EasyChair Preprint

№ 6547

Multiple Face Analysis and Liveness Detection Using CNN

Logeswari Saranya and K Umamaheswari

September 7, 2021

MULTIPLE FACE ANALYSIS AND LIVENESS DETECTION USING CNN

[1]Logeswari Saranya R [2]Umamaheswari K
[1]Research Scholar, [2]Professor, Department of Information Technology, PSG College of Technology, Coimbatore

Abstract: *In the fast-developing world, there is always a need for authentication for most systems. Face Recognition is one such system. Face recognition is the process of identifying or verifying a person from an image captured from either image capturing device or an individual frame in a video. A computer is not capable of processing such high-level processes by itself. For detecting and recognize faces, advanced concepts like deep learning can be used. Face Recognition is used in several fields such as login mechanism, unlock the device, User Authentication, etc. They can also play a pivotal role in recognition of multiple areas such as Student Attendance, Entry Cameras, etc. where the multiple persons may enter/present at a single instance of time. Also, there is a vulnerability for security breaches using still images of printed or digital images. The multiple face recognition and liveness detection models detect multiple faces in the image and recognize them based on the sample images provided and also verifies the liveness of identified person in the frame.*

Index Terms*: Multiple Face Recognition, Liveness Detection, Convolutional Neural Network.*

1. INTRODUCTION

Facial image processing is an area of research dedicated to the extraction and analysis of information about human faces; information that is known to play a central role in social interactions including recognition, emotion, and intention. Over the last decade, it has become a very active research field that deals with face detection and tracking, facial feature detection, face recognition, facial expression and emotion recognition, face coding, and virtual face synthesis.

With the introduction of new powerful machine learning techniques, statistical classification methods, and complex deformable models, recent progress have made possible a large number of applications in areas such as image retrieval, surveillance and biometrics, visual speech understanding, virtual characters for e-learning, online marketing or entertainment, intelligent human-computer interaction, and others.

However, much remains to be done to provide more robust systems, especially when dealing with pose and illumination changes in complex natural scenes. If most approaches focus naturally on processing from still images, emerging techniques may also consider different inputs. For instance, video is becoming ubiquitous and very adorable, and there is a growing demand for vision-based human-oriented applications, ranging from security to human-computer interaction and video annotation. Capturing 3D data may as well become very adorable and processing such data can lead to enhanced systems, more robust to illumination effects, and where discriminate information may be more easily retrieved.

2. LITERATURE SURVEY

2.1 Face Detection

Face detection algorithms are a very integral part of any facial analysis system depending on the ability to identify the human face part on an image. Due to the increasing demand for face recognition systems in the past, face detection algorithms are used in many real-life applications. To resolve this problem, different methods have been proposed such as Cascade Classifier [1], MTCNN [3], Dlib HOG [5], DlibCNN[2].

2.2 Face Recognition

Although the face recognition method for in-depth learning has high accuracy, the model is complex and the recognition speed is slow. To realize real-time face recognition of students while learning video, a real-time face recognition method based on Dlib is proposed. The researchers [4, 6] describe the actual methods and technologies for all stages of the development of the recognition system since in the field of recognition, a huge number of unique solutions have been developed. Also, scholars [9, 10] describe the method of recognition using the method of support vector machine (SVM), which can significantly improve the speed of the recognition process. Researchers [8] describe face landmarks estimation algorithms that are used to position faces. It enables you to increase the quality of the system by aligning the face for better recognition.

2.3. Liveness Detection

The face is most significant according to these properties so the face is also used for recognition systems due to its unique property and is widely used in the area which needs security [7].To protect from the attacks, it is necessary to carry out liveness detection in the recognition system. Liveness detection or spoof detection improves the robustness of the recognition system by determines whether the input image is of a real subject or non-real subject i.e. the image of a real subject [11].

