

Face Detection and Recognition for Criminal Identification System

Sanika Tanmay Ratnaparkhi
CSE Department
Amity University
Noida, Uttar Pradesh, India
sanikaratnaparkhi@gmail.com

Aamani Tandasi
CSE Department
Amity University
Noida, Uttar Pradesh, India
aamani99tandasi@gmail.com

Shipra Saraswat
CSE Department
Amity University
Noida, Uttar Pradesh, India
sshipra1510@gmail.com

Abstract— The process of identifying and spotting a criminal is slow and difficult. Criminals, these days are getting smarter by not leaving any form of biological evidence or fingerprint impressions on the crime scene. A quick and easy solution is using state-of-the-art face identification systems. With the advancement in security technology, CCTV cameras are being installed at most of the buildings and traffic lights for surveillance purposes. The video footage from the camera can be used to identify suspects, criminals, runaways, missing persons etc. This paper explores a way to develop a criminal identification system using ML and deep neural networks. The following method can be used as an elegant way to make law enforcement hassle-free.

Keywords—machine learning, face recognition, neural networks, criminal identification, CCTV

I. INTRODUCTION

Throughout the years, tracking down a criminal has been a difficult process. Earlier, the entire method consisted of leads based on evidence found on the crime scene. Biological evidence can be easily tracked down. However, criminals have evolved and are smarter than ever in terms of covering tracks and not leaving behind any kind of traceable evidence. Face recognition and detection come into play here. The face is significant for human identity and due to its distinguishable nature, every face is unique. Face recognition for criminal identification is one of a kind biometric technique that possesses the merit of high accuracy and low intrusiveness. It is a technique that uses the person's face to automatically detect and verify their identity from video frames or images.

The face identification system presented in this paper is a unique combination of the best techniques available today for face detection, feature extraction and finally classification. The deep learning methods like MTCNN for detection and FaceNet for embeddings have previously been proven to be elegant and state of the art.

Automatic face recognition is a method where the system extracts meaningful facial features such as the length of the nose or jawline, the distance between the eyes, the color of the eye, etc. These features are useful for classification and performing matches with the database. There are two significant processes involved in this system: Detection and Identification. Face recognition triggers two major methods: training and evaluation. Training deals with feeding the algorithm with a sample of images model is trained on the training set. The evaluation phase of the face recognition compares the newly acquired test image with the already existing database [1].

This paper is structured in the following manner. The next section deals with the literature review, the third section

consists of an overview of face recognition, the fourth section explains approach towards building the model, MTCNN and FaceNet, the fifth section is the implementation process. The sixth section consists of results & discussion, the seventh and eighth sections are conclusions and future work respectively.

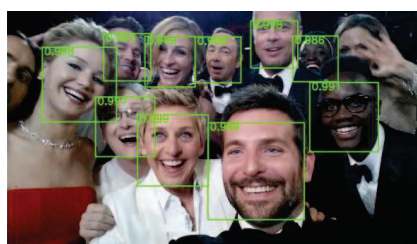


Fig.1. An Example of Face Detection

II. LITERATURE REVIEW

The applications of face recognition have been evolving since the 1960s as mentioned in [2] which coined the approach of using a RAND tablet for coordinating features on the face. A RAND tablet was a device which could be used with a stylus emitting electromagnetic pulses to input vertical and horizontal coordinates on a grid. The whole system was used to record the coordinate locations of several facial characteristics manually, such as the eyes, hairline, mouth and nose.

Reference [3] takes face recognition to a different level by using 21 special facial features like the color of hair, chin, nose elevation, skin color etc.

During the late 1980s [4] & [5] introduced the world to eigenfaces and statistical approach to face recognition. Eigenfaces use Eigenvalues and Eigenvectors to reduce dimensionality and project a sample/training data on small feature faces. This is the main idea behind Principal Component Analysis (PCA).

