IOSR Journal of Computer Engineering (IOSR-JCE)
e-ISSN: 2278-0661,p-ISSN: 2278-8727, Volume 22, Issue 4, Ser. I (Jul.–Aug. 2020), PP 01-07
www.iosrjournals.org

Multi Target Facial Recognition System Using Deep Learning

Aishwarya Ramesh[1], AnirbanGhosh[2]

[1]*(Information Science and Engineering, BMS College of Engineering, India)*
[2]*(Information Science and Engineering, BMS College of Engineering, India)*

Abstract:
Object/pattern recognition is a technique of computer vision for image or video classification of objects. Facial recognition is one of the most popular streams of object recognition. Facial recognition is an automated classification that mathematically finds a person's attributes regarding the face using deep learning. This paper elucidates the phases of facial detection, extraction of facial features, and finally identification of the person based on said features. As concepts of facial recognition become more and more relevant in the modern industry, researchers have created and refined many recognition models. A few of the drawbacks of these facial recognition solutions are their heavy training times and limited reusability. This paper addresses these problems and attempts to solve them by creating an ensemble of best in the market technologies to achieve high accuracy at high speeds. The system is designed to allow fast customization for any niche dataset, for example, students attending a class, a group of wanted individuals, patients in a hospital, etc.

Background*: Computer vision and Facial Recognition are two among the most up and coming technologies in the IT market. This paper utilizes the power of two highly sought after algorithms, YOLOv3 (known as one the fastest object detection algorithms), and Google FaceNet (known as one of the most accurate facial feature extraction algorithm), to produce a highly customizable facial recognition system.*

Results*: The system generated in conjunction with the concepts presented in this paper is capable of providing customized facial recognition upon an image dataset populated with 1930 facial images of 7 persons under a minute on a computer system of average performance without the help of a GPU.*

Conclusion: *The results are indicative of a highly customizable facial recognition system.*

Key Word*: Facial Recognition; Google FaceNet; YOLOv3; Computer Vision; Image Processing; Deep Learning.*

Date of Submission: 20-06-2020 Date of Acceptance: 10-07-2020

I. Introduction

This paper brings together 3 different machine learning models (YOLOv3, Google FaceNet, and a custom neural network) and interfaces between them so they work together to locate a face in the image, extract its features and then use those features to identify the person in the image. The proposed system uses the 3 models to locate, represent, and identify faces in an image frame. The system extracts raw pixel data from the input video/ image/ camera source and then feeds them sequentially to the 3 models. The output of each model is used as input for next.

The system uses the YOLOv3 model to locate the faces in the image. A Google FaceNet model is used to create a 128-byte representation of each facial image. Finally, a custom neural network uses the 128-byte representations of the facial images to identify the person in the image.