Many different IQA based techniques are developed by authors for presentation attacks [13]. Another texture-based approach is static [14] and dynamic [12]. The multi-task cascaded convolutional neural networks (MTCNN) are used to achieve rapid face detection and face alignment, and then the FaceNet with improved loss function is used for face recognition[15].

3. PROPOSED SYSTEM

To use Multiple Face Recognition and Liveness Detection using CNN, the face must be detected. The camera must be placed in the way that person's face appears clear enough. Then it is frame by frame analyzed to detect the face and recognized by the face recognition module. Finally, it checked for liveness detection.

CNN in Multiple Face Recognition and Liveness Detection involves three modules:

- Face Detection
- Face Recognition and
- Liveness Detection.

To detect multiple faces in the given image frame the model should be capable and the detected faces must be recognized based on the given dataset. Finally, faces must be evaluated for liveness and classified whether it is life or not.

Camera or mobile is used for capturing real-time video. The video is read frame by frame and it is store on the laptop. Next, the image is processed for face detection. Based on the detection, the face recognition model recognizes the person. Finally liveness detection model classifies whether it is life or not.

3.1. System Work Flow

The main idea of the project is to create a face recognition model using CNN. The camera or mobile should be able to capture the real-time images and send them to the laptop and later every video frame is processed by series of steps to detect the face using the MTCNN package to help face recognition and liveness detection model to handle things easier. Finally, the output video with bounding boxes is processed and saved.

Figure 3.1: Proposed System Work Flow

3.2. Dataset for Face Recognition

The "100-Bollywood-celebrity-faces" dataset from Kaggle is used for Face Recognition.

Images	12,400
Persons	100
Images / person	100-130
Size	2.0 GB

Table 3.1: Dataset – 100 Bollywood Celebrity Faces

3.2.1. Dataset Preprocessing

Figure 3.2: 100 Bollywood Celebrity Faces:

Preprocessing-cropping the face from the original image.

3.3. Dataset for Liveness Detection

The "ROSE-Youtu Face Liveness Detection Dataset" is used for Liveness Detection.

Videos	3350
Subjects	25
Size	5.45 GB
Video Clips	150-200/subject
Video Duration	10 secs
Standoff Distance	30-50 cm

Table 3.2: Dataset – ROSE Youtu Face Liveness Detection

3.3.1. Dataset Preprocessing

The Dataset Preprocessing for Liveness Detection involves extracting images from the video and cropping the face using the MTCNN model. Dataset Preprocessing involves converting the video into 3 image frames. The Image frame is created from the video of 10 secs at

- 1st Second (beginning)
- 5th Second (middle)
- 10th Second (end)

Then the faces in the images are cropped using the MTCNN model. Then the cropped faces are trained.

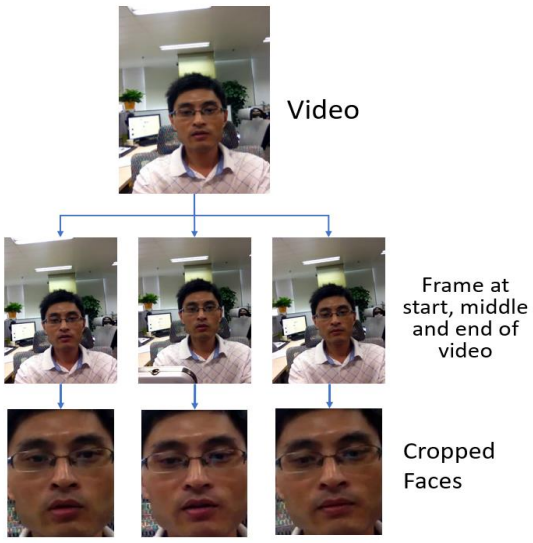


Figure 3.3: ROSE Youtu Dataset – Preprocessing

3.4. Frame by Frame Video Processing

A mobile camera installed with IPWebcam is used for capturing real-time videos. The captured frame-by-frame image is sent and read by the laptop since IPWebcam hosts the live camera in the local network in which the laptop is also connected. Thus, the image is read by the laptop. Once the image is captured and sent to the laptop, the model will process it in series of steps. The image from IPWebcam is processed using URL library in python and the image is read and it is sent to the face detection model for further process.

