

# **FACE LIVENESS DETECTION IN PYTHON**

Project Report submitted in the partial fulfilment of the requirement for the  
degree of Bachelor of Technology

in

**Computer Science and Engineering/Information technology**

By

Deepika Gupta(161201)

Kanika Puri(161227)

Under the supervision of

Supervisor: Dr.Vivek Sehgal

to



Department of Computer Science & Engineering and information Technology  
**Jaypee University of Information Technology Wanknaghat, Solan-173234,**  
**Himachal Pradesh**

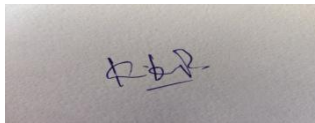
## CERTIFICATE

We hereby declare that the work presented in this report entitled "FACE LIVENESS DETECTION IN PYTHON" in partial fulfillment of the requirements for the award of the degree of **Bachelor of Technology in Computer Science and Engineering/Information Technology** submitted in the department of Computer Science & Engineering and Information Technology, Jaypee University of Information Technology Wakhnaghat is an authentic record of our own work carried out over a period from July 2019 to May 2020 under the supervision of Dr. Vivek Sehgal (Associate Professor in department of Computer Science and Engineering/Information Technology) and Dr. Hemraj Saini (Associate Professor in department of Computer Science and Engineering/Information Technology) .

The matter embodied in the report has not been submitted for the award of any other degree or diploma.



(Student signature)  
Deepika Gupta, 161201



(Student signature)  
Kanika Puri, 161227

This is to certify that the above statement made by the candidate is true to the best of my knowledge.



Supervisor Name:  
Designation:  
Department Name:  
Dated: 26-05-2020

(Supervisor Signature)  
Dr. Vivek Sehgal  
Associate Professor  
CS & IT

## **ACKNOWLEDGEMENT**

We have taken efforts in this project. However, it would not have been possible without the kind support and help of many individuals and organizations. We would like to extend our sincere thanks to all of them.

We are highly indebted by Dr. Vivek Sehgal and Dr. Hemraj Saini for their guidance and constant supervision as well as for necessary information regarding the project and also for their support in completing the project.

We would like to express our gratitude towards our parents and Jaypee University of Information Technology for their kind co-operation and encouragement which helped us in completion of the project.

Our thanks and appreciation also go to our fellow students in developing the project and people who are willingly helped us out with their abilities.

# TABLE OF CONTENTS

i. Certificate.....	ii
ii. Acknowledgement.....	iii
iii. Table of contents.....	iv
iv. List of Figures.....	v
v. List of Tables.....	vii
vi. Abstract.....	viii
1. Introduction.....	1
1.1. Introduction.....	1
1.2. Problem Statement.....	2
1.3. Objective.....	3
1.4. Methodology.....	4
1.5 Organization.....	7
2. Literature Survey.....	13
3. System Development.....	20
4. Performance Analysis.....	38
5. Conclusion.....	44
5.1 Conclusion.....	44
5.2 Future Scope.....	44
5.3 Application and Contribution.....	45
<i>References</i> .....	51
<i>Plagiarism Report</i> .....	55

## LIST OF FIGURES

Figure Number	Description	Page Number
1.1.	Fake images from dataset	4
1.2	Real images from dataset	5
1.3	Images depicting face as ROI	5
1.4	. Process of saving facial region of interest	6
1.5	The process of training LivenessNet. Using both “real” and “spoofed/fake” images as our dataset, we can train a liveness detection model with OpenCV, Keras, and deep learning.”	7
1.6	Haar Features	9
1.7	Convolutional Neural Networks using Dlib/OpenCV	9
1.8.	Convolutional neural networks model	11
3.1	Running gather_example.py script	30
3.2	Screenshot of fake images gathered by the script gather_examples.py	30
3.3	FNMR	31
3.4	Running train_liveness.py script	33
3.5	Screenshot depicting training of the network for 50 epochs	33
3.6	Screenshot depicting statics about the trained network, an accuracy of 0.99	34

3.7	Running liveness_demo.py script	35
3.8	Screenshot depicting starting of the video stream	36
3.9	Screenshots of frame window depicting various real and fake faces along with their individual probabilities	37
4.1	Haar features	38
4.2	Convolutional Neural Networks using Dlib/OpenCV	39
4.3	Plot.png depicting statistical data about the network	40
4.4	Also the model depicts faces correctly	42
4.5	Frame window screenshot depicting an error in the model	42

## LIST OF TABLES

Table Number	Description	Page Number
2.1	Comparative analysis of non-intrusive detection techniques	19
3.1	Keras Vs TensorFlow	26

## ABSTRACT

Facial recognition has become popular in the field of biometrics. The technology has developed rapidly as it is more direct, convenient and user friendly. But face recognition systems are vulnerable to spoof attacks made by non-real faces. Thus, the negatives can't be ignored as we live in a technology savvy world with hacking becoming more common, everyday. Spoofing may involve pictures and portraits. Hence, liveness detection can be a possible solution to this problem. Categorization helps understand different spoof attacks scenarios. The main aim is to provide a simple path for the future development of the model and more secure face liveness detection approach.

It can become a very popular in today's world demanding a contact-less authentication and biometric systems. The only problem related to this system is the breach and spoofs attacks which can shake the very basis and purpose of this system. Although, many anti-spoofing techniques like optical principles, texture and feature analysis etc. are under research and development, but integration of many systems can make it quite heavy and time-inefficient, which is also not a very suitable feature in today's fast moving world. Thus, integration of techniques and optimization are also important features that will have to be considered with the development of the system.



# CHAPTER 1

## INTRODUCTION

### 1.1 INTRODUCTION

Systems of facial recognition have become quite popular over the last few years, causing them to be included in our daily lives as an important inseparable digital enhancement. The existing examples involve mobile phones, building entries, security systems, cryptos and locking systems using facial recognition for authentication, widely. The benefits of the technology are also visible in industrial sector. The globally spread mobile industry is using facial recognition as an authentication system to provide an extra layer of biometric security. It is being used to make communities more secure by law enforcement agencies. Travellers' safety and comfort have benefitted at airports. The technology has also helped keep an eye on crimes and violence.

But the major feature here is being able to distinguish between fake faces and real faces. This will provide an extra boom to this popular facial recognition system.

This is an important aspect to ponder and work over because spoofing could destroy the very virtue of the technology by allowing spoofers to hack into and get through authentication, leaving data vulnerable to misuse and heavy losses. Thus, if an unauthorised person is able to get through the facial authentication step, the whole model becomes useless.

Amongst the common and popular technologies of biometrics such as handwriting verification, fingerprint sensing and scanning technologies, which have been growing and progressing recently, is the face recognition approach, which has gained popularity among the public as well as the IT hubs is because of its distinct and attractive features such as more directness, user friendliness and convenience. Therefore, it has been applied and used widely, in biometric authentication systems. But, in general, the face recognition algorithm is not able to separate the 'live' face from the fake face recognition system by facial pictures and portraits. To avoid such hacking, a safe system requires liveness detection. The significance of biometrics in today's society has been hugely reinforced by the requirement for recognition management systems for this is to avoid the perpetrators' fingerprint recognition and iris recognition communities recently. But in face recognition, approaches to deal with this problem are very limited. The liveness model's work is to differentiate the feature space in alive and non-living.

The very approach of distinguishing between users based on their property of whether living or not is known as liveness detection. Perpetrators can try to introduce a large number of hacked authentications into the systems. With the assistance of physiological property detection and verification, the accuracy of a biometric system can improve.

It's a very important and difficult concern that Liveness is especially supported by researching out patterns on facial features over a series of photos.

The model is based on finding a true user using approaches based on hardware training and combined with the software. The mostly utilized in characteristic of physiological property detection of the face space unit is 'Eye detection and alter detection of eyes.'

The Anti-spoof/anti-hacking downside had to be deciphered before face recognition system can be widely applied in our systems of authentication. A time period and non-intrusive methodologies assisted the diffusion velocity of one single image that is planned to tackle the problem of face spoofing and hacking of images or videos. Above all, the separation in surface properties between an alive face and a fake one is discovered efficiently and accurately within the diffusion speed. Anti spoofing/hacking options are exploited by utilizing the full variation flow theme. Additionally also, the process of the native patterns of the diffusion speed, the alleged "native speed patterns, because the options, that are input into the linear SVM classifier is planned to work out whether or not the given face is fake or not. One vital advantage of the planned approach is that, in contrast to previous approaches, it precisely identifies numerous malicious attacks" and hacks no matter the medium of the image, like, paper or screen. Although, the approach that has been proposed for use does not demand any particularities in user action. Results of the experiment on varied wide sets of data and information displays that the methodology that has been proposed is precisely effective for facial physiological property detection.

Thus, a lot of efforts are being made to make the system hack proof. This can be done by integrating eye-ball movement or light reflection or lip movement features. Technological advances can help make the system more and more secure.

## 1.2 PROBLEM STATEMENT

The growing use of mechanical autonomy and forging creativity in the present well-informed world makes it difficult to recognize whether an approved individual gets to desired assets or a vindictive individual can get to all the data.

Approach exploits improved learning and region standardization by expelling the summary involves representation for crossspace face line discovery tasks. Specifically, 3D Convolutionary Neural Networks (3D CNNs), which have been shown to be effective for behavior recognition assignments, are used to collect satirizing explicit data based on daily printed and video attack replay.

Approach exploits improved learning and region standardization by expelling the summary involves representation for crossspace face line discovery tasks. Specifically, 3D Convolutionary Neural Networks (3D CNN), which have been shown to be effective for behavior recognition assignments, are used to collect satirizing explicit data based on daily printed and video attack replay. These capabilities allow us to take another step in identifying assaults under obscure conditions or under different conditions. Face bioscience bolstered in confirmation frameworks, caricaturing assaults zone unit in some cases executed abuse pictures, recordings or cast covers. While it may also be necessary to carry out or rectify medical procedures, as suggested by the caricature, image and documenting territorial unit, in all probability the first natural wellsprings of mocking attacks.

Therefore, due to the growing existence of parodying interpersonal organization sites, the opportunity to be expressed was created.

### 1.3 OBJECTIVES

The main objective is to implement face detection algorithm to detect faces, and then predict whether whatever face displayed in front of the screen is real or fake.. The algorithmic formulation works in real time with the help of a webcam and recognises the face by displaying a frame around it. In simple language, the program runs as follows:

Collect dataset.

Fake and real images.

Train on the collected dataset.

Do repetitive training.

Run the code.

Opens webcam.

Detect faces or faces.

For each detected face, display a frame around it.

Predict the amount of realness or fakeness probability on top of the frame. Also, keep taking pictures of the frames to feed into dataset.

Plot, the accuracy .

The further steps involve detecting whether a person is live or somebody is trying to spoof by using liveness detection algorithms. For the detection and recognition of face, face recognition libraries are to be installed, that gives very useful deep learning methods for finding and identifying faces of an image. Particularly, the face locations ,the face encodings and comparison of faces functions are the 3 most valuable. The two methods that can be used for detection of face location are: Histogram of oriented Gradients (HoG) and Convolutional Neural Network (CNN). HoG method is used because of time restrictions.