DOI: 10.9790/0661-2204010107

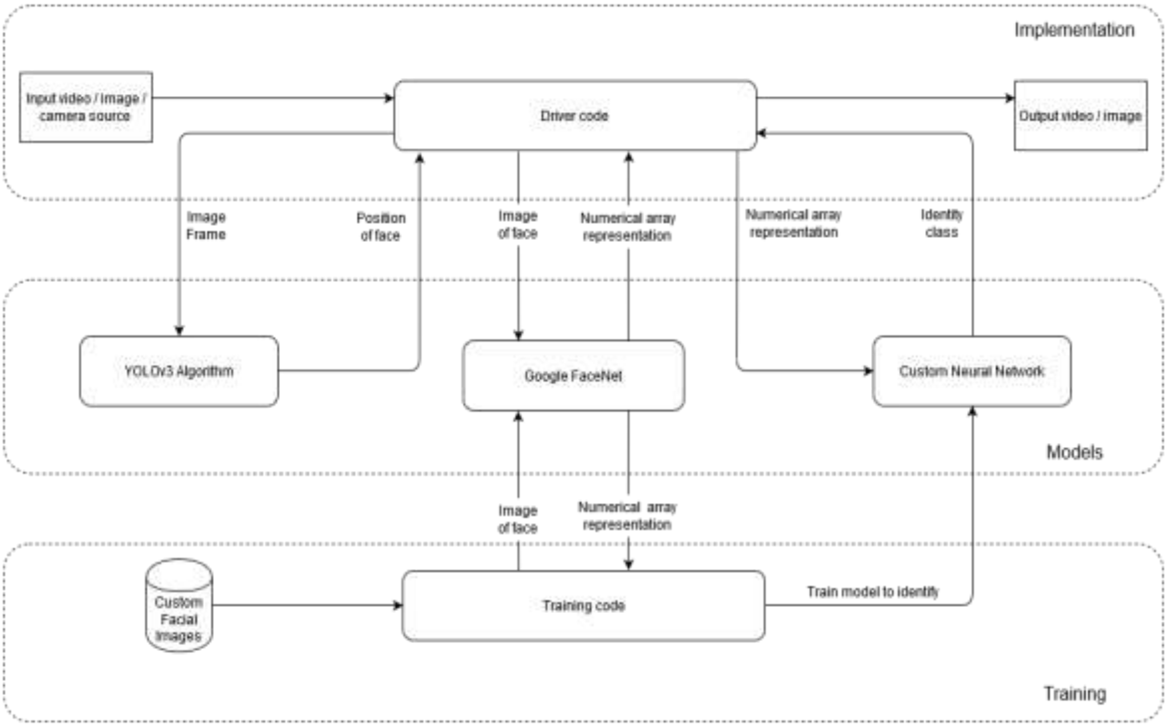


Figure 1 Proposed System

(The following sections will provide an insight into YOLOv3 and Google FaceNet)

A. Computer vision: [1][2][3]

Computer vision combines many branches of computer science that utilize machines to derive information from digital images or videos. It aims to replicate and hence automate functions of the human visual system from an engineering perspective. Computer vision has numerous applications such as collection, storage, and interpretation of digital images, and extraction of high-dimensional real-world data to generate numeric information to be used in decision-making processes.

B. Object recognition: [11][12]

Object recognition is a technique of computer vision for image or video classification of objects. Persons, items, acts, and graphic particulars can be spotted when people view an image or a video source. The goal is to impart onto a machine the working of what naturally is perceived by a human: to understand and interpret objects and their contents.

C. YOLOv3:[13][14]

"You Only Look Once" is an algorithm for detecting objects using convolution neural networks. YOLO is better in terms of speed with respect to procedures out there for object detection. Although it is not the most accurate algorithm for object detection, it is a very good choice in real-time detection, without losing too much precision. This algorithm passes the entire image through a single neural network. This gives the algorithm a tremendous boost in terms of speed.

For the purpose of this paper, a human face may be considered an object to be detected by the YOLOv3 model. Although YOLOv3 is an object detection algorithm, the model can be trained with a multitude of facial images so as to perform facial recognition and get the desired result. The system utilizes the InseptionResNetV1 to achieve the desired results. Pre-trained model weights to recognize faces using the said architecture can be found online.

D. Facial recognition: [4][5][8]

Facial recognition is an automated classification that mathematically finds a person's attributes regarding the face and saves the data as a face print. To achieve this it uses deep learning. The code recognizes a human face with several nodal points (often over 65 points) which are indicators that are utilized to calculate face differences such as nose dimensions, eye socket size, and cheekbone shape.[6] The system uses this information to differentiate between different faces.

Facial recognition utilizes CNNs (Convolutional Neural Networks). CNNs are heavily utilized for pattern recognition in Computer Vision. The numerical information extracted by facial recognition systems includes representations of several characteristics such as face width, face height, nose width, lips, eyes, width ratio, etc.

E. Google FaceNet: [7][9][10]

FaceNet is a face recognition software developed by Google researchers in 2015 that obtained state-of-the-art performance on a number of facial recognition benchmark datasets. It extracts features from a facial image in the form of 128-byte numerical array per object i.e. a face.

It reaches ninety-nine percent accuracy on the Labelled Faces dataset (LFW) and ninety-five percent on the YouTube Faces DB dataset in terms of verification and recognition of facial images.