3.5. Face Detection

The following steps have been followed to get a detected faces as output from the video frames as inputs:

- Inputs RGB Image
- RGB color space is converted into BGR color space
- Image is passed to MTCNN model
- Coordinates of detected faces are returned
- Detected faces Coordinates are cropped from Image
- Outputs all the faces

3.5.1. Coordinates of detected faces

*{"**box**":[1942,716,334,415],*
*"**confidence**":0.9999997615814209,*
*"**keypoints**":{*
"left_eye":(2053,901),
"right_eye":(2205,897),
"nose":(2139,976),
"mouth_left":(2058,1029),
"mouth_right":(2206,1023)}}

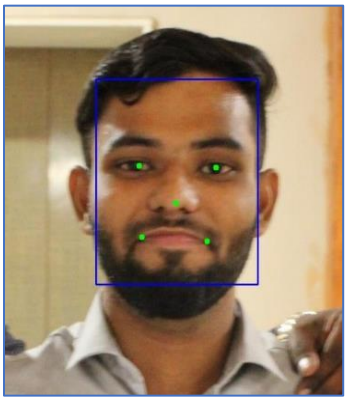


Figure 3.4: Points Obtained from Detected Face

The Coordinates of detected faces return as the list of dictionaries similar to the above-mentioned points. It also includes confidence of detected face and some key points.

- box – [x, y, width, height] representing the whole face where (x,y) is starting coordinates and (x+width,y+height) is ending coordinates of the face.
- All key points have point (x,y) values detecting respective positions in the detected face.

3.6. Face Recognition

To recognize face the following steps are to be followed to generate video with face-recognized labels from input videos:

- Dataset
  - Encoding
- Working
  - Encode
  - Compare faces
  - Votes
  - Label

3.6.1. Dataset Encoding

- Encoding the faces involves extracting facial features using the dlib library.
- For every face, 128-d embedding is generated.
- They are mapped with the label name and stored in an encodings pickle file which is later used by the model.
- For every single image, the encoded value is stored with a label.

The function generates encodings as output from the detected faces inputs. The encodings are stored with the label name(Person's name).

3.6.2. Working

Face Recognition involves in series of steps given below

- Load and detect the faces (Face Detection).
- The detected faces are compared with all the encodings stored previously.
- For every match, the count of that label is incremented.
- Label with the maximum count is considered as the desired label.
- Then face is labeled with the desired label or 'Unknown' if there are no matches.

3.7. Liveness Detection

The following steps are involved in Liveness Detection which takes the image frame as input and generates output as image frame labeled with detected class.

- Dataset
  - Training
  - Encodings
- Working
  - Encodings
  - Predict Class
  - Label

The dataset consists of 4 classes namely Genuine, Mask, Paper Print, and digital Photo. The Genuine class is considered real and Paper Print and Digital Photos are together considered fake. Here the mask is not considered, since it falls under paper print.

For Class Classification, 3350 Videos of four classes are reduced to 1781 Videos of two classes. Each class contains the following number of images

- Real
  - Genuine - 2668
- Fake
  - Photo print – 1315
  - Digital - 1430

Figure 3.5: ROSE Youtu Dataset – Classes (Genuine, Paper Print, and Digital)

3.7.1. Dataset for Training and Encoding

The Dataset is trained using TensorFlow.Keras using the following steps

- Fix the Epochs Size and Training-Test Size
- Images are read
- Image resized to 32*32 pixels
- Extract Region of Interest
- Based on the value, it is encoded with the label.
- Based on the encodings with labels(class), the model is trained and tested for the given train-test size, and model and encodings are saved.