Convolutuional Neural Network pre-trains the face encoding function which helps to convert the image into a 128 featured vector. The converts vector holds sufficient information to distinguish between 2 different persons.

Distance is calculated between the embedding vectors to compare faces. It will allow the algorithm to recognize face extracted from a webcam frame and compare its embedding vector with all encoded faces in the dataset. The vector with the closest set represents the same person.

## 1.4 METHODOLOGY

Considering liveness detection problem to be of binary type, two types of dataset have been used in the model, namely fake images and real images for training.

A convolutional neural network is trained as we input an image in the system to help it train to distinguish fake and real images.

The dataset contains some percent of our images inserted into the data set picked up from the online sources.

3 main directories constituted in the model are:

Dataset: consisting of two folders of images:

- o Fake images were taken by clicking pictures from pictures displayed on any electronic device.

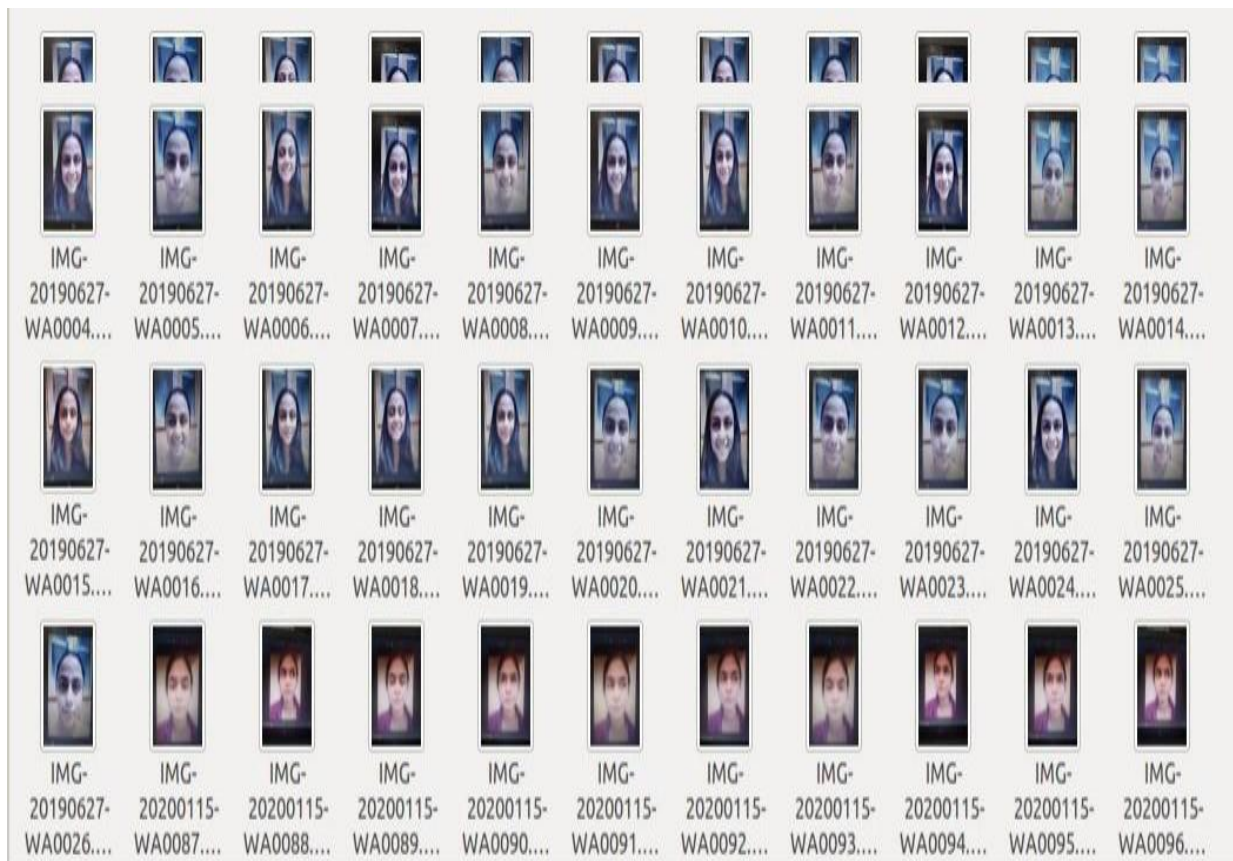


Fig1.1. Fake Images from dataset

Real images were inserted by taking own images from any camera.

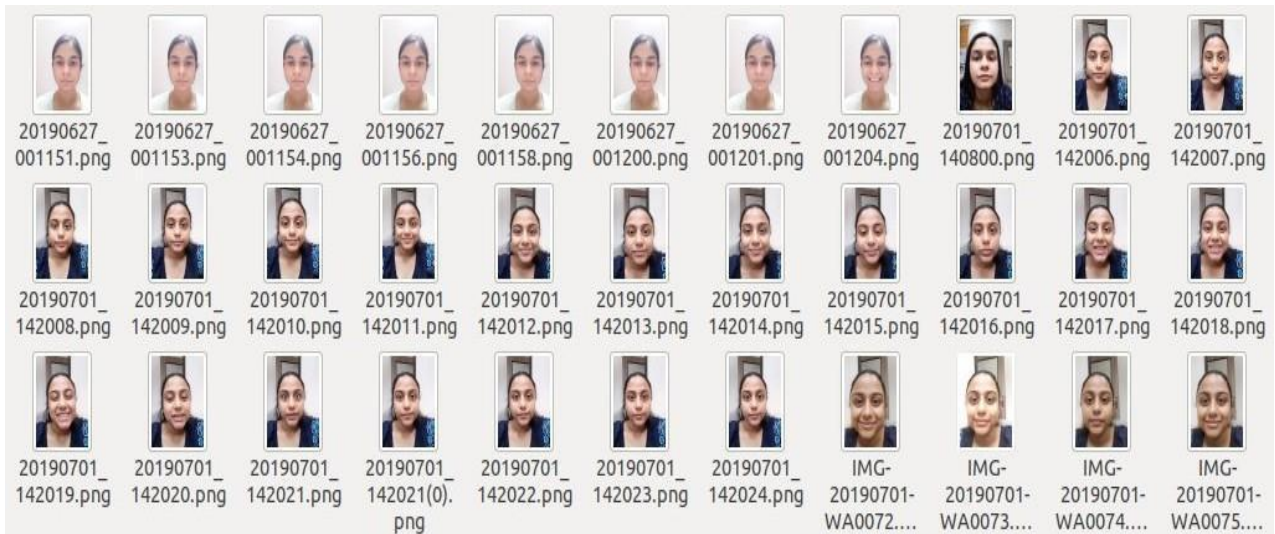


Fig1.2.Real Images from dataset

Face Detector: Consisting of a pre-trained Caffe face detector to locate face ROIs.

ROI is the Region of interest, which in this model is the face region of the picture. The dataset created was specifically aimed at inserting images of the torso only as we need only the facial region for processing. Also when the model starts the webcam the objects or things around the face need to be ignored and big faces near the screen, need to be taken into consideration .



Fig1.3. Images depicting face as ROI

## Python scripts:

`gather_examples.py`: This is the script that extracts face ROIs from input video files for the creation a diversified dataset for face liveness detection model.

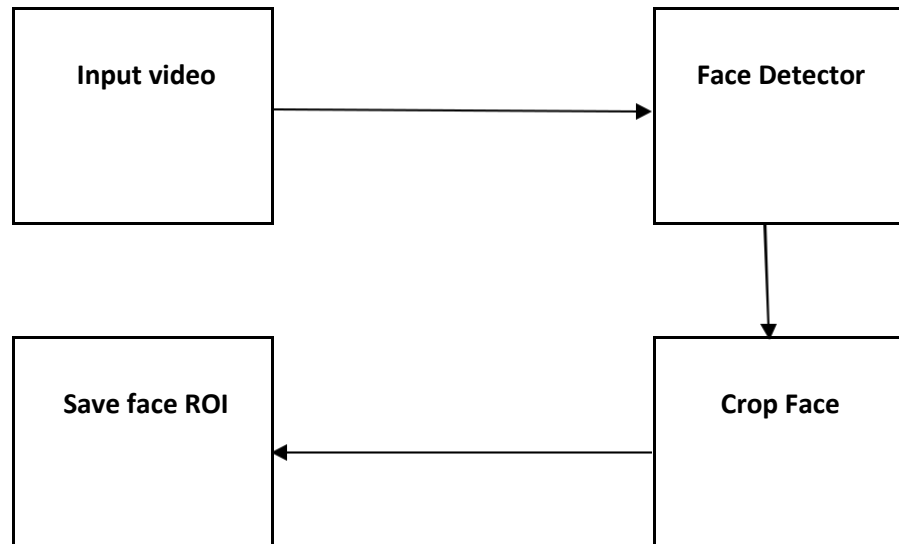


Fig1.4.Process of saving facial region of interest

`train_liveness.py` : It is very clear from the filename itself that it will train LivenessNet classifier.Keras and TensorFlow were used as basic tools to train the model.The training process results in a few files:

`le.pickle`: The class label encoder. (Label Encoding means to transform the labels into numeric form so that they become into the machine-readable format.)

`liveness.model`: Here the serialized Keras model come into play, which detects face liveness.

`plot.png`: The training history plot is a performance indicator showing accuracy and loss curves assisting in assessing the model (i.e. overfitting/underfitting).

`liveness_demo.py`: This script for demonstration will fire up the webcam,extracting frames to apply face liveness detection in real-time.