II. Proposed Methodology

Over the years many facial recognition algorithms have been implemented concentrating on the speed of forward-pass (image input -> text output) and high accuracy. Most of these systems implement a single model that processes the entire image (or image frame of a video) using a single model to achieve the aforementioned objectives. The models hence turn out to be very complex. For instance, the CNN used for YOLOv3 for the purpose of this paper consists of 106 layers (75 CNN layers and 31 other layers). This results in a tremendously complex network, and hence large training times which may require days to train from scratch on moderately powered systems. The resulting facial-recognition model also becomes non-reusable for facial-recognition of new persons without a very long training period.

This paper proposes to use 3 distinctive models, 1 for detection, 1 for feature extraction, and 1 for recognition. In all honesty, this pattern does compromise on the speed of forward-pass, but it also helps drastically reduce the training time of the entire system, as will be demonstrated further. The YOLOv3 model detects a face and the Google FaceNet model extracts features from it. These 2 processes may be considered common between recognition of all possible datasets of facial images.

A third decision model of minimal complexity (1 input layer, 1 or 2 hidden layers, and 1 output layer) can be used to perform final recognition. The input size of the third model is fixed as 128 which the output size of the second model, i.e. Google FaceNet. As the third model is not complex it can be retrained in minutes for different datasets.

YOLOv3 and Google FaceNet are used in conjunction to create 128-byte array representations of all images in the new dataset. This doesn't take too long as it requires a single forward pass per image to achieve its 128-byte representation. These numeric representations are then used to train the third model in a matter of minutes if not seconds.

Hence the largest part of the recognition system i.e. the detection and feature extraction models become reusable across all datasets.