3.7.2. Working

The Liveness Detection involves in following steps

- Load Encodings
- Read Image frame with a detected face
- Resize face to 32 x 32 pixels
- Extract face Region of Interest
- Predict the given face ROI and label using stored encodings
- And label the image.

4. EXPERIMENTAL RESULTS

4.1. Face Detection

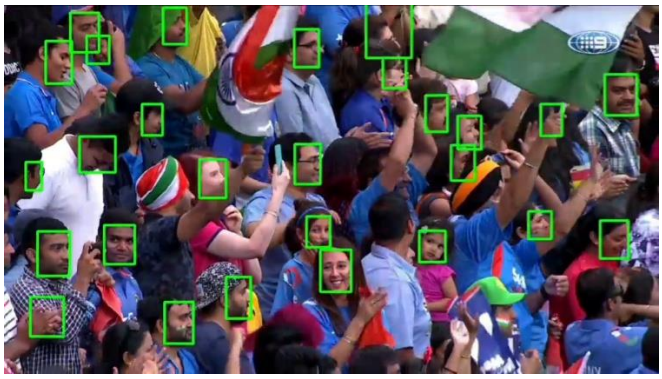


Figure 4.1: Results – Face Detection

4.2. Face Recognition

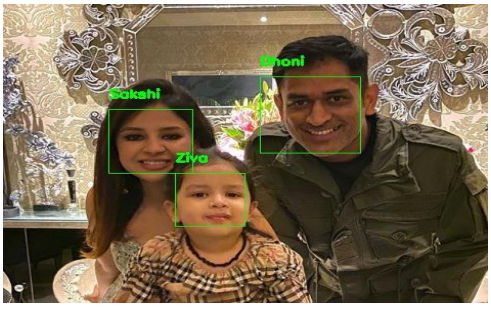
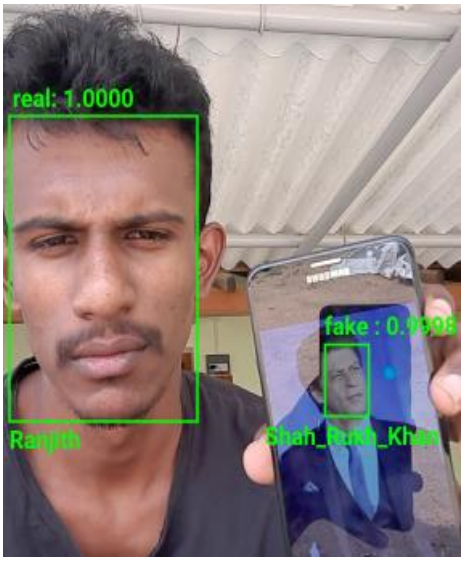


Figure 4.2: Results – Face Recognition

4.3. Liveness Detection



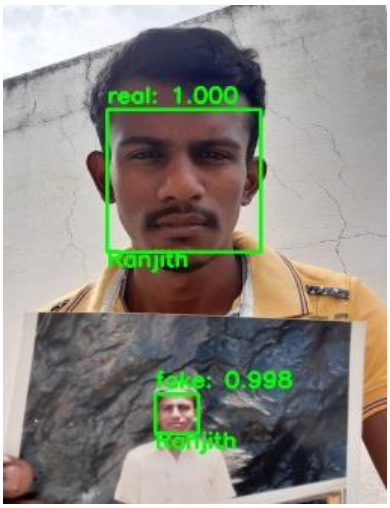


Figure 4.3, 4.4: Results – Liveness Detection

4.3.1. Accuracy based on Liveness Detection

The model is trained on the various train and test data splits such as 70-30,80-20 and 90-10. And the 80-20 split was found to be effective. Each split ran for 100 Epochs.