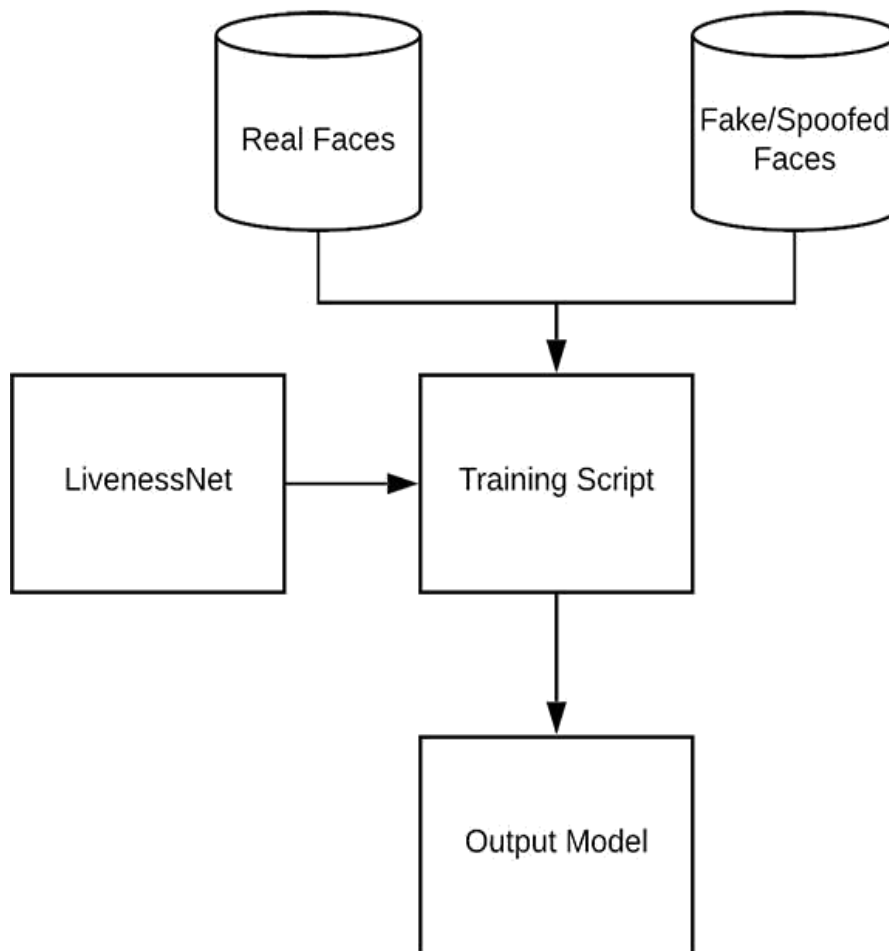


Fig1.5. The process of training LivenessNet. Using both “real” and “spoofed/fake” images as our dataset, we can train a liveness detection model with OpenCV, Keras, and deep learning

## 1.5 ORGANIZATION

### Geometric Based / Template Based Facial recognition System

Algorithms used for face recognition are categorized into Geometry-based Algorithms/ Template-based Algorithms. The ones that can be made with the help of statistical tools like “SVM” [Support Vector Machines], “PCA” [Principal Component Analysis], “LDA” [Linear Discriminant Analysis], “Kernel methods” or “Trace Transforms” are Geometry-based or Template-based algorithms. It uses facial features analysis and relationship within their geometry.

### Piecemeal/Wholistic Facial recognition System

The connection within the elements or the connection of a function with the whole face not underwent into the amount. This very approach was followed by many researchers with the main aim of wanting to simplify the most desired features. Thus, some experimented with the use of eyes, a combination of various elements and so on. Some of the “Hidden Markov Model” approaches also come in this classification, and feature processing is quite popular in the system of face recognition.

## Appearance-Based / Model-Based Facial recognition System

The appearance-based method presents a particular face with respect to several images. An image is to be taken as a high dimensional vector. This method allows us to derive an element space from the image division. The sample image compared to the training set. Whereas, the model-based approach attempts at modelling a face. The new sample applied to the given model and the parameters of the model are helpful in recognising the image. This method is based on use of PCA and 2D/3D elements.

## Neural Networks Based Facial Recognition System

Neural Network is the one that has continued to implement pattern recognition and classification. Kohonen was the first present that a neuron network could also be targeted at recognising aligned and normalised faces. This involves performing feature extraction using neural networks. Various methods exist that combine tools such as PCA or LCA for making a high end classifier. Feed Forward Neural Network with additional bias, Self-Organizing Maps with PCA, and Convolutional Neural Networks with multi-layer perception, etc. are examples of such systems. This on the whole helps increase efficiency.

## Feature Extraction

The projected feature extraction is predicated on the extent of fogginess of the input image. In terms of the grey level, this implies that there are sturdy intensity variations between pixels if the input image is sharp. In distinction, there are weak intensity variations between pixels if the input image is already blurred.

## IMPLEMENTATION OF FACIAL RECOGNITION SYSTEM “

### Haar Cascade Classifiers using OpenCV

Cascade classifier, with other names such as cascade of boosted classifiers combinedly employing with haar-like characteristics, is a unique case of ensemble learning, called boosting. The training of cascade classifiers on around hundred test sample images of image constituting the object to be detected, and other images that do not involve those images. There is an algorithm to know if there is a face or not is called Viola-Jones object detection framework, which involves all the procedural steps needed for live face detection.



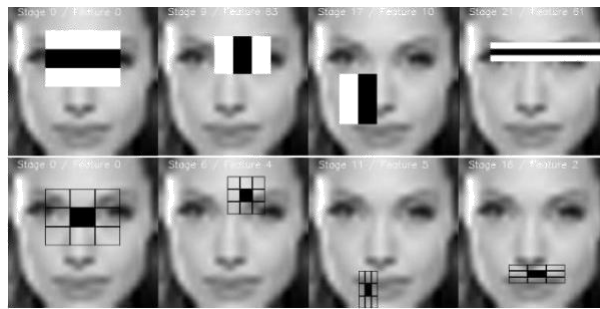


Fig1.6.Haar features

### Histogram of Oriented Gradients using Dlib

Second method that is quite popular and widely used is under Dlib and uses basic features called Histogram of Oriented Gradients (HOG). This is an application of the authentic paper by Dalal and Triggs.[14]. The main concept underlying HOG is extraction of elements into a vector, and giving it into an algorithm classification like a Support Vector Machine such as the one that will assess if a face (or any other entity trains it to identify originally) is present within a region or not. The elements that are derived using the technique are the distribution (histograms) of directions of gradients (oriented gradients) of the image. Gradients help us to detect such regions as they are basically huge around edges.”

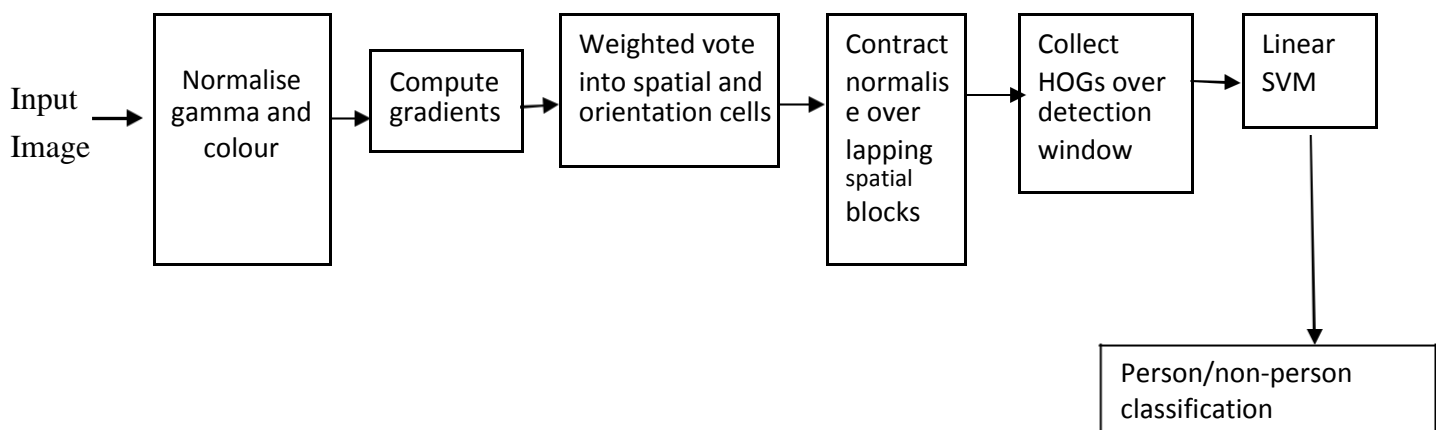


Fig1.7 Convolutional Neural Networks using Dlib/OpenCV

Convolutional Neural Network abbreviated and popularly known as CNN are feed-forward neural network. Their main purpose is in the field of computer vision. They can be used to pre-treat an automated image and also a dense neural network part. CNNs are specially designed types of neural networks that help process data having grid-like topology. Their outlay is based on the visual cortex of

animals. The very name of the CNNs resulted from the truth that we convolve the initial image input within a huge set of filters. The parameter to choose remains the number of filters to apply, and the dimension of the filters. The stride length is the dimension the particular filter. The value of strides ranges between 2 and 5, may vary depending on the model being integrated with.”

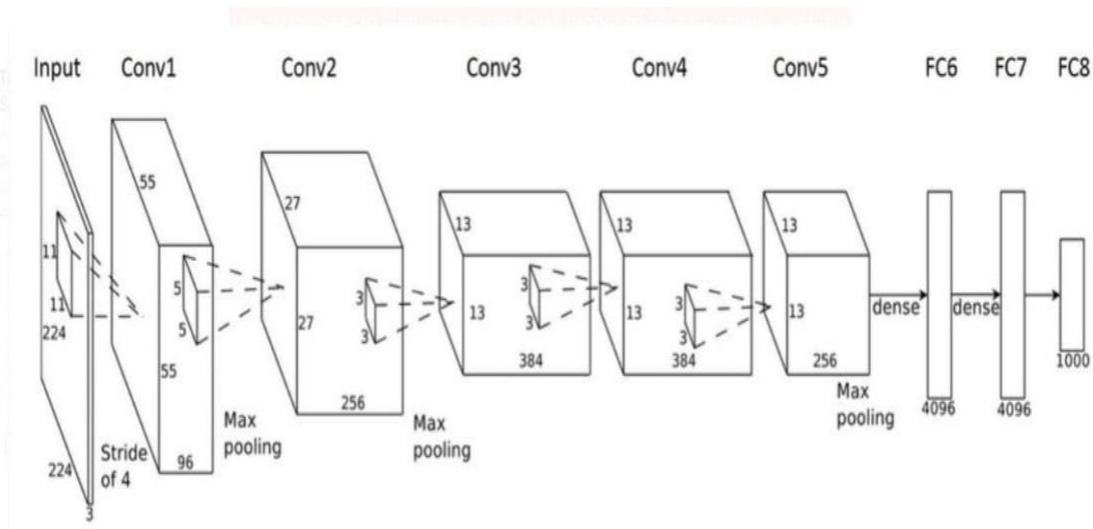


Fig1.8. Convolutional neural networks model

An approach neural networks based is chosen, focusing on convolution neural networks along with OpenCV in python.

Dataset containing 2 folders for fake and real images.:

gather\_examples.py: This is the script that extracts face ROIs from input video files for the creation a diversified dataset for face liveness detection model.

train\_liveness.py : It is very clear from the filename itself that it will train LivenessNet classifier. Keras and TensorFlow were used as basic tools to train the model. The training process results in a few files:

le.pickle: The class label encoder. (Label Encoding means to transform the labels into numeric form so that they become into the machine-readable format.)

liveness.model: Here the serialized Keras model come into play, which detects face liveness.

plot.png: The training history plot is a performance indicator showing accuracy and loss curves assisting in assessing the model (i.e. overfitting/underfitting).

liveness\_demo.py: This script for demonstration will fire up the webcam, extracting frames to apply face liveness detection in real-time.

liveness\_demo.py: This script for demonstration will fire up the webcam, extracting frames to apply face liveness detection in real-time.

