Face Detection and Recognition with Mask

Mohd Asif and Deepali Kamthania
Decimal Technologies, School of Information Technology,
Vivekananda Institute of Professional Studies, Delhi, India

Vivekananda Journal of Research
October, 2021, Vol. 10, Special Issue, Pg No.121-126
ISSN 2319-8702(Print)
ISSN 2456-7574(Online)
Peer Reviewed Refereed Journal
© Vivekananda Institute of Professional Studies
http://www.vips.edu/vjr.php

Abstract

Face or facial feature detection, and face recognition from picture or video is a famous point in biometrics research. The public places have surveillance cameras for video catch and these cameras have their huge incentive for security reason. It is the undertaking of making a positive distinguishing proof of a face in a photograph or video picture against previous information in database. It starts with face detection, recognizing human appearances from different items in the database and then works on identification of those detected faces. However, episode of novel Covid 2019 (COVID-19) has prodded the critical improvement of biometric acknowledgment advances that can analyse and recognize subjects wearing facial mask, particularly in high security applications. In this paper an attempt has been made to identify and perceive the subjects wearing mask and propose a framework for picture-based face detection and recognition without removing mask with higher exactness, better response rate as an initial step for video surveillance. The proposed framework dependent on tests performed on different data of with and without mask face.
Keywords: Face Detection, Face Recognition, Mask-Detection, Face Identification.

1. INTRODUCTION

Face mask and facial recognition will be the norm in the coming days and it's amazing how quickly the community has accepted it. Before pandemic most of us would look with suspicion if someone has face masked in public but it is a reversed in today's scenario. Face recognition with mask involves using a computer to identify people's faces in photos or videos and to measure certain facial features. This can include the distance between the eyes, and the related areas of the nose, faces posture, and the shape of face. This information is combined to create a face signature, or profile. When used for personal recognition - such as turning on the phone, a picture from the camera is compared to a recorded profile in registry offices. This process of verification on the face with mask can be applied. However, when facial recognition is used to identify faces in a crowd, it requires a significant database of profiles against which to compare the main image. These profiles can be legally collected by enrolling large numbers of users into systems**.** Over the past decade a lot of work has been done on facial detection and recognition [1] as it is the best way to identify a person [2] because it does not require human collaboration [3] to be a hot topic in biometrics. As more and more methods are introduced the discovery and recognition [4, 5, 6, 7] is considered historic. Although these methods have been used several times for the same purpose in terms of a limited amount of data in the past, very less work has been found to provide face detection and recognition with mask and by testing them on solid data sets. In the current work the system is designed to test the method used as the first landmark for image detection and recognition with a visual mask. The

human face is a sophisticated multidimensional structure that can convey a lot of information about the individual, including expression, feeling, and facial features [8]. Effectively and efficiently analysing the features related to facial information is a challenging task that requires lot of time and efforts.

In our research work, is divided into two main phases: The first phase focuses on improving facial detection while the second phase focuses on finding a masked face based on the face of a well-known person. In the first step, images will be stored in the face database of the logs and then the Haar cascade algorithm will be used for photographs to detect individual faces, which will be used as in Face Recognition Program.

2. RELATED WORK

In computer vision many studies has been carried out on face detection. Before 2000, despite many extensive studies, the practical performance of face detection was far from satisfactory until the milestone work proposed by Viola and Jones [9][10]. The VJ framework [9] was the first one to apply rectangular Haar-like features in a cascaded Adaboost classifier for achieving real-time face detection. However, it has several critical drawbacks like large feature size, in detection window of 24*24 there were 160,000 Haar-like features [10] and it could not non-frontal faces and faces efficiently. The detection robustness was enhanced by combining multiple detectors being trained separately for different views or poses. This was used by multiple deformable part models to capture faces with different views and expressions. With discriminative learning a retrieval-based method was combined.

Many other approaches like random forest, SVM classifiers using Dlib C++ Library [11] have been used for face detection but they are time consuming and less efficient. Recently, Jian Sun at el. [12] constructed a model to perform face detection in parallel with face alignment, and achieved high performance in terms of both accuracy and speed. Compared to the traditional method face detection using deep learning methods have better performance like Faster R-CNN [12] for generic object detector, integration of ConvNet and 3D mean face model in multi-task discriminative learning an end-to-end framework.