The first instance of using face recognition for law enforcement was seen in 2002. From that point forward Criminal Identification has become a major application of face recognition. [6] Uses dimensionality reduction technique- Principal component analysis for creating a criminal identification system known as "FRCI". [7], [8] and [9] uses the Haar-Features method as explained in [10]. In a Haar features system, the detection window consists of solid rectangles at the place of specific features. A Haar-like

feature considers adjacent rectangular regions in a detection window at a specific location, summaries the intensities of the pixels in each region, and calculates the difference between these sums. In [11] Adriana Kovashka, Margaret Martonosi proposed a system which uses 18 features including RGB which finds usage in [12].

After 2010 neural networks have been successful to deal with computer vision problems like face recognition, [13] explains the usage of AdaBoost and ANN together to create a hybrid model “ABANN”. Many deep learning models have been applied specifically for face recognition such as Retinal Connected Neural Network, Rotational Invariant Neural Network, Back Propagation Neural Network, Fast Neural Network, etc.

The main focus of our project will be based on [14] (FaceNet) an embedding technique which maps facial features to a “compact Euclidean Face-Map” which is used to detect differences in the faces present in the database. A deep convolutional neural network approach is used, each face is mapped to 128 bytes and the task of recognition, detection, and clustering is carried out. An accuracy rate of over 95% has been observed for two datasets.

III. FACE RECOGNITION OVERVIEW

Face Recognition in simple words means identifying and recognizing an individual on the basis of pictures of their face. Face Recognition is a broader term which basically involves two tasks-

- 1) *Face Verification*: To verify if the person is the same as the one in the given dataset.
- 2) *Face Identification*: To identify who the person is.

The process of face recognition is divided into four parts-

A. Face Detection

The task of detecting faces from an image, surrounding it using bounding boxes and cropping out unnecessary parts of the image. The detection process is the foundation for further tasks as a face cannot be identified before it is recognised in a picture. An image may contain more than one face. This step is achieved using various object detection techniques. The main task of these models is to find the position of the face and the coordinates of the extent of face. Then a bounding box is made using this information. The face detection models can be divided into two categories-

- 1) *Feature-Based Detection using filter matching technique.*
- 2) *Image-Based using Neural Networks*

B. Face Alignment

The detected faces need to be normalised with images present in the database for consistency. Certain models require images to be a particular size, colour, channel etc hence images need to be aligned in that order.

C. Face Extraction

Extracting features from face to make feature vectors. To perform matches the most significant features of a face needs to be converted to vectors. Feature extraction makes data manageable and reduces dimensionality.

D. Face Recognition

The final task of matching the given face with faces from the database and providing results.

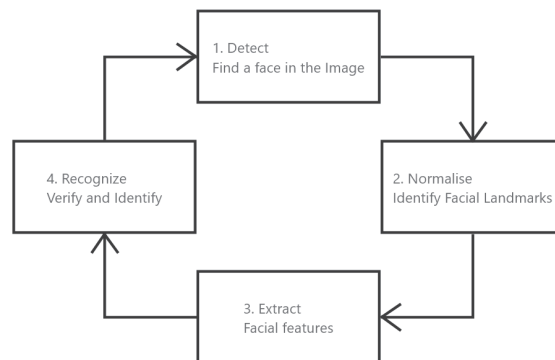


Fig. 2. Example of a figure caption.

IV. FACE RECOGNITION APPROACH

To make a reliable facial recognition system various technique need to be implemented one after the other for detection, creating embeddings and recognition.

A. Multi-task Cascaded Convolutional Networks(MTCNN)

MTCNN is based on [15], it presents a method for facial detection and alignment in pictures. It consists of 3-part CNN which can recognize landmarks on faces such as nose, forehead, eyes etc. There are 3 stages to mtcnn. In the first one, the image is resized to create a pyramid of images so detection for every size can be done then it is passed through a neural network known as P Net which gives coordinates of face and bounding box as output. In the second stage faces which are only partly visible are dealt with an R net is used to give bounding boxes as output. The result from P net and R net is mostly similar. In the third and final stage, an O net is applied which gives three outputs- coordinates, landmarks on face and confidence level of bounding boxes. After every stage, a non-max suppression method is used to remove bounding boxes with low confidence.

