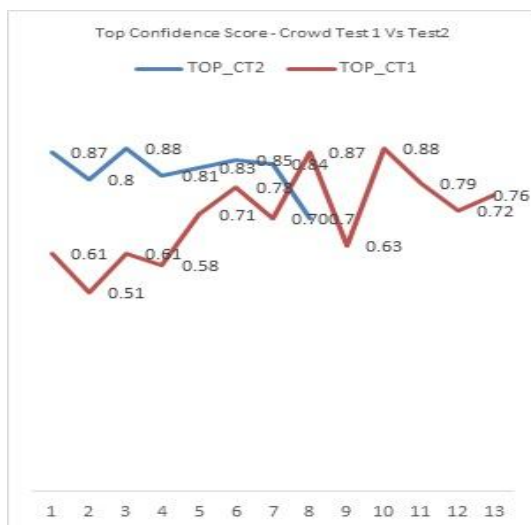
**Fig 17: Graph showing top confidence scores comparison (CT1 vs CT2) with trend lines**

Average daily scores took a dip in last day of CT 2 (possibly because of number of employees who showed up for crowd testing 2). But overall average confidence scores were found to be higher in CT 2. Charts below show improvements made in crowd testing 2. Linear trend lines showing possible downward trends for crowd testing 2. Possible reasons being - CT2 had a shorter test duration and fewer people showing up for testing than CT1. Analyzing the data, CT2 will possibly show upward trends, provided there exactly is same setup like in CT1 (Number of days and number of people in dataset)

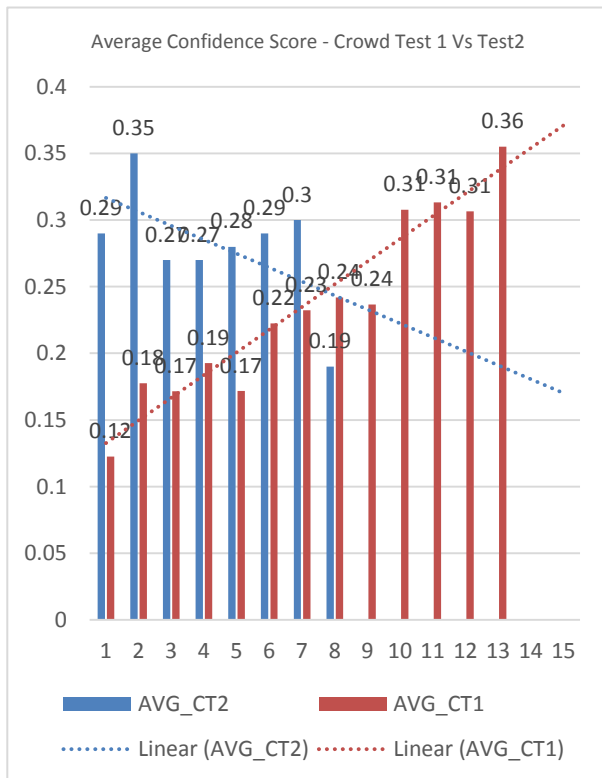
The models were also tested with 10 employees outside the dataset daily to determine the threshold. Average confidence score for individuals not in dataset were used to determine the threshold as 17%.

## 6. CROWD TESTING (1&2) COMPARISON AND INSIGHTS

Top daily scores were found to be higher in CT 2.



**Fig 18: Graph showing average confidence scores comparison (CT1 vs CT2)**



**Fig 19: Graph showing Average confidence scores comparison (CT1 vs CT2) with trend lines**

- Retraining has been done only for last day of testing and results did not show any major difference after retraining.
- Cases (2) with spectacles showed major difference. Model was unable to recognize the candidates prior retraining (75-80 % cases). Post retraining, confidence scores were higher.
- There is a difference in confidence scores viz testing them in lab (ideal) conditions versus testing in live conditions (low light, camera angles and different facial expressions)
- In cases where the model is unable to recognize a candidate, retraining them also did not deliver expected results.