3. FACE RECOGNITION METHODOLOGY

The overview of current system is demonstrated in Fig.1

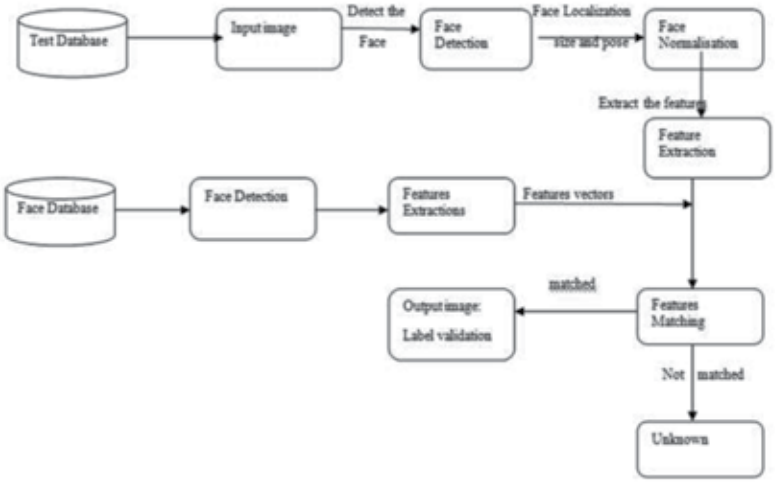


Figure1: System's Overview

3.1 Face Detection and Feature Extraction

In face detection the photo is searched for a face and then to make recognition easier the image is processed to crop and extract the face of an individual. The Dlib facial landmark detector generates 68 (x, y) coordinates that correspond to certain facial structures. Training a shape predictor on the labels yielded 68 point mappings as shown in Fig. 2.

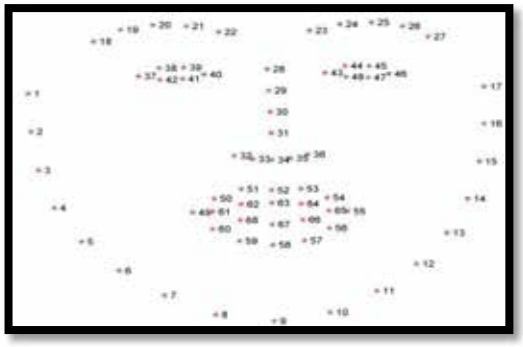


Figure 2: Facial landmarks using dlib and python

It is observed from Fig. 2 that facial regions can be accessed using zero python indexing. The various features can be extracted using the following points. The mouth [48, 68], nose [27, 35], jaw [0, 17] right and left eyebrow [17, 22] and [22, 27] and right and left eye [36, 42] and [42, 48] respectively.



Figure 3: Facial extraction using dlib and python

Fig. 3 shows identifying faces in photos or videos, but this information isn't enough. To create powerful applications, we need more information about the person's face, like position, whether the mouth is opened or closed, whether the eyes are opened, closed, looking up and etc.

3.2 Face Recognition

In face recognition the detected and processed face is compared with the face database. This can be done using OpenCV [1], Intel's open-source framework, having an inbuilt Face Detector with 90-95% accuracy with person looking straight at camera. The 3D head pose estimation is required to detect a person face from an angle. The brightness, blurry picture can affect the recognition. For better recognition eigenface technique and combinations of multiple methods is applied.



(a) (b)

Figure 4: Face recognition with Opencv dlib and python

Fig. 4 shows facial recognition via dlib and Python using the `face_recognition` module method and a 128-d real-valued number feature vector per face has been generated.

3.3 Dataset

The dataset used in this project compromises of two main folders one with mask and another without mask. The test of the image is being processed by taking an image with mask and then checking in the folder of without mask. If the coordinates if the found image matched with the test image, then it creates a rectangle around the face and tag a name otherwise tag with "unknown".

4. EXPERIMENTAL RESULTS

Step1: In this step, system would be given an image from the training dataset. The image should be in either .jpeg or .png or .jpg format. Then system will extract the face features using dlib in the backend and then store in into the encoding variable.