## CHAPTER 2

### LITERATURE SURVEY

Amongst the common and popular technologies of biometrics such as handwriting verification, fingerprint sensing and scanning technologies, which have been growing and progressing recently, is the face recognition approach, which has gained popularity among the public as well as the IT hubs is because of its distinct and attractive features such as more directness, user friendliness and convenience. Therefore, it has been applied and used widely, in biometric authentication systems. But, in general, the face recognition algorithm is not able to separate the 'live' face from the fake face recognition system by facial pictures and portraits. To avoid such hacking, a safe system requires liveness detection. The significance of biometrics in today's society has been hugely reinforced by the requirement for recognition management systems for this is to avoid the perpetrators' fingerprint recognition and iris recognition vulnerabilities recently. But in face recognition, approaches to deal with this problem are very limited. The liveness model's work is to differentiate the feature space in alive and non-living.

On research it has been found that a lot of theoretical and some level of practical research has been done in integrating anti-spoofing techniques into the facial recognition system to make it more efficient. A lot of technologies have been applied behind this concept to get better and efficient results.

The very approach of distinguishing between users based on their property of whether living or not is known as liveness detection. Perpetrators can try to introduce a large number of hacked authentications into the systems. With the assistance of physiological property detection and verification, the accuracy of a biometric system can improve. It's a very important and difficult concern that liveness is especially supported by researching out patterns on facial features over a series of photos.

The model is based on finding a true user using approaches based on hardware training and combined with the softwares. The mostly utilized in characteristic of physiological property detection of the face space unit is 'Eye detection and alter detection of eyes.'

The Anti-spoof/anti-hacking downside had to be deciphered before face recognition system can be widely applied in our systems of authentication. A time period and non-intrusive methodologies assisted the diffusion velocity of one single image that is planned to tackle the problem of face spoofing and hacking of images or videos. Above all, the separation in surface properties between an alive face and a fake one is discovered efficiently and accurately within the diffusion speed. Anti spoofing/hacking options are exploited by utilizing the full variation flow theme. Additionally also, the process of the native patterns of the diffusion speed, the alleged "native speed patterns, because the options, that are input into the linear SVM classifier is planned to work out whether or not the given face is fake or not. One vital advantage of the planned approach is that, in contrast to previous approaches, it precisely identifies numerous malicious attacks" and hacks no matter the medium of the image, like, paper or screen. Although, the approach that has been proposed for use does not demand any particularities in user action. Results of the experiment on varied wide sets of data and information displays that the methodology that has been proposed is precisely effective for facial physiological property detection.

[1] A liveness detection model integrated with elements of optical flow field are being discussed. It uses a very basic and evident fact that optical illumination and reflection from a 2 dimensional object is very different from that happening from a 3 dimensional object. The paper had a proposal to use concepts of varied optical illumination in new liveness detection method for face recognition. The author assumed that test region was restricted a 2 dimensional plane.

But what is very important and the very basis of this theory is the accurate calculation of optical illumination by the device for best results, which is a task in itself. Also varied illumination will affect the calculation of optical illumination further affecting the results. Thus, further advancement in technology and research is required to solve the problem.

The Face Recognition Vendor Tests (FRVT) were started by the National Institute of Standards and Technology (NIST) in the early 2000's. FRVT was a further advancement of the Facial recognition Technology (FERET). The major aim of FRVTs was to be able to provide relevant information to designated authorities such as law and order enforcement agencies for the application of facial recognition systems. This was done through repeated evaluations to make the system commercially viable and secure.

[2] viewed research on the method of a linear fusion combination between static and video analysis. The motion and vitality were to be detected by the video analysis frame by frame with precise and efficient algorithms. The task of finding frame by frame clues was of static analysis. The dynamic score matching technology is exploited with visual elements and SVM classifiers, by the static analysis. The results were quite promising as they showed that this method easily helped detect spoofing attacks by static analysis. Also the proposed static analysis can do its job frame by frame, which is a distinguishing feature and provides a clear edge and advantage over other approaches which compulsorily have to process a portion of the video for clues of visual degradation. But as it is evident that vitality knowledge cannot be skipped and ignored especially by a biometric system. Thus, the combination of static and video analysis is helpful for better results and easier classification.

[3] proposed the use of micro-texture analysis. The methodology is helpful in detecting whether the person sitting in front of the system is living or dead, and is also used to print faces by the help of binary patterns. The observations, as can be seen were quite promising but are valid only for 2 dimensional set ups as they have not been tried and tested on 3 dimensional set ups. There is a tendency to note that given the three varied/distinguished categories of faking attacks mentioned earlier, they can be simply counter secured by detection of the 3 dimensional structure of the face. Video-based spoofing is harder to decipher as a result of facial videos of the target user might be more durable to return by; furthermore, such hacks may be with success conquered. Designing and a 3D mask is arguably a lot of long and conjointly requires specialised instrumentality. However, attributable to the threat this attack vector poses, abundant analysis has gone into detection the textures of 3D masks

Face recognition is one amongst the foremost common biometric ways to spot or to verify people as a result of its noninvasive property. Thanks to the advent of the advanced image sensors and therefore the subtle image process techniques, acquiring facial info becomes straightforward and this approach

. can eliminate the danger of forgetting login key and secret information for a individual account. For these reasons,using facial info to access security systems becomes popular, and this approach will build the systems a lot of convenient and reliable.

In spite of the fact that face acknowledgment frameworks are wide utilized instability frameworks like shrewd passage monitor frameworks at organizations and universities, bank client account logins, and site account login forms, and so forth., they're as yet subject to shifted purposeful assaults. For the framework, it's irksome to see whether the face before of the camera could be a bonafide face or a caricaturing assault. In apply, even some basic ways will pass the security framework. There are four basic assortments of level face mocking assaults: (I) utilizing a printed symbol, (ii) showing a photo by exploitation top notch (HD) screen, (iii) slanted print assaults, partner degreed (iv) showing a video exploitation a HD screen. A harder type of assault, that will be that the three-dimensional (3-D) facial veil assault, has been raised as of late. of these assaults will give facial data from that the framework will get a sound acknowledgment result. Therefore, a specialized methodology for the protection against satirizing assaults is basic for a face acknowledgment security framework. aliveness discovery intends to separate the testing face of a truthful individual from a ridiculing assault.Past analysis during this space will be classified into approaches—the systems with further devices, 2-D data ways, and 3D data ways. further device approaches are utilized in industries, that use infrared sensors or an additional camera.

[4] stated the use of live focus, a built-in camera function for the liveness detection.The problem with live focus is that it needs a minimum distance of focus,which is not feasible and applicable to all situations

[5]proposed the use of frequency and texture analysis for livemess detection.For better results the technique can be integrated with micro-texture analysis and a system using combination of static and video analysis.The exploitation of texture and frequency comes to use in the extraction of distinguishing shapes,features or elements and also detailed review of faces and features in real and fake images. The results here again,were satisfactory for 2 dimensional implementations but with 3 dimensional implementations the results were not upto the mark.

[6] proposed for a combination of various number of photos-attacks for derection in facial recognition systems.There were expermintals with both static and videoi analysis for better and efficient results ,and also helping gather extra information about texture,movement and liveness,which further helped obtain better and error free dingushing of models.The prototype provided excellent results with respect to performance when SVM classifiers with varied visual elements and features were exploited through a dynamic technique.

First, Gaussian sifting to the face picture is done all together that the pressed 3D bend is acquired. Inside the bend, we tend to separate all the local essentials exploitation the strategy of the inclination plummet to downsize the invalid eye up-and-comers, the consideration classifier, that is prepared by Viloa'sAdaBoost instructing methodologies, is utilized. From that point forward, face locale is being standardized by a couple of size and revolution by exploitation focus motivation behind eyes because of the information face will change in size and direction

Preparing is a significant piece of any model dependent on man-made reasoning, and an indivisible part of machine learning.Facial acknowledgment framework snaps a photo and changes over it into a grayscale picture before any handling. So,we picked a dataset with pictures previously changed over to grayscale for simpler and quicker preparing of the model.

Biometric frameworks are not 100% exact. Biometric frameworks exactness during the format examination procedure of validation relies upon outside factors, in particular, temperature, preparing level of the

enlistment procedure specialists, physical state of the person to be confirmed, and so on. Biometric frameworks precision is likewise reliant on inner factors, for example, nature of the gear and the restrictive calculations being utilized. Most biometric frameworks get their major exactness from the accompanying parameters:

- Bogus Match Rate (FMR): Is the likelihood that a faker will be acknowledged as a certifiable client by inaccurately passing judgment on a match in their enlistment layout

- False Non-Match Rate (FNMR): Is the likelihood that a veritable client will be dismissed by inaccurately passing judgment on a confound in their enlistment layout

- Failure To Enroll (FTE): Is the likelihood that a given client will be not able try out a biometric framework FMR and FNMR are needy factors and their relationship to each other can be portrayed by the Receiving Operating Characteristic Curve (ROC)

[8] brought to light the proposal of combination of convolutional neural networks and light field camera data helping the detection procedure. A single shot depth detection of any object is provided by the LFC. This serves as a basic feature to help fight various spoofing attacks and techniques, by helping distinguish a real face from a fake one.

The strategy bolstered optical stream field was presented by Bao et al. It examines the varieties and properties of optical stream created from 3D objects and second planes. The movement of optical stream field could be a blend of four essential development types: interpretation, revolution, moving and swing. The creators found that the essential three fundamental assortments are producing very comparable optical stream fields for each second and for 3D pictures. The fourth kind makes the specific varieties in optical stream field. Their methodology is to a great extent bolstered the idea that the optical stream field for second items will be depicted as a projection change. The optical stream licenses to reason the reference field, Gregorian schedule month 2014 sixteen so allows to work out whether the investigate district is levelled or is it not levelled. The trial was tested on 3 groups of test data. the essential group contained a hundred composed face photographs that were interpreted and randomly turned, the subsequent bunch contains a hundred photographs from bunch one that could collapse and be twisted prior to the investigation, and the bunch three was made of physical features of authentic people (twenty users, each used various times) and actions such as swing of face, shake to the left and right, and so forth. The creators directed the analysis for ten seconds.

The detailed analysis of various papers brings to notice the fact that a number of anti-spoofing techniques have been proposed to fight spoof attacks, constituting various concepts on texture analysing, contextual information approaches, life signs identification, motion analysis, colour analysis, or deep learning-based concepts [9],[12].

[7] We see the implementation of various anti spoof concepts by reverse decomposition of a real face into a fake one and analysis of a pattern of spoof noise. [13] attempted at detecting and cancelling spoof attacks by using the texture analysis method which was based on a different concept of Local Binary Pattern (LBP). This technique could be used on various types of attacks involving use of printed copies of the face, and videos displayed on devices of multiple dimensions. Incorporating a 3 dimensional camera into a spoof detection technique is known as hardware method.