## 7. FUTURE PROSPECT

Giving an insight into the potential and use case of face recognition, authors cite the extensive use of face recognition in field of surveillance and security that is currently dominated by non-intelligent CCTVs. With the power of face recognition infused into the surveillance and security domain, the domain will be further strengthened to curb crimes and terrorism. Another interesting use case proposed by the authors is to track individuals who are reported to be not found. For institutions that cannot afford a biometric authentication, facial recognition is an alternative - provided facial recognition is developed into an accurate form of authentication.

This study is a small step towards the future of computer vision and encourages further study and research to overcome the observed hurdles.

## 8. ACKNOWLEDGMENTS

Authors are thankful to the entire team who contributed in the research and development of the application and helped in gathering images for the dataset, training models, and analyzing the result sheet. Special mention -the Nagpur and Pune teams who gave their consent to be part of this crowd testing activity and regularly came for 3 weeks to test the system. Special mention - employees whose images (with consent) are used in the paper to depict system details and some of the observations.

## 9. REFERENCES

- Mandeep Kaur and Jasjit Kaur (2017) "Review of Face Recognition Techniques." In: International Journal of Computer Applications (0975 – 8887) Volume 164 – No 6
- Zhang W. and Guo Y. (2000) "Feature-Based Face Recognition: Neural Network Using Recognition-by-Recall" In: Mizoguchi R., Slaney J. (eds) PRICAI 2000 Topics in Artificial Intelligence. PRICAI 2000. Lecture Notes in Computer Science, vol 1886. Springer, Berlin, Heidelberg
- Xiaoyang Tan, Songcan Chen, Zhi-Hua Zhou, Fuyan Zhang (2006) "Face recognition from a single image per person: A survey" In: Journal Pattern Recognition Archive Volume 39 Issue 9, September, 2006 Pages 1725-1745
- Ranjana Sikarwar and Pradeep Yadav (2017) "An Approach to Face Detection and Feature Extraction using Canny Method" In: International Journal of Computer Applications (0975 – 8887) Volume 163 – No 4, April 2017
- Arai, K. and Mardiyanto R (2011) "Comparative Study on Blink Detection and Gaze Estimation Methods for HCI, in Particular, Gabor Filter Utilized Blink Detection Method" In: Proceedings of the 2011 Eighth International Conference on Information Technology: New Generations. ITNG '11, Washington, DC, USA, IEEE Computer Society, 2011, pp. 441-446.
- Akshata. S. (2016) "Eye Blink Detection Using Adaboost Approach and Morphological Operation" In: International journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering. Vol 5, Issue 4, April 2016.
- Dominic Asamoah, Peter Amoako-Yirenkyi, Stephen Opoku Oppong and Nuku Atta Kordzo Abiew (2017) "Establishing the Blink Cycle of the Eye using OTSU Method and Gaussian Filter" In: International Journal of Computer Applications (0975 – 8887) Volume 175 – No.4, October 2017
- X. Wang and X. Tang (2009) "Face Photo-Sketch Synthesis and Recognition" In: IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI), Vol. 31, 2009.
- Jian Yang, David Zhang, Alejandro F. Frangi, and Jingyu Yang (2004) "Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition" In: Journal IEEE Transactions on

Pattern Analysis and Machine Intelligence Volume 26  
Issue 1, January 2004 Page 131-137

- [10] Taigman, Y. (2014) “Deepface: Closing the gap to human-level performance in face verification.” In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2014
- [11] Benuwa, B.B (2016) “A Review of Deep Machine Learning.” In International Journal of Engineering Research in Africa. 2016. Trans Tech Publ.

[12] Robert Yao Aaronson, Wu Chen and Ben-Bright Benuwa. (2017) “Robust Face Detection using Convolutional Neural Network.” In: International Journal of Computer Applications 170(6):14-20, July 2017.

[13] Vijayalakshmi A. (2017) “Recognizing Faces with Partial Occlusion using in painting.” International Journal of Computer Applications 168(13):20-24, June 2017