B. FaceNet

FaceNet [16] was presented in 2015 by a team of researchers from Google. The recognition, verification and clustering task for face identification was dealt with uniquely. FaceNet employs neural networks and a “triple loss function” to achieve highly accurate results. The input given to the model is the detected face and the output is a face embedding which is a vector containing 128 elements which represent unique features in this face. The deep CNN used is trained using a triplet loss function. It is based on the concept that feature vectors of identical faces will be more similar than of different faces. After the embeddings have been made simple machine learning techniques like k-NN, SVM can be used for identification.

1) *Triplet Loss*: A triplet loss function defines two images of the same person (which contains one anchor) that should be more similar than images of a different person. This means the L2 distances of embeddings between the anchor and positive image is less than that of anchor and negative image.

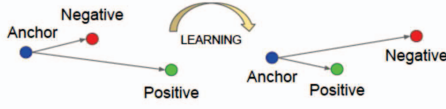


Fig.3. Visualization of triplet loss

Mathematically triplet loss function can be represented as-

$$\sum [\|f(x^a_i) - f(x^p_i)\|_2^2 - \|f(x^a_i) - f(x^n_i)\|_2^2 + \alpha]_+ \quad (1)$$

In (1) x_i are Images, $f(x_i)$ are embeddings of x_i and α difference between image pairs.

Selection of correct images to be paired is extremely important as multiple set pairs can check the conditions and the model will fail to learn accurately. The criteria to select positives and negatives is as follows-

$$\text{Argmax} \|f(x^a_i) - f(x^p_i)\|_2 \quad (2)$$

$$\text{Argmin} \|f(x^a_i) - f(x^p_i)\|_2 \quad (3)$$

According to (2) the distance between a positive image and anchor should be maximised. And in (3) distance between anchor and negative should be minimised.

2) *Training of CNN model*: The CNN model is trained using "Stochastic Gradient descent" which aids optimisation of the triplet loss function. AdaGrant is used for learning rates of every CNN layer. A ReLU activation function is used. FaceNet CNN architecture contains namely two models as follows.

a) *Zeiler and Fergus Model*

b) *Inception Model*

3) *Evaluation of FaceNet*:

- For correct classification- True accepts

$$TA(d) = \{(i,j) \in P_{\text{same}}, \text{ with } D(x_i, x_j) \leq d\} \quad (4)$$

- For incorrect classification – False accepts

$$TA(d) = \{(i,j) \in P_{\text{diff}}, \text{ with } D(x_i, x_j) \leq d\} \quad (5)$$

In (4) and (5) P_{same} is the pair of same identities, P_{diff} is the pair of different identities, $D(x_i, x_j)$ is square of L2 distance between pairs and d is the distance threshold.

V. IMPLEMENTATION OF THE CRIMINAL IDENTIFICATION SYSTEM

To implement FaceNet one can use various open-source models which are pre-trained or train a model on their own dataset. The most famous open-source FaceNet model is OpenFace which is trained using PyTorch another one is known as "FaceNet by David Sandberg" which is trained using TensorFlow. For the purpose of the Criminal Identification system "FaceNet by Hiroki Tanai," a pre-

trained Keras model is used. This Keras model was trained on "MS-celeb-1M dataset". For the criminal Identification System, a "Criminals Dataset" is used which contains around 200 snapshots of most wanted criminals in the world. These pictures are clicked at various angles, poses and illumination with/without headgear, sunglasses, accessories etc. Some pictures also have no visible faces. The data set is divided into two parts: a training set and a validation/test set. The step-wise implementation is-

A. Face Detection

MTCNN was directly implemented using the MTCNN library by "ipazc/mtcnn project". All images were loaded as a NumPy array and converted to RGB values. Then an MTCNN face detector class was made to detect faces in the image. The output was a list of bounding boxes where length and width were identified from the x and y-axis. After faces from all images have been detected they are stored as a compressed file.