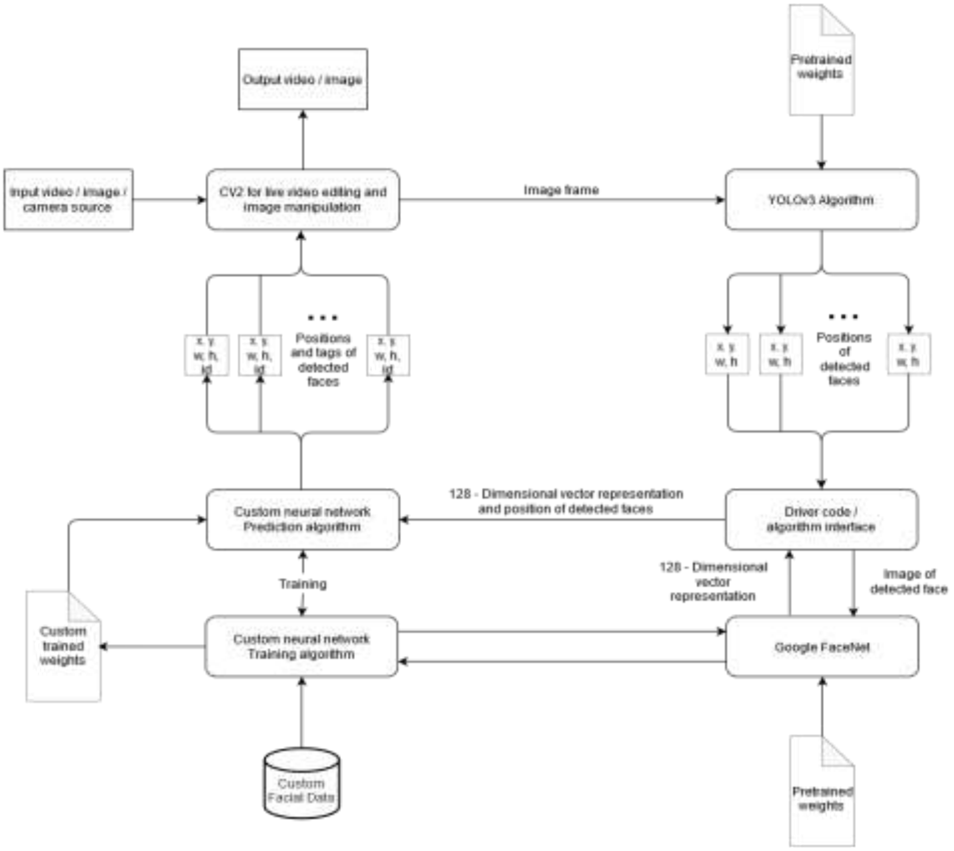

Output video / image
Pretrained weights
Input video / image / camera source
CV2 for live video editing and image manipulation
Image frame
YOLOv3 Algorithm
x, y, w, h, id
x, y, w, h, id
...
Positions and tags of detected faces
x, y, w, h, id
x, y w, h
x, y w, h
...
Positions of detected faces
x, y w, h
128 - Dimensional vector representation and position of detected faces
Custom neural network Prediction algorithm
Driver code / algorithm interface
128 - Dimensional vector representation
Image of detected face
Training
Custom trained weights
Custom neural network Training algorithm
Google FaceNet
Custom Facial Data
Pretrained weights

Figure 2 Proposed Decision Flow

A total of 7 candidates were chosen for the model to recognize, and a sample video (mug-shots) of them was acquired to populate the model. A third party application is used to frame burst the sample videos and get separate frames.

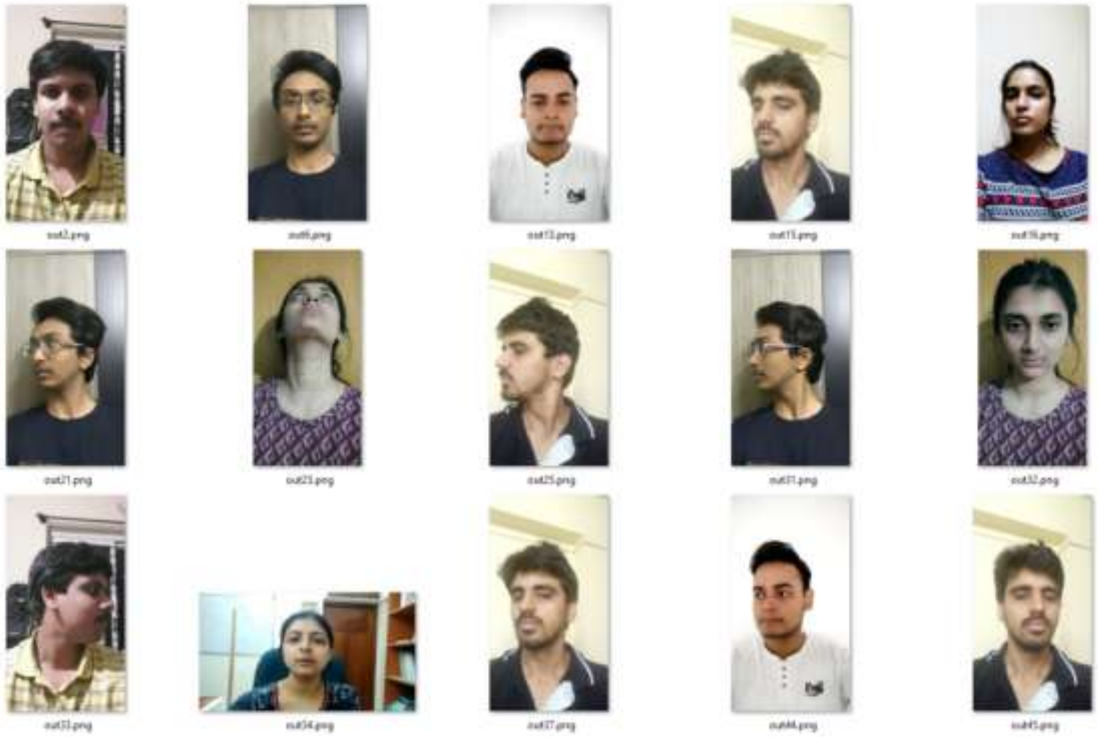


Figure 3Sample of raw input images (mug shots)

The image dataset is given the exact same treatment as the input during implementation, i.e. forward pass the image through a prepared YOLOv3 model, use the locations of faces to cut out facial images from the frame, and feed these to the FaceNet model to get a 128-byte representation of the images.