[10] proposed the use of depths maps to help differentiate a 2 dimensional fake face from a 3 dimensional real one. The approaches involving movements of various parts has also been studied and observed carefully but the results were not quite satisfactory. [9] This holds true for eye blinking detection as a technique to counter spoof attacks as in real life situation it is not feasible because it requires a lot of processing and a totally different dataset and also can be spoofed with ease. Research work has taken

into account a lot of different types of techniques and concepts constituting detection of motion, color, texture, image distortion for [11] liveness detection [13].

A forensic dataset was being made by the office of Sheriff in Pinellas County which helped designated persons to tap into the photo archives of the state's Department of Highway Safety and Motor Vehicles (DHSMV), in 2009. By 2011, around 170 authorized personnel had been outfitted with cameras that allowed clicking pictures of suspects that could be cross-checked and verified against the existing database. This affected the policing by helping in more arrests and criminal investigations that was not seen before without the technology,

The procedure of Component-based face coding approach for liveness acknowledgment was used by Jianwei Yang et al. The makers have proposed a system which contains four phases: (1) finding the pieces of face; (2) coding the low-level features exclusively for all of the sections; (3) surmising the critical level face depiction by pooling the codes with loads got from Fisher measure; (4) interfacing the histograms from all fragments into a classifier for unmistakable confirmation.

Two-dimensional (2-D) image based ways have comparatively low machine price and can be embedded in moveable devices. The 2-D image based ways will be any separated into three main categories supported the kinds of liveness indicator they used: motion analysis, texture analysis, and life sign detection. Because the definition and color vary of screens becomes higher and wider, it's harder to sight the feel distinction between a true object and a screen image. Additionally, for the motion analysis and life sign detection ways, the system cannot perform well once the spoofing attack is a video sequence.

The presentation of this sort of technique is incredibly connected with the hole between the camera and testing face. When the hole goes to be gigantic, the presentation corrupted harshly. As of late, light-field cameras (LFC) are created. Exploitation light-field strategy is extremely encouraging to determine face liveness recognition issues and numerous elective issues that require 3D information, such as separation expectation, building demonstrating, object modeling, etc. We will present the attempts that are bolstered the light-field imaging. LFC will give profundity information. We will in general recommend that this additional information will be very much utilized by the convolutional neural systems (CNNs) to ask higher order results for level face ridiculing assault. Owing to this explanation, we endeavor to bring CNNs into light-field-based liveness discovery setting. CNNs have as of late got a great achievement in picture or video characterization, acknowledgment, and recovery since 2012.13–17 At that year, Krizhevsky et al. begun Associate in Nursing in AI and CNNs, by enacting a major advancement of the picture acknowledgment exactness exploitation CNNs. Almost each year when 2012, the score of ImageNet,18 which has become the quality benchmark for large-scale object recognition within the past seven years, has been improved a lot by CNNs-based technique. Liveness detection can also be thought-about as a picture classification drawback.

However, the difficulty of liveness detection is that each one categories solely have a small distinction, and for the light-field knowledge, the data structure isn't appropriate for the CNNs model.



[9]This holds true for eye blinking detection as a technique to counter spoof attacks as in real life situation it is not feasible because it requires alot of processing and a totally different dataset and also can be spoofed with ease.Research work has taken into account alot of different types of techniques and concepts constituting detection of motion,color ,texture ,image distortion for [11]liveness detection[13]. The detailed analysis of various papers brings to notice the fact that a number of anti-spoofing techniques have been proposed to fight spoof attacks, constituting various concepts on texture analysing, contextual information approaches, life signs identification, motion analysis, colour analysis, or deep learning-based concepts [9],[12].

[7]We see the implementation of various anti spoof concepts by reverse decomposition of a real face into a fake one and analysis of a pattern of spoof noise.[13] attempted at detecting and cancelling spoof attacks by using the texture analysis method which was based on a different concept of Local Binary Pattern(LBP).This technique could be used on various types of attacks involving use of printed copies of the face,and videos displayed on devices of multiple dimensions. Incorporating a 3 dimensional camera into a spoof detection technique is known as hardware method

The detailed analysis of various papers brings to notice the fact that a number of anti-spoofing techniques have been proposed to fight spoof attacks, constituting various concepts on texture analysing, contextual information approaches, life signs identification, motion analysis, colour analysis, or deep learning-based concepts [9],[12].

[7]We see the implementation of various anti spoof concepts by reverse decomposition of a real face into a fake one and analysis of a pattern of spoof noise.[13] attempted at detecting and cancelling spoof attacks by using the texture analysis method which was based on a different concept of Local Binary Pattern(LBP).This technique could be used on various types of attacks involving use of printed copies of the face,and videos displayed on devices of multiple dimensions. Incorporating a 3 dimensional camera into a spoof detection technique is known as hardware method

Table 2.1. Comparative analysis of non-intrusive detection techniques

Liveness Indicators	Cost and Methods	Advantages	Disadvantages
<b>1.Texture</b>	Low cost  Non-intrusive method	1.Simple implementation.  2.No user collaboration needed.	1.Low image or video quality. 2.Low textual attacks. 3.Need diverse datasets.
<b>2.Motion</b>	Medium cost  Intrusive method	1.Texture independent.  2.Hard to spoof. 3. No user collaboration needed.	1.Needs high quality data. 2.Needs video. 3.Difficult to use when low motion information. 4.Illumination problem.
<b>3.Life Signs</b>	High cost  Both intrusive(e.g some motion activity on face) & non- intrusive(e.g eye blinking)	1.Texture independent.  2.Handle all attacks. 3.Good performance under all illumination conditions.	1.Needs extra hardware or 2.Sensor needs videos and may need user collaboration.

## **CHAPTER 3**

### **SYSTEM DEVELOPMENT**

Systems of facial recognition have become quite popular over the last few years, causing them to be included in our daily lives as an important inseparable digital enhancement. The existing examples involve mobile phones, building entries, security systems, crypts and locking systems using facial recognition for authentication, widely. The benefits of the technology are also visible in industrial sector. The globally spread mobile industry is using facial recognition as an authentication system to provide an extra layer of biometric security. It is being used to make communities more secure by law enforcement agencies. Travellers' safety and comfort have benefitted at airports. The technology has also helped keep an eye on crimes and violence.

#### **TYPES OF FACIAL RECOGNITION SYSTEMS**

##### **Geometric Based / Template Based**

Face recognition algorithms are classified into geometry based or template based algorithms. The ones that can be made with the help of statistical tools like SVM [Support Vector Machines], PCA [Principal Component Analysis], LDA [Linear Discriminant Analysis], Kernel methods or Trace Transforms are geometry or template-based algorithms. It uses facial features analysis and relationship within their geometry.

##### **Piecemeal/Wholistic**

The connection within the elements or the connection of a function with the whole face not underwent into the amount. This very approach was followed by many researchers with the main aim of wanting to simplify the most desired features. Thus, some experimented with the use of eyes, a combination of various elements and so on. Some of the Hidden Markov Model approaches also come in this classification, and feature processing is quite popular in the system of face recognition.

##### **Appearance-Based / Model-Based” “**

The appearance-based method presents a particular face with respect to several images. An image is to be taken as a high dimensional vector. This method allows us to derive an element space from the image division. The sample image compared to the training set. Whereas, the model-based approach attempts at modelling a face. The new sample applied to the given model and the parameters of the model are helpful in recognising the image. This method is based on use of PCA and 2D/3D elements.”

##### **Template / Statistical / Neural Networks Based**

## Template matching

Template matching method presents the sequential elements in the form of samples, models, pixels, textures, etc. Correlation or the distance measure method is used for the recognition function. Statistical Approach:-

The Statistical approach method uses the patterns to demonstrate the features. The recognition function is considered as a discriminant function. Image representation is done regarding the features. Thus, the purpose is to pick and implement the correct tools in statistics for the extraction and analysis.

## Neural Networks:-

Neural Network is the one that has continued to implement pattern recognition and classification. Kohonen was the first present that a neuron network could also be targeted at recognising aligned and normalised faces. This involves performing feature extraction using neural networks. Various methods exist that combine tools such as PCA or LCA for making a high end classifier. Feed Forward Neural Network with additional bias, Self-Organizing Maps with PCA, and Convolutional Neural Networks with multi-layer perception, etc. are examples of such systems. This on the whole helps increase efficiency.

## SOME CONSTRAINTS FACED

There always exist some restraints and barriers in the designing of model. Presented below are some of such constraints faced in the development on the "face liveness detection using python" model.

Dataset: "The problem associated with dataset is that the model requires a very specific and typical type and the ones readily available in the market are all mostly based on Caucasian or Southeast Asian.

Thus, for testing the model on people with Indian features data set requires to have such features for proper training and results. Also, if model is not provided a huge, diverse database, the system may not work properly and efficiently

Specific orientation: The data sets also contain videos of some individual faces for better training of the model. However, the specificity of the trained model doesn't pick up people of different regions with characteristically different features. Limiting its overall performance.

Additional image/face sources: "The time constraint doesn't allow one to integrate high quality printed pictures to be differentiated against as fake ones from the real ones. Thus, the model will always require better advances and technological integration.

Pose variations: Changing one's alignment with respect to the camera or making different kinds of faces by yawning or covering some part of the face make it difficult to match the features in the existing set to features on which the model is trained, as doing so changes the face features to a large extent.

Varying illumination conditions: "Sitting in varied types of lighting systems directly affect the model by bringing variations of illuminations. Just as in low levels of illumination it becomes very difficult to find coordinates of features, same is true for an extremely illuminated scene. Just as in a completely dark room, there will be no illuminations so there will no face detection leaving feature recognition far behind. Thus, illumination can affect the accuracy of the model. Also, some lighting patterns affect facial features to be appearing different what they originally are, and again affecting the performance of the model.

## PYTHON

Python is one among those rare languages which may claim to be each easy and powerful. you'll end up pleasantly shocked to ascertain however simple it's to think about the answer to the matter instead of the syntax and structure of the language you're programming in. The official introduction to Python is: Python is a simple to be told, powerful programming language. it's economical high-level information structures and a straightforward however effective approach to object-oriented programming. Python's elegant syntax and dynamic typing, in conjunction with its taken nature, build it a perfect language for scripting and fast application development in several areas on most platforms.

Guido van Rossum, the creator of the Python language, named the language when the BBC show "Monty Python's Flying Circus". He does not significantly like snakes that kill animals for food by winding their long bodies around them and crushing them.

Features of Python:

### Simple

Python may be a easy and minimalistic language. Reading a decent Python program feels virtually like reading English, though terribly strict English! This pseudo-code nature of Python is one among its greatest strengths. It permits you to think about the answer to the matter instead of the language itself.

### Easy to be told

As you'll see, Python is very simple to urge started with. Python has a very easy syntax, as already mentioned.

Free and Open supply. Python is collaborative in Nursing case of a "FLOSS" (Free/Libre and Open supply Software). In simple language, anyone can openly distribute versions of the code, study its ASCII file, construct changes to that, and can use the features in their programs. "FLOSS" is considered on the conceiving idea of a community that exchanges information. This helps us understand the intelligence of Python – it is been made and is constantly enhanced by a community who simply needs to discover the far better Python.