Fig.4. An example of the detected face

B. Face Embeddings

Face embeddings need to be created so comparisons with different vectors can be done. This is the step where the FaceNet model was used for creating embeddings. After loading the compressed file of detected faces the pixel values need to be standardized as it is required for FaceNet. The pre-trained Keras facenet model is loaded. Each face is enumerated to find its prediction and embedding from the train and test set. The embeddings were saved as a compressed NumPy array.

C. Face Classification

In this part of the process, embeddings are classified using machine learning models to be identified as one of the criminals. Before applying a classification, model vector normalization is applied so values are scaled. The scikit learn normalization library is used for this purpose. Next, the names of the criminals are converted from string to integer format. This is done using LabelEncoder of scikit learn. The classification model used is Linear Support Vector Machine as it is effective for differentiating between the face embeddings. The linear SVM model is fit on the training data.

D. Plotting Faces

To visualize the working of this entire model a face from the compressed test set is picked. Then embeddings for this image are created. This face embedding is used as input to fit in the model and get predictions.

VI. RESULTS AND DISCUSSION

The above implementation was done in the python language in a jupyter notebook. In the face detection phase pixels for the face and bounding boxes were perfectly created. The below image shows pixels for detected faces of one criminal from Fig.3.

```
[[[ 41 41 41]
[ 49 49 49]
[ 62 62 62]
...
[203 203 203]
[195 195 195]
[191 191 191]]
[[ 20 20 20]
[ 21 21 21]
[ 20 20 20]
...
[ 45 45 45]
[ 45 45 45]
[ 43 43 43]]

[[ 38 38 38]
[ 46 46 46]
[ 57 57 57]
...
[205 205 205]
[194 194 194]
[188 188 188]]
[[ 18 18 18]
[ 18 18 18]
[ 18 18 18]
...
[ 40 40 40]
[ 41 41 41]
[ 43 43 43]]

[[ 39 39 39]
[ 47 47 47]
[ 57 57 57]
...
[204 204 204]
[194 194 194]
[188 188 188]]
[[ 20 20 20]
[ 19 19 19]
[ 18 18 18]
...
[ 44 44 44]
[ 43 43 43]
[ 43 43 43]]]
```

Fig.5. Pixels for detected faces of one criminal

Moving further the face embeddings for all the images in the dataset using FaceNet were created. It is clearly observed that the FaceNet model is loaded 40 and 88 images were converted to face embedding each containing 128 vectors.

```
Loaded Model
WARNING:tensorflow
(40, 128)
(88, 128)
```

Fig.6. Loaded model and created face embeddings


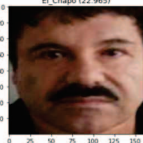



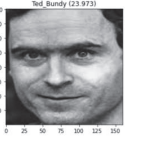





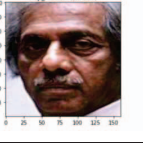
In the classification phase, the model is trained and after evaluation on training and testing dataset, the following accuracies were obtained.

Accuracy: train=92.500, test=90.909

Fig.7. Classification accuracy

Finally, random images were plotted and the identity of the random face is predicted along with the probability. The following table shows the results.

TABLE I. PREDICTIONS FOR VARIOUS RANDOMLY PICKED IMAGES ALONG WITH PROBABILITY

Image	Expected Name	Predicted Name and probability	Result snapshot
	El Chapo	El Chapo 22.965	Predicted: El_Chapo (22.965) Expected: El_Chapo El_Chapo (22.965) 
	Veerapan	Veerapan 12.919	Predicted: Veerapan (12.919) Expected: Veerapan Veerapan (12.919) 
	Ted Bundy	Ted Bundy 23.973	Predicted: Ted_Bundy (23.973) Expected: Ted_Bundy Ted_Bundy (23.973) 
	Osama Bin Laden	Osama Bin Laden 24.925	Predicted: osama_bin_laden (24.925) Expected: osama_bin_laden osama_bin_laden (24.925) 
	Joseph Kony	Joseph Kony 20.957	Predicted: Joseph_Kony (20.957) Expected: Joseph_Kony Joseph_Kony (20.957) 
	Haji Mastan	Haji Mastan 29.167	Predicted: Haji_Mastan (29.167) Expected: Haji_Mastan Haji_Mastan (29.167) 