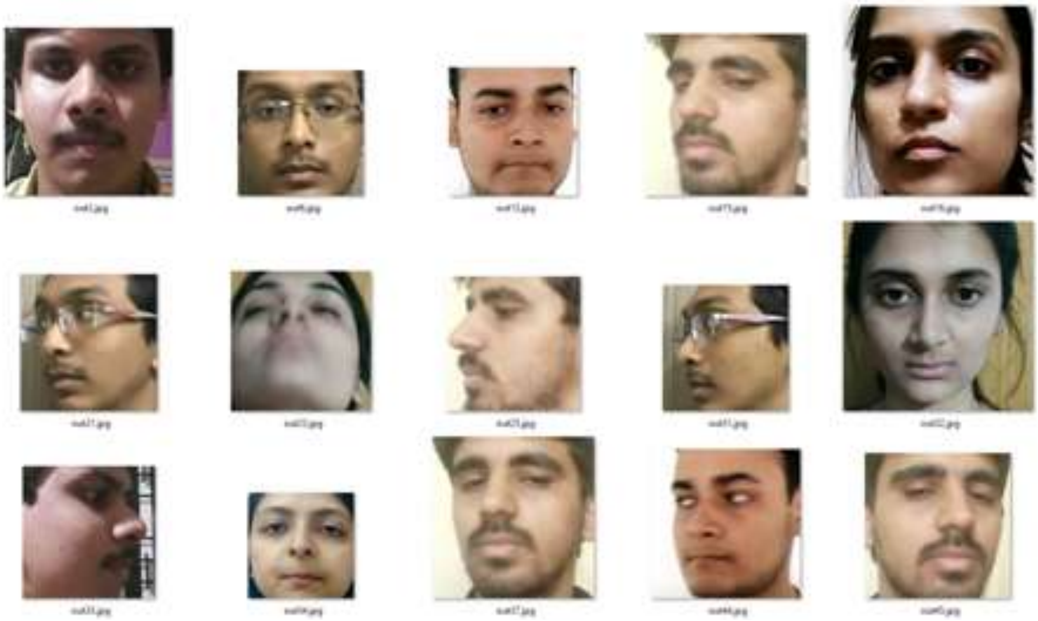
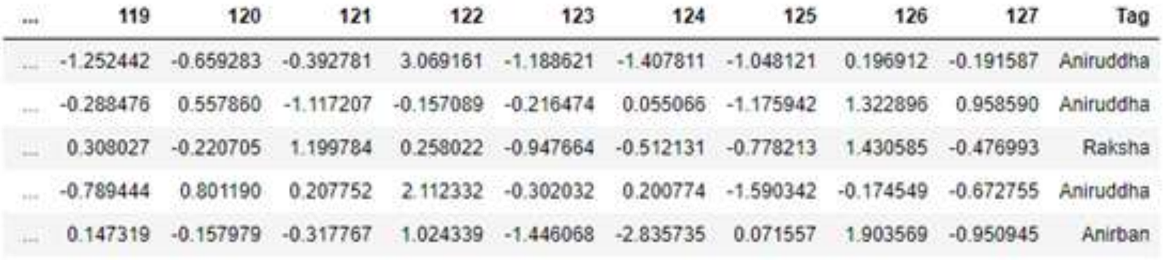


Figure 4Sample output of YOLOv3 saved as images

The above is the transformation provided by YOLOv3. As is evident from the difference between Figure 3 and Figure 4, the YOLOv3 model trained to detect faces provides the location of the face in the image, and hence we can get rid of the excess background noise and zoom into only the face for the Google FaceNet model to process.



...	119	120	121	122	123	124	125	126	127	Tag
...	-1.252442	-0.659283	-0.392781	3.069161	-1.188621	-1.407811	-1.048121	0.196912	-0.191587	Aniruddha
...	-0.288476	0.557860	-1.117207	-0.157089	-0.216474	0.055066	-1.175942	1.322896	0.958590	Aniruddha
...	0.308027	-0.220705	1.199784	0.258022	-0.947664	-0.512131	-0.778213	1.430585	-0.476993	Raksha
...	-0.789444	0.801190	0.207752	2.112332	-0.302032	0.200774	-1.590342	-0.174549	-0.672755	Aniruddha
...	0.147319	-0.157979	-0.317767	1.024339	-1.446068	-2.835735	0.071557	1.903569	-0.950945	Anirban

Figure 5 Sample FaceNet representation values

These FaceNet representation values (128 byte-arrays) are used to train the third customized model.