## High-level Language

Upon writing the programs in Python, we never got to trouble regarding the low-level details like the management of the memory employed by the program, etc.

## Portability

As Python files are ASCII coded, they can be easily portable (i.e., modification for creating it to work on) to various different platforms. The Python programs will be working easily on any of the platforms while not requiring any changes, just noting to be careful enough to avoid any of the system-dependent choices. Python can be used on "GNU/Linux", "Windows", "FreeBSD", "Macintosh", "Solaris", "OS/2", "Amiga", "AROS", "AS/400", "BeOS", "OS/390", "z/OS", "Palm OS", "QNX", "VMS", "Psion", "Fruit Computer Architecture OS", "VxWorks", "PlayStation", "Sharp Zaurus", "Windows Cerium" and "PocketPC". For creation of games, platform like "Kivy" for the personal computer, iPhone, iPad can also be used.

## Interpreted

This requires a touch of rationalization. A program written in a very compiled language like C or C++ is regenerate from the language i.e. C or C++ into a language that's spoken by your pc (binary code i.e. 0s and 1s) employing a compiler with numerous flags and choices. Once you run the program, the linker/loader software package copies the program from magnetic disc to memory and starts running it. Python, on the opposite hand, doesn't would like compilation to binary. You simply run the program directly from the ASCII text file. Internally, Python converts the ASCII text file into associate intermediate type referred to as byte codes and so interprets this into the language of your pc and so runs it. All this, actually, makes mistreatment Python abundant easier since you do not got to worry regarding collection the program, ensuring that the right libraries area unit connected and loaded, etc. This additionally makes your Python programs way more moveable, since you'll be able to simply copy your Python program onto another pc and it simply works!

## Object directed

Python supports procedure-oriented programming moreover as object-oriented programming. In procedure-oriented languages, the program is constructed around procedures or functions that area unit nothing however reusable items of programs. In object-oriented languages, the program is constructed around objects that mix knowledge and practicality.

Python includes a terribly powerful however oversimplified approach of doing OOP, particularly when put next to huge languages like C++ or Java.

## Extensible

If you wish a crucial piece of code to run in no time or need to possess some piece of formula to not be open, you'll be able to code that a part of your program in C or C++ and so use it from your Python program.

## Embeddable

You can plant Python among your C/C++ programs to grant scripting capabilities for your program's users.

Extensive Libraries The Python normal Library is large so. It will assist you do numerous things involving regular expressions, documentation generation, unit testing, threading, databases, internet browsers, CGI, FTP, email, XML, XML-RPC, HTML, WAV files, cryptography, graphical user interface (graphical user interfaces), and alternative system-dependent stuff. Remember, all this can be invariably offered where Python is put in. this can be referred to as the Batteries enclosed philosophy of Python.

## OPENCV

OpenCV was started at Intel in 1999 by metropolis Bradsky and also the initial unharness came move into 2000. Vadim Pisarevsky joined metropolis Bradsky to manage Intel's Russian package OpenCV team. In 2005, OpenCV was used on Stanley, the vehicle UN agency won 2005 office Grand Challenge. Later its active development continued underneath the support of Willow Garage, with metropolis Bradsky and Vadim Pisarevsky leading the project. Right now, OpenCV supports plenty of algorithms associated with pc Vision and Machine Learning and it's increasing day-by-day. Currently OpenCV supports a good type of programming languages like C++, Python, Java etc and is offered on completely different platforms as well as Windows, Linux, OS X, Android, iOS etc. Also, interfaces supported CUDA and OpenCL also are underneath active development for high-speed GPU operations. OpenCV-Python is that the Python API of OpenCV. It combines the most effective qualities of OpenCV C++ API and Python language. Python may be a general purpose artificial language started by Guido van Rossum, that became very hip briefly time chiefly due to its simplicity and code readability. It allows the computer programmer to specific his ideas in fewer lines of code while not reducing any readability.

Compared to alternative languages like C/C++, Python is slower. however another vital feature of Python is that it are often simply extended with C/C++. This feature helps North American nation to jot down computationally intensive codes in C/C++ and build a Python wrapper for it in order that we will use these wrappers as Python modules. this provides North American nation 2 advantages: initial, our code is as quick as original C/C++ code (since it's the particular C++ code operating in background) and second, it's terribly straightforward to code in Python. This can be however OpenCV-Python works, it's a Python wrapper around original C++ implementation. And the support of Numpy makes the task additional easier. Numpy may be a extremely optimized library for numerical operations.

It provides a MATLAB-style syntax. All the OpenCV array structures area unit regenerate to-and-from Numpy arrays. therefore no matter operations you'll be able to waste Numpy, you'll be able to mix it with OpenCV, that will increase range of weapons in your arsenal. Besides that, many alternative libraries like SciPy, Matplotlib that supports Numpy are often used with this. So OpenCV-Python is Associate in Nursing acceptable tool for quick prototyping of pc vision issues.

## KERAS

Keras is Associate in Nursing Open supply Neural Network library written in Python that runs on high of Theano or Tensorflow. it's designed to be standard, quick and straightforward to use. it had

been developed by François Chollet, a Google engineer. Keras does not handle low-level computation. Instead, it uses another library to try to do it, known as the "Backend." therefore Keras is high-level API wrapper for the low-level API, capable of running on high of TensorFlow, CNTK, or Theano. Keras High-Level API handles the means we have a tendency to create models, shaping layers, or discovered multiple input-output models. during this level, Keras additionally compiles our model with loss and optimizer functions, coaching method with work perform.

Keras does not handle Low-Level API like creating the procedure graph, creating tensors or alternative variables as a result of it's been handled by the "backend" engine.

## TENSORFLOW

Whether knowledgeable or a beginner, TensorFlow is an end-to-end platform that produces it straightforward for you to make and deploy cc models. Watch the video Case studies An entire scheme to assist you solve difficult, real-world issues with machine learning Easy model building TensorFlow offers multiple levels of abstraction thus you'll be able to opt for the correct one for your wants. Build and train models by mistreatment the high-level Keras API, that makes obtaining started with TensorFlow and machine learning straightforward. If you wish a lot of flexibility, eager execution permits for immediate iteration and intuitive debugging. for big cc coaching tasks, use the Distribution Strategy API for distributed coaching on completely different hardware configurations while not dynamic the model definition. TensorFlow has forever provided a right away path to production. whether or not it's on servers, edge devices, or the web, TensorFlow helps to train and deploy your model simply, notwithstanding what language or platform you employ. Use TensorFlow Extended (TFX) if you wish a full production cc pipeline. For running logical thinking on mobile and edge devices, use TensorFlow fat-free. Train and deploy models in JavaScript environments mistreatment TensorFlow.js.

Powerful experimentation for analysis build and train progressive models while not sacrificing speed or performance.

TensorFlow provides you the flexibleness and management with options just like the Keras practical API and Model Subclassing API for creation of advanced topologies. For simple prototyping and quick debugging, use eager execution.

TensorFlow additionally supports AN scheme of powerful add-on libraries and models to experiment with, together with Ragged Tensors, TensorFlow likelihood, Tensor2Tensor and BERT.



Table 3.1. Keras Vs TensorFlow

Parameters	Keras	Tensorflow
Type	High-level API wrapper	Low -level
Complexity	Easy to use if used in Python	One has to understand the basic syntax and functions
Purpose	Rapid deployment for making model with standard layers	Allows to make arbitrary computational graphs or model layers
Tools	Uses other API debug tools such as TFDBG	Tensorboard visualization tools can be used
Community	Large active communities	Large active communities and widely shared resources

## SCIPY

SciPy (articulated/'saɪpər/'"Moan Pie"[3]) is a free and open-source Python library utilized for logical processing and specialized figuring. SciPy contains modules for enhancement, direct polynomial math, joining, insertion, extraordinary capacities, FFT, flag and picture handling, ODE solvers and different assignments basic in science and building. SciPy expands on the NumPy cluster object and is a piece of the NumPy stack which incorporates devices like Matplotlib, pandas and SymPy, and an extending set of logical figuring libraries. This NumPy stack has comparable clients to different applications, for example,

## SCIKIT

Scikit-learn is a free AI library for Python. It highlights different calculations like help vector machine, arbitrary backwoods, and k-neighbors, and it additionally underpins Python numerical and logical libraries like NumPy and SciPy. In this instructional exercise we will figure out how to code python and apply Machine Learning with the assistance of the scikit-learn library, which was made to make doing AI in Python simpler and progressively powerful. To do this, we'll be utilizing the Sales\_Win\_Loss

informational collection from IBM's Watson storehouse. We will import the informational index utilizing pandas, investigate the information utilizing pandas strategies like head(), tail(), dtypes(), and afterward take a stab at utilizing plotting procedures from Seaborn to imagine our information. At that point we'll plunge into scikit-learn and use preprocessing.LabelEncoder() in scikit-learn to figure out how to process the information, and train\_test\_split() to part the informational index into test and train tests. We will likewise utilize a cheat sheet to assist us with choosing which calculations to use for the informational index. At last we will utilize three unique calculations (Naive-Bayes, LinearSVC, K-Neighbors Classifier) to make expectations and think about their exhibition utilizing strategies like accuracy\_score() gave by the scikit-learn library. We will likewise imagine the exhibition score of various models utilizing scikit-learn and Yellowbrick perception.

## DLIB

Dlib is a broadly useful cross-stage programming library written in the programming language C++. Its plan is intensely impacted by thoughts from configuration by agreement and part based programming building. Along these lines it is, as a matter of first importance, a lot of autonomous programming parts. It is open-source programming discharged under a Boost Software License. Since improvement started in 2002, Dlib has developed to incorporate a wide assortment of apparatuses. Starting at 2016, it contains programming parts for managing organizing, strings, graphical UIs, information structures, direct variable based math, AI, picture preparing, information mining, XML and content parsing, numerical improvement, Bayesian systems, and numerous different errands. As of late, a significant part of the advancement has been centered around making a wide arrangement of measurable AI apparatuses and in 2009 Dlib was distributed in the Journal of Machine Learning Research. Since then it has been utilized in a wide scope of domains.

## NUMPY

NumPy, full form is for Numerical Python, focuses on the CPython reference execution of Python, which is a non-improving bytecode mediator. Scientific calculations composed for this form of Python regularly run much more slow than assembled reciprocals. NumPy addresses the gradualness issue halfway by giving multidimensional exhibits and capacities and administrators that work productively on clusters, requiring revising some code, generally inward circles utilizing NumPy.

Utilizing NumPy in Python gives usefulness practically identical to MATLAB since they are both interpreted, and the two of them enable the client to compose quick projects as long as most activities deal with clusters or networks rather than scalars. In examination, MATLAB brags an enormous number extra tool kits, remarkably Simulink, while NumPy is characteristically incorporated with Python, a progressively present day and complete programming language. In addition, correlative Python bundles are accessible; SciPy is a library that includes more MATLAB-like usefulness and Matplotlib is a plotting bundle that gives MATLAB-like plotting usefulness. Inside, both MATLAB and NumPy depend on BLAS and LAPACK for productive direct polynomial math calculations.