VII. CONCLUSION

This paper presents an innovative approach to face recognition and how it can be implemented for an important purpose which is Criminal Detection and identification. Face Recognition technologies have a wide range of applications, similar approaches can be used for solving a lot of real-world problems. We believe developing a system as such is an enthusiastic step towards making the process of catching criminals and law enforcement speedy and efficient. This system can be further implemented to detect

criminals in real-time using a dynamic dataset. Application of computer vision can be challenging but create solutions to difficult problems easier.

VIII. FUTURE WORK

An elegant face identification system like this can be automated to detect criminals through CCTV cameras installed at multiple places. This system can also be used to detect missing people at the time of disasters and mishappenings. This system can be extended to identify multiple faces at once and identify from images which are blurry or cropped. Criminal identification system can also give details of where the criminal was exactly spotted using locations of cameras. The database can also incorporate more details such as age, crimes committed, associated people last spotted etc. to provide additional details of the criminal.

REFERENCES

- [1] P. M. Corcoran and C. Iancu, "Automatic face recognition system for hidden markov model techniques," *New Approaches to Characterization and Recognition of Faces*, pp. 3-28, 2011.
- [2] Bledsoe, "Manual measurements", 1960.
- [3] A.J. Goldstein, L.D. Harmon and A.B. Lesk, "Identification of human faces," in *proceedings of the IEEE*, vol 59, pp. 748-760, May 1971.
- [4] L.Sirovich and M.Kirby, "Low dimensional procedure for the characterisation of human faces," in *Journal of the Optical Society of America A*, vol 4, pp. 519-524, 1987.
- [5] M. Turk and A. Pentland, "Eigenfaces for Recognition," in *Journal of cognitive neuroscience*, vol 3, pp. 71-86, Jan 1991.
- [6] N. A. Abdullah, Md. J. Saidi, N. H. A. Rahman, C. C. Wen, and I. R. A. Hamid, "Face recognition for criminal Identification: An implementation of principal component analysis for face recognition," *AIP Conference Proceedings* 1891:1, Oct 2017.
- [7] P. Kakkar and V. Sharma, "Criminal identification system using face detection and recognition," in *International Journal of Advanced Research in Computer and Communication Engineering*, vol 7, pp. 238-243, March 2018
- [8] P.Apoorva, H.C. Impana, S.L. Siri., M.R.Varshitha and B.Ramesh, "Automated criminal identification by face recognition using open computer vision classifiers," in *2019 3rd International Conference on Computing Methodologies and Communication (ICCMC)*, pp. 775-778, 2019
- [9] P. Chhoriya, "Automated criminal identification system using face detection and recognition", in *International Research Journal of Engineering and Technology (IRJET)*, vol 6, pp. 910-914, Oct 2019.
- [10] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision And Pattern Recognition*, vol 1, pp. 511-519, Feb 2001.
- [11] A. Kovashka and M. Martonosi, "Feature-based face recognition for identification of criminals vs. identification for cashless purchase".
- [12] A. Chevelwalla, A. Gurav, S. Desai and S. Sadhukhan, "Criminal face recognition system," in *International Journal Of Engineering Research & Technology (IJERT)*, vol 4, pp. 47-50, March 2015
- [13] T. H. Le, "Applying artificial neural networks for face recognition", in *Hindawi Publishing Corporation Advances in Artificial Neural Systems*, vol 2011, pp. 1-16, 2011
- [14] F. Schroff, D. Kalenichenko, J. Philbin, " FaceNet: A unified embedding for face recognition and clustering", in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815-823, 2015.
- [15] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multi-task cascaded convolutional networks".
- [16] F. Schroff, D. Kalenichenko and J. Philbin, " FaceNet: A unified embedding for face recognition and clustering," in *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 815-823, 2015.