III. Results

Here is a sample of the output produced by a program implemented in accordance with the system proposed in this paper. It is a single frame of the entire video that was processed:

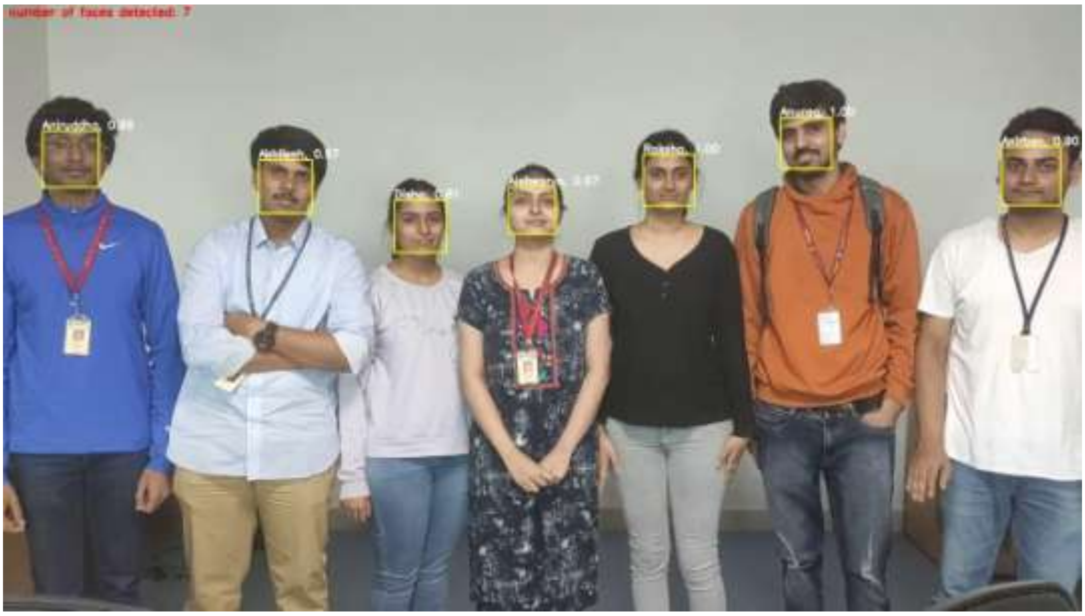


Figure 6 Sample output from YOLOv3 saved as images

Testing the speed of forward-pass:

The video file which was fed to the program was an 18-second long 1920x1080p video at 15 fps (reduced with a third party application). More precisely the number of frames to be processed was exactly 279.

The program took 1 min 12 sec to finish execution. This means the frames were processed at 3.88 frames per second without the use of a GPU.

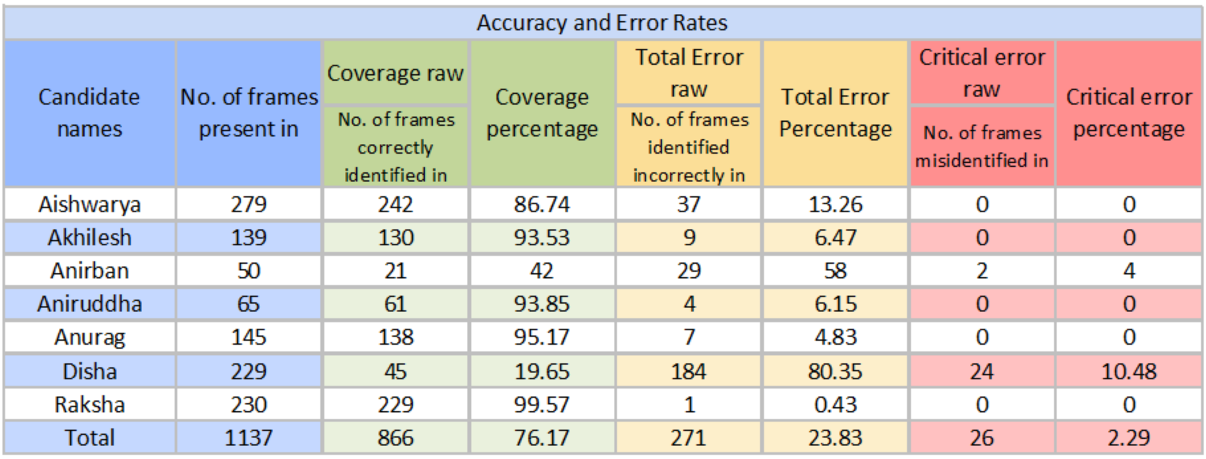
The program was run in an environment with limited resources (2.60 GHz i7 processor, 8GB RAM and ***no GPU***). Its speed performance will be drastically enhanced on a platform with larger resources or a distributed system.