Python ties of the broadly utilized PC vision library OpenCV use NumPy clusters to store and work on information. Since pictures with different channels are just spoken to as three-dimensional exhibits, ordering, cutting or concealing with different clusters are proficient approaches to get to explicit pixels of a picture. The NumPy exhibit as all inclusive information structure in OpenCV for pictures, removed element focuses, channel pieces and a lot more unfathomably disentangles the programming work process and troubleshooting.

It contains numerous options together with these vital ones:

- (i) A powerful N-dimensional array object
- (ii) Sophisticated (broadcasting) functions
- (iii) Tools for group action C/C++ and algebraic language code
- (iv) Useful algebra, Fourier rework, and random range capabilities

## STRUCTURE OF THE MODEL

### 1. Dataset

fake dataset(constituting 800 entries)  
real dataset(constituting 960 entries)

### 2. face detector

deploy.protxt  
res10\_300x300\_ssd\_iter\_140000.caffemodel

### 3. image search

\_\_init .py  
livenessnet.py

4. Videos  
fake.mp4  
real.mp4
5. gather\_examples.py
6. train\_liveness.py
7. liveness\_demo.py
8. le.pickle
9. liveness.model
10. plot.png”

## GATHERING THE DATASET

“gather\_examples.py was put to use for this function. This script helps grabs face ROIs from input video files on the webcam ,further creating a deep learning face liveness dataset.

It does the following things:

- Taking a saved video file as input and taking out sample 160 images in the .png format and saving it directly to the dataset. It requires 2 commands, one taking fake image samples, while other taking real ones and then saving them in their respective places on the disk.”
- “Load serialized face detector from disk
- Loop over frames from the video file stream on the webcam
- Grab dimensions of frame and construct a blob from the frame,pass the blob through the network and obtain the detections. Assuming that each image has only one face, so finding the bounding box with the greatest probability.
- Compute (x, y)-coordinates of the detected bounding box for the face and extract the face ROI,further saving it on the disk.”

```
riya@riya:liveness-detection-opencv $ python3 gather_examples.py --input videos/real.nov --output dataset/real --detector face_detector --skip 1
[INFO] loading face detector...
[ INFO:0] Initialize OpenCL runtime...
[INFO] saved dataset/real/0.png to disk
[INFO] saved dataset/real/1.png to disk
[INFO] saved dataset/real/2.png to disk
[INFO] saved dataset/real/3.png to disk
[INFO] saved dataset/real/4.png to disk
[INFO] saved dataset/real/5.png to disk
[INFO] saved dataset/real/6.png to disk
[INFO] saved dataset/real/7.png to disk
[INFO] saved dataset/real/8.png to disk
[INFO] saved dataset/real/9.png to disk
[INFO] saved dataset/real/10.png to disk
[INFO] saved dataset/real/11.png to disk
[INFO] saved dataset/real/12.png to disk
[INFO] saved dataset/real/13.png to disk
[INFO] saved dataset/real/14.png to disk
[INFO] saved dataset/real/15.png to disk
[INFO] saved dataset/real/16.png to disk
[INFO] saved dataset/real/17.png to disk
[INFO] saved dataset/real/18.png to disk
[INFO] saved dataset/real/19.png to disk
[INFO] saved dataset/real/20.png to disk
```

Fig3.1. Running gather\_example.py script

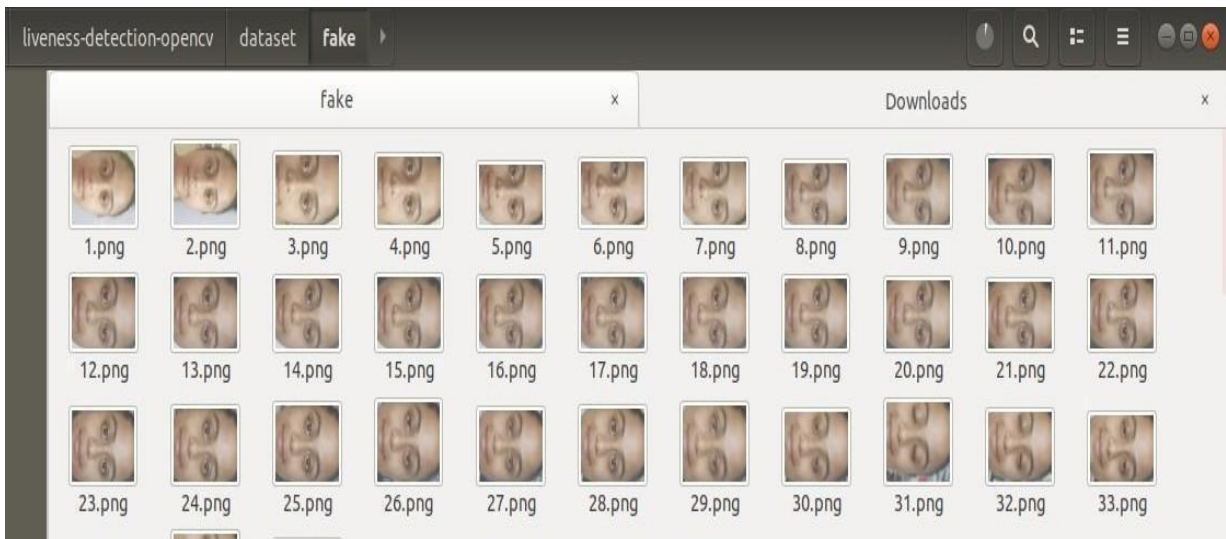


Fig3.2.Screenshot of fake images gathered by the script gather\_examples.py

## TRAINING THE MODEL

Preparing is a significant piece of any model dependent on man-made reasoning, and an indivisible part of machine learning. Facial acknowledgment framework snaps a photo and changes over it into a grayscale picture before any handling. So, we picked a dataset with pictures previously changed over to grayscale for simpler and quicker preparing of the model.

Biometric frameworks are not 100% exact. Biometric frameworks exactness during the format examination procedure of validation relies upon outside factors, in particular, temperature, preparing level of the enlistment procedure specialists, physical state of the person to be confirmed, and so on. Biometric frameworks precision is likewise reliant on inner factors, for example, nature of the gear and the restrictive calculations being utilized. Most biometric frameworks get their major exactness from the accompanying parameters 1 : -

- Bogus Match Rate (FMR): Is the likelihood that a faker will be acknowledged as a certifiable client by inaccurately passing judgment on a match in their enlistment layout
- False Non-Match Rate (FNMR): Is the likelihood that a veritable client will be dismissed by inaccurately passing judgment on a confound in their enlistment layout
- Failure To Enroll (FTE): Is the likelihood that a given client will be not able try out a biometric framework FMR and FNMR are needy factors and their relationship to each other can be portrayed by the Receiving Operating Characteristic Curve (ROC) appeared in given figure.

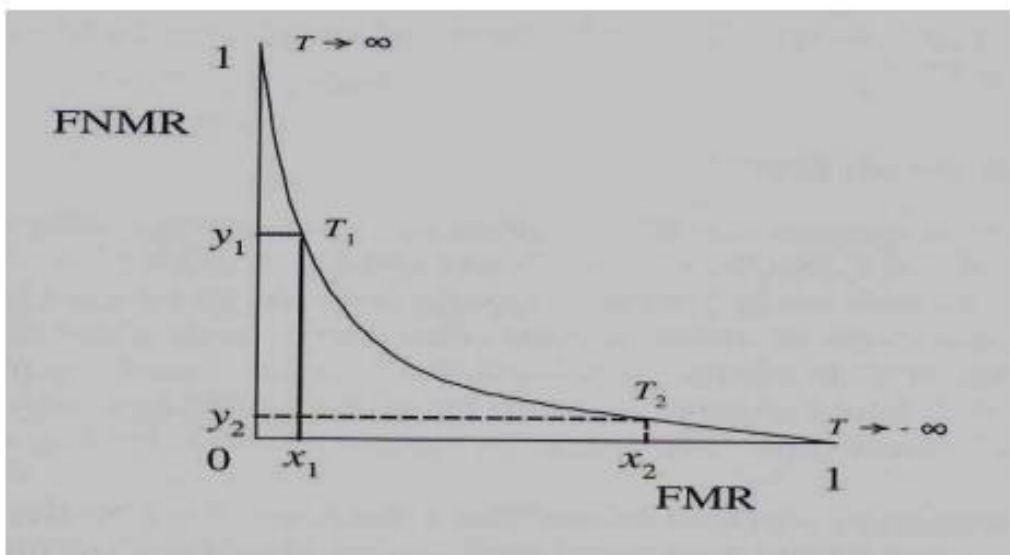


Fig 3.3 FNMR

train\_liveness.py was put to use for this very function. This script is to train the livenessNet classifier. 50 epochs have been used in training of the model. The LivenessNet class is defined in this file, consisting of one static method, and build. The build method accepts four parameters:

- width  
: How wide is the image/volume.
- height  
: How tall is the image .
- depth  
: The number of channels for the image (if 3 because working with RGB images).
- classes  
: The number of classes. There are two total classes: “real” and “fake”.

The script is adapted to serve the following functionalities:

- Set the matplotlib backend so as to save figures in the background.
- Initialize the initial learning rate, batch size, and number of epochs to be trained for.
- Grab the list of images in dataset directory, further initialize the list of data (i.e., images), class images.
- Extract the class label from the filename, load the image and resize it to be a fixed 32x32 pixels, ignoring aspect ratio and updates the data and labels lists, respectively.
- Conversion of the data into a NumPy array, then preprocess it by scaling all pixel intensities to the range [0, 1]
- The dataset is partitioned into training and testing sets which comprises the training dataset of 75% and a testing dataset of 25%
- Build training image generation for the augmentation of the data, also initializes the optimizer and model. Train, evaluates and save the network to disk and also save the label encoder to the disk.
- Also, plotting the training loss and accuracy.”

```

Terminal
File Edit View Search Terminal Help
welcome Riya
riya@riya:~ $ cd liveness-detection-opencv
riya@riya:liveness-detection-opencv $ python3 train_liveness.py --dataset dataset --model liveness.model --le le.pickle
Using TensorFlow backend.
/usr/local/lib/python3.6/dist-packages/tensorflow/python/framework/dtypes.py:516
: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated
; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)typ
e'.
_np_qint8 = np.dtype [("qint8", np.int8, 1)]
/usr/local/lib/python3.6/dist-packages/tensorflow/python/framework/dtypes.py:517
: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated
; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)typ
e'.
_np_quint8 = np.dtype [("quint8", np.uint8, 1)]
/usr/local/lib/python3.6/dist-packages/tensorflow/python/framework/dtypes.py:518
: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated
; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)typ
e'.
_np_qint16 = np.dtype [("qint16", np.int16, 1)]
/usr/local/lib/python3.6/dist-packages/tensorflow/python/framework/dtypes.py:519
: FutureWarning: Passing (type, 1) or '1type' as a synonym of type is deprecated
; in a future version of numpy, it will be understood as (type, (1,)) / '(1,)typ
e'.