Train-Test Split	70-30	80-20	90-10
Training Accuracy	94%	99%	90%
Training Loss	10%	1%	25%
Validation Accuracy	98%	99%	97%
Validation Loss	5%	3%	10%

Table 4.1: Accuracy – Liveness Detection

5. CONCLUSION

This model is based on recorded existing videos and found to be working effectively. The camera, face detection & recognition model, and liveness detectors were integrated successfully. The model was tested and evaluated using existing and recorded videos. The Concept of face recognition plays important role in the advanced human authentication system. Since most of the organization and offices rely on CCTV footages for monitoring purposes, the combination face recognition system with the live feed, play a crucial role for an automated monitoring system. Thus, the applications of the Face Recognition Systems are not limited.

REFERENCES

[1] Paul Viola Michael Jones, Rapid Object Detection using a Boosted Cascade of Simple Features, ACCEPTED CONFERENCE ON COMPUTER VISION AND PATTERN RECOGNITION 2001

[2] FAREC - CNN Based Efficient Face Recognition Technique using Dlib, Sharma S, Karthikeyan Shanmugasundaram, Sathees Kumar Ramasamy, International Conference on Advanced Communication Control and Computing Technologies (ICACCCT), 2016

[3] Analysis on Face Recognition based on five different viewpoints of face images using MTCNN and FaceNet By Al-Imran 15201007 Baniamin Shams 15301030 Faysal Islam Nasim 15201051. 2019.

[4] Z. Rybchak, and O. Basystiuk, "Analysis of computer vision and image analysis technics" ECONTECHMOD: an international quarterly journal on the economics of technology and modeling processes, Lublin: Polish Academy of Sciences, vol. 6, no. 2, pp. 79-84, 2017.

[5] Comparison of Haar-like, HOG, and LBP approaches for face detection in video sequences, Amal Adouani, WiemMimoun Ben Henia, ZiedLachiri, 16th International Multi-Conference on Systems, Signals & Devices (SSD'19),2019

[6] A toolkit for making real-world machine learning and data analysis applications [Electronic resource] – Access mode: https://github.com/davisking/dlib/blob/master/python_examples/face_detector.py

[7] Garg, S., Mittal, S. and Kumar, P., 2019. Performance Analysis of Face Recognition Techniques for Feature Extraction. Journal of Computational and Theoretical Nanoscience, 16(9), p.3830-3834.

[8] A. Rosebrock, Facial landmarks with dlib, OpenCV, and Python [Electronic resource] - Access mode: https://www.pyimagesearch.com/2017/04/03/facial-landmarksdlib-opencv-python/

[9] Support Vector Machines [Electronic resource] – Access mode: http://scikit-learn.org/stable/modules/svm.html

[10] F. Schroff, D. Kalenichenko, and J. Philbin, FaceNet: A Unified Embedding for Face Recognition and Clustering [Electronic resource] – Access mode: https://www.cvfoundation.org/openaccess/content_cvpr_2015/app/1A_089.pdf

[11] Chan, P.P., Liu, W., Chen, D., Yeung, D.S., Zhang, F., Wang, X. and Hsu, C.C., 2017. Face liveness detection using a flash against a 2D spoofing attack. IEEE Transactions on Information Forensics and Security, 13(2), pp.521-534

[12] Arashloo, S.R., Kittler, J. and Christmas, W., 2015. Face spoofing detection based on multiple descriptor fusion using multiscale dynamic binarized statistical image features. IEEE Transactions on Information Forensics and Security, 10(11), pp.2396-2407.

[13] Galbally, J. and Marcel, S., 2014, August. Face anti-spoofing based on general image quality assessment. In 2014 22nd International Conference on Pattern Recognition (pp. 1173-1178). IEEE.

[14] Raheem, E.A. and Ahmad, S.M.S., 2019. Statistical Analysis of Image Quality Measures for Face Liveness Detection. In 10th International Conference on Robotics, Vision, Signal Processing and Power Applications (pp. 543-549). Springer, Singapore.

[15] ChunmingWu, Ying Zhang,2021, January.MTCNN and FACENET Based Access Control System for Face Detection and Recognition. In Automatic Control and Computer Sciences 55, pp.102-112.