Figure 6: Input image

Step2: In this step, system will extract the image's key and values and store it into the encodings and face_name. Then we applied the "face_location" method of "face_recognition" that returns an array of bounding boxes of human faces in an image. A list of tuples representing (top, right, bottom, left) location of found face is stored in css.

Step3: In this step, now we run a loop for the image that returns the 128-dimention face encoding for each face in the image from the database to find the match. Then the comparison is made between the database image and the given image.



Figure 7: Output image

Fig. 7 shows the output image with the label of Bill gates as it is saved in the database with that name and the other person is not in the database hence, he is unknown for the system. If the distance calculated by the Euclidean distance are the minimum between the given image and the image in the database then put label to the image in the database and prints the result.

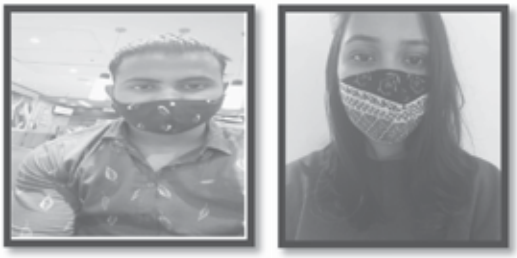


Figure 8: Input masked image from mask database

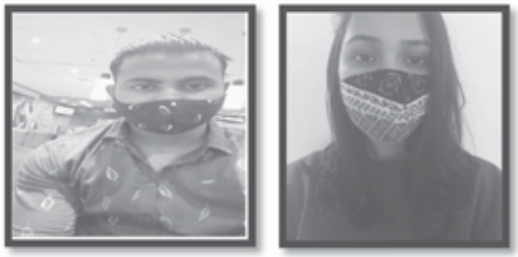


Figure 9: Output Image with the tagged name from the face dataset.

Step 4: In this step, we applied the same steps for the masked faces and it again do the above said process and find the maximum matched image and show the result with 97% accuracy.

CONCLUSION

In the paper a system for face detection and recognition with masking and unmasking has been developed. It has been implemented using the dlib face_recognition. The experiments are performed more consistently than with different data sets and some experiments behave randomly but based on the performance of standard test results are tested, two sets of data are used for this purpose. One is for a masked faces and the other is for unmasked faces. In the current system that uses dlib with facial features

has some false positives as well, which could be considered a future work in view of reducing false positives in this project.

REFERENCES

1. W. Zhao, R. chellappa, P. J. Phillips, Face recognition: A literature survey, "ACM Computing Surveys (CSUR)", December 2003.
2. Suman, Automated face recognition: Applications within law enforcement. Market and technology review "NPIA", 2006.
3. G. L. Marcialis, F. Roli, Chapter: Fusion of Face Recognition Algorithms for Video-Based Surveillance Systems, Department of Electrical and Electronic Engineering- University of Cagliari- Italy.
4. T. Ahonen, A. Hadid, M. Peitikainen, Face recognition with local binary patterns. "In Proc. of European Conference of Computer Vision", 2004.
5. M. A. Turk and A.P. Pentland, Face recognition using eigenfaces, "Proceedings of the IEEE", 586-591, 1991.
6. J Lu, K. N. Plataniotis, A. N. Venetsanopoulos, Face recognition using LDA-based algorithms, "IEEE Neural Networks Transaction", 2003.
7. L. Wiskott, M. Fellous, N. Krger, and C. Malsburg, Face recognition by elastic bunch graph matching, "IEEE Trans", on PAMI, 19:775–779, 1997.
8. Serign modou bah, Fang ming, An improved face recognition algorithm and its application in attendance management system, Changchun University of Science and Technology, Changchun, China.
9. Paul Viola and Michael Jones. Rapid object detection using a boosted cascade of simple features. In Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on, volume 1, pages I–511. IEEE, 2001.
10. Paul Viola and Michael J Jones. Robust real-time face detection. International journal of computer vision, 57(2):137–154, 2004.
11. Davis E. King. Dlib-ml: A machine learning toolkit. Journal of Machine Learning Research, 10:1755–1758, 2009.
12. Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In Advances in neural information processing systems, pages 91–99, 2015.