Now looking into the accuracy:

Three different kinds of values have been measured:

1. No. of frames correctly identified in (Coverage):
   As the name suggests, this value indicates the no. of frames in which a person was in the frame and was correctly identified. This is the value to be maximized.
2. No. of frames incorrectly identified in (Total error):
   This value describes the no. of frames in which a person was in the frame, but was either not identified due to a low confidence score or identified as the wrong person.
3. No. of frames misidentified in (Critical Error):
   This value describes the no. of frames in which a person was falsely identified as the wrong person with a high confidence score. This is considered the most critical error.

The evaluation was done manually, and the following were the scores retrieved:



Accuracy and Error Rates

Candidate names	No. of frames present in	Coverage raw	Coverage percentage	Total Error raw	Total Error Percentage	Critical error raw	Critical error percentage
		No. of frames correctly identified in		No. of frames identified incorrectly in		No. of frames misidentified in	
Aishwarya	279	242	86.74	37	13.26	0	0
Akhilesh	139	130	93.53	9	6.47	0	0
Anirban	50	21	42	29	58	2	4
Aniruddha	65	61	93.85	4	6.15	0	0
Anurag	145	138	95.17	7	4.83	0	0
Disha	229	45	19.65	184	80.35	24	10.48
Raksha	230	229	99.57	1	0.43	0	0
Total	1137	866	76.17	271	23.83	26	2.29

Figure 7Accuracy and error data

The following is a graphical representation of the accuracy and error data:

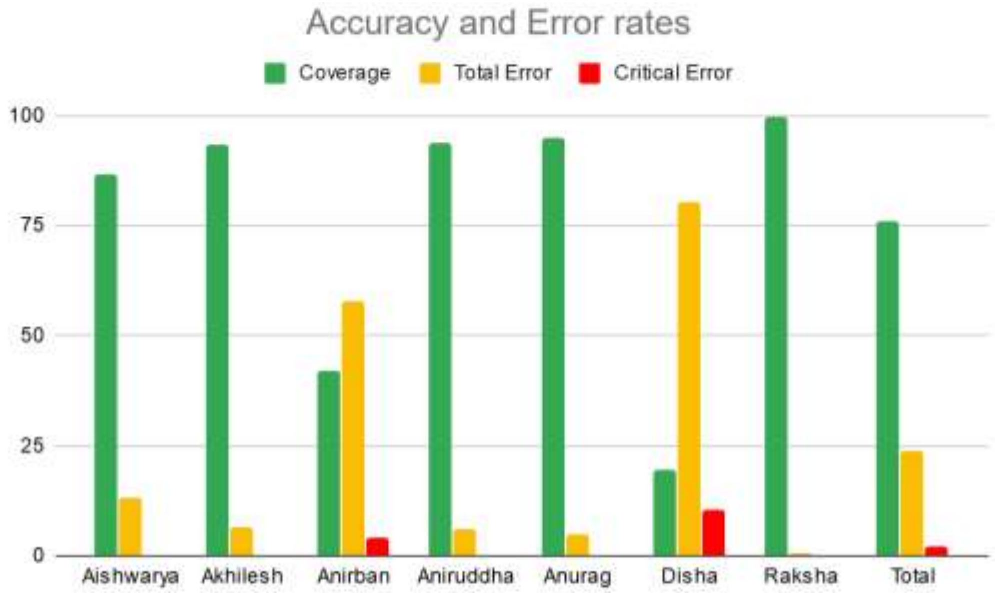


Figure 8Accuracy and error graph

Total Coverage Percentage:	76.17%
Total Error Percentage:	23.83%
Total Critical Error Percentage:	2.29%

Training speed:

The third neural network took a training time of 11.50 seconds to complete 200 epochs over 1930 numerical representation of facial images of 7 individuals to reach an accuracy of 98.7%. It is evident from this result that the facial recognition model is now easily customizable for different image datasets.

IV. Conclusion

The biggest motivation of this system is that if a new face needs to be added to the database for recognition, or the system needs to be trained for an enitirely new dataset, all the models need not be retrained especially because YOLOv3 and FaceNet are really large models and take extensive amounts of time to train, similar to other CNNs. Although the proposed system compromises by increasing the number of models and hence the duration of the forward pass it is compensated by the heavily reduced overhead of training time.

References

[1]. Wiley, Victor & Lucas, Thomas. (2018). Computer Vision and Image Processing: A Paper Review. International Journal of Artificial Intelligence Research.

[2]. Wu, Juan & Peng, Bo & Huang, Zhenxiang&Xie, Jietao. (2013). Research on Computer Vision-Based Object Detection and Classification. IFIP Advances in Information and Communication Technology. 392. 183-188. 10.1007/978-3-642-36124-1_23.

[3]. Gupta, Abhishek. (2019). Current research opportunities of image processing and computer vision. Computer Science. 20. 10.7494/csci.2019.20.4.3163.

[4]. Anwarul, Shahina & Dahiya, Susheela. (2020). A Comprehensive Review on Face Recognition Methods and Factors Affecting Facial Recognition Accuracy. 10.1007/978-3-030-29407-6_36.

[5]. Lal, Madan & Kumar, Kamlesh & Arain, Rafaqat&Maitlo, Abdullah & Ruk, Sadaquat & Shaikh, Hidayatullah. (2018). Study of Face Recognition Techniques: A Survey. International Journal of Advanced Computer Science and Applications. 9. 10.14569/IJACSA.2018.090606.

[6]. Jafri, Rabia & Arabnia, Hamid. (2009). A Survey of Face Recognition Techniques. JIPS. 5. 41-68. 10.3745/JIPS.2009.5.2.041.

[7]. Schroff, Florian & Kalenichenko, Dmitry & Philbin, James. (2015). FaceNet: A unified embedding for face recognition and clustering. 815-823. 10.1109/CVPR.2015.7298682.

[8]. Kulkarni, Hrishikesh. (2018). Deep Learning for Facial Recognition.

[9]. William, Ivan & Setiadi, De Rosal Ignatius Moses & Rachmawanto, Eko & Santoso, Heru & Sari, Atika. (2019). Face Recognition using FaceNet (Survey, Performance Test, and Comparison). 1-6. 10.1109/ICIC47613.2019.8985786.

[10]. Jose, Edwin & Manikandan, Greeshma & T P, MithunHaridas& M H, Supriya. (2019). Face Recognition based Surveillance System Using FaceNet and MTCNN on Jetson TX2. 10.1109/ICACCS.2019.8728466.

[11]. Sharma, Anshika. (2016). A survey on object recognition and segmentation techniques.

[12]. Uçar, Ayşegül&demir, Yakup&Güzeliş, Cüneyt. (2017). Object recognition and detection with deep learning for autonomous driving applications. SIMULATION. 93. 003754971770993. 10.1177/0037549717709932.

[13]. Redmon, Joseph &Farhadi, Ali. (2018). YOLOv3: An Incremental Improvement.

[14]. Zhao, Liquan & Li, Shuaiyang. (2020). Object Detection Algorithm Based on Improved YOLOv3. Electronics. 9. 537. 10.3390/electronics9030537.

Anirban Ghosh. "Multi Target Facial Recognition System Using Deep Learning." *IOSR Journal of Computer Engineering (IOSR-JCE)*, 22(4), 2020, pp. 01-07.