```

Fig3.4. Running train\_liveness.py script

```

Terminal
File Edit View Search Terminal Help

[INFO] training network for 50 epochs...
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/tensorflow/python
/ops/math_grad.py:1250: add_dispatch_support.<locals>.wrapper (from tensorflow.p
ython.ops.array_ops) is deprecated and will be removed in a future version.
Instructions for updating:
Use tf.where in 2.0, which has the same broadcast rule as np.where
WARNING:tensorflow:From /usr/local/lib/python3.6/dist-packages/keras/backend/ten
sorflow_backend.py:422: The name tf.global_variables is deprecated. Please use t
f.compat.v1.global_variables instead.

Epoch 1/50
1/62 [.....] - ETA: 34s - loss: 0.6346 - accuracy: 0.6
5/62 [=>.....] - ETA: 7s - loss: 1.0523 - accuracy: 0.57
9/62 [====>.....] - ETA: 3s - loss: 1.2347 - accuracy: 0.47
13/62 [=====>.....] - ETA: 2s - loss: 1.2878 - accuracy: 0.49
17/62 [=====>.....] - ETA: 2s - loss: 1.3051 - accuracy: 0.50
21/62 [=====>.....] - ETA: 1s - loss: 1.2305 - accuracy: 0.55
25/62 [=====>.....] - ETA: 1s - loss: 1.2340 - accuracy: 0.55
29/62 [=====>.....] - ETA: 1s - loss: 1.1904 - accuracy: 0.57
33/62 [=====>.....] - ETA: 0s - loss: 1.1501 - accuracy: 0.59
37/62 [=====>.....] - ETA: 0s - loss: 1.1151 - accuracy: 0.60
41/62 [=====>.....] - ETA: 0s - loss: 1.0994 - accuracy: 0.61
45/62 [=====>.....] - ETA: 0s - loss: 1.0821 - accuracy: 0.61
49/62 [=====>.....] - ETA: 0s - loss: 1.0315 - accuracy: 0.63
53/62 [=====>.....] - ETA: 0s - loss: 1.0056 - accuracy: 0.63
57/62 [=====>.....] - ETA: 0s - loss: 0.9876 - accuracy: 0.64
61/62 [=====>.....] - ETA: 0s - loss: 0.9826 - accuracy: 0.64
62/62 [=====>.....] - 2s 25ms/step - loss: 0.9726 - accuracy:
0.6437 - val_loss: 0.6785 - val_accuracy: 0.7083
Epoch 2/50

```

Fig3.5 Screenshot depicting training of the network for 50 epochs



```

Terminal
File Edit View Search Terminal Help
Epoch 50/50
1/62 [.....] - ETA: 0s - loss: 0.0044 - accuracy: 1.00
5/62 [=>.....] - ETA: 0s - loss: 0.0264 - accuracy: 1.00
9/62 [====>.....] - ETA: 0s - loss: 0.1715 - accuracy: 0.94
13/62 [=====>.....] - ETA: 0s - loss: 0.1233 - accuracy: 0.96
17/62 [=====>.....] - ETA: 0s - loss: 0.1117 - accuracy: 0.96
21/62 [=====>.....] - ETA: 0s - loss: 0.1087 - accuracy: 0.95
24/62 [=====>.....] - ETA: 0s - loss: 0.1022 - accuracy: 0.96
27/62 [=====>.....] - ETA: 0s - loss: 0.0936 - accuracy: 0.96
31/62 [=====>.....] - ETA: 0s - loss: 0.1030 - accuracy: 0.96
35/62 [=====>.....] - ETA: 0s - loss: 0.0944 - accuracy: 0.97
39/62 [=====>.....] - ETA: 0s - loss: 0.0926 - accuracy: 0.97
43/62 [=====>.....] - ETA: 0s - loss: 0.0953 - accuracy: 0.97
47/62 [=====>.....] - ETA: 0s - loss: 0.1387 - accuracy: 0.95
50/62 [=====>.....] - ETA: 0s - loss: 0.1396 - accuracy: 0.95
52/62 [=====>.....] - ETA: 0s - loss: 0.1344 - accuracy: 0.95
54/62 [=====>.....] - ETA: 0s - loss: 0.1317 - accuracy: 0.96
58/62 [=====>.....] - ETA: 0s - loss: 0.1397 - accuracy: 0.95
62/62 [=====>.....] - 1s 19ms/step - loss: 0.1482 - accuracy:
0.9575 - val_loss: 0.0163 - val_accuracy: 0.9940
[INFO] evaluating network...
          precision    recall  f1-score   support

 fake         0.99         1.00         0.99         71
  real         1.00         0.99         0.99         97

 accuracy
macro avg         0.99         0.99         0.99         168
weighted avg         0.99         0.99         0.99         168

[INFO] serializing network to 'liveness.model'...

```

Fig3.6. Screenshot depicting statics about the trained network, an accuracy of 0.99

## RUNNING THE MODEL

“liveness\_demo.py is the main script that is used to run the model. Leading to the opening up of the webcam of the device and hence starting up the model. After initialization, grab and extract ROI and then classifying image/ person in front of it as real or fake. The model also tells about probability of realness or fakeness out of 1, upto three decimal points. To quit, “q” key has to be pressed on the keyboard.

The script is to do the following tasks:

- Loading of the serialized face detector from disk

- Loading of the liveness detector model and label encoder from disk

- Initialization of the video stream and allowing the camera sensor to warm up

The frame is being grabbed from a video stream that is threaded and the pixels are being resized by a maximum width that is 600 pixels. Frame dimensions are also grabbed and they are being converted to a blob. The blob is passed into a network to obtain the detected result and the predictions made.

Extraction of the confidence (i.e., probability) related with the probability and extract the face ROI and then pre process it in the exact same manner as was done with the training data. Then passing of the face ROI through the trained liveness detector model to deduce if the face is "real" or "fake". Drawing the label and bounding box on the frame is also to be done



```
Terminal
File Edit View Search Terminal Help
[INFO] starting video stream...
[ INFO:0] Initialize OpenCL runtime...
QObject::moveToThread: Current thread (0x9115230) is not the object's thread (0x
684d730).
Cannot move to target thread (0x9115230)

QObject::moveToThread: Current thread (0x9115230) is not the object's thread (0x
684d730).
Cannot move to target thread (0x9115230)

QObject::moveToThread: Current thread (0x9115230) is not the object's thread (0x
684d730).
Cannot move to target thread (0x9115230)

QObject::moveToThread: Current thread (0x9115230) is not the object's thread (0x
684d730).
Cannot move to target thread (0x9115230)

QObject::moveToThread: Current thread (0x9115230) is not the object's thread (0x
684d730).
Cannot move to target thread (0x9115230)

QObject::moveToThread: Current thread (0x9115230) is not the object's thread (0x
```

Fig3.8. Screenshot depicting starting of the video stream

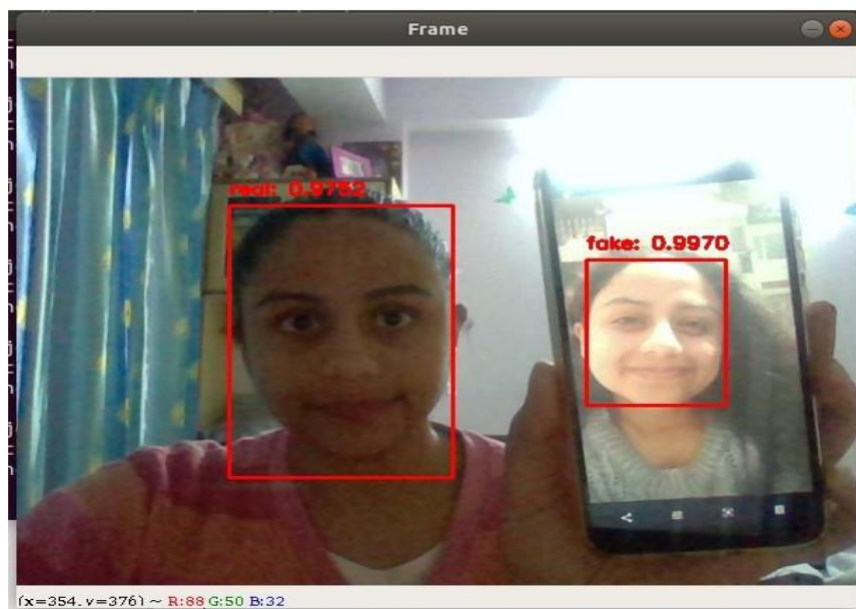
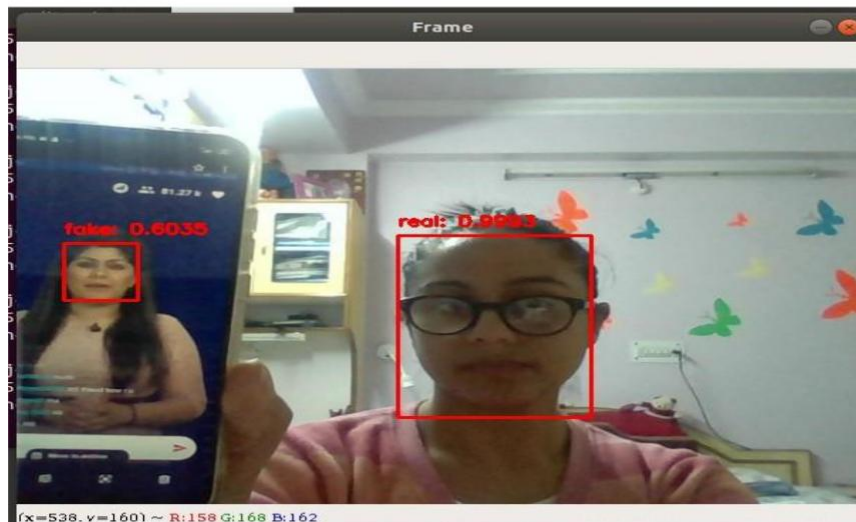


Fig3.9. Screenshots of frame window of the model depicting various real and fake faces along with their individual probabilities

- a) real: 0.9998 and fake: 0.9624
- b) real: 0.9993 and fake 0.6244
- c) real: 0.9752 and fake: 0.9970

## CHAPTER 4 PERFORMANCE ANALYSIS

### IMPLEMENTATION OF FACIAL RECOGNITION SYSTEM “

#### Haar Cascade Classifiers using OpenCV

Cascade classifier, with other names such as cascade of boosted classifiers combinedly employing with haar-like characteristics, is a unique case of ensemble learning, called boosting. The training of cascade classifiers on around hundred test sample images of image constituting the object to be detected, and other images that do not involve those images. There is an algorithm to know if there is a face or not is called Viola-Jones object detection framework, which involves all the procedural steps needed for live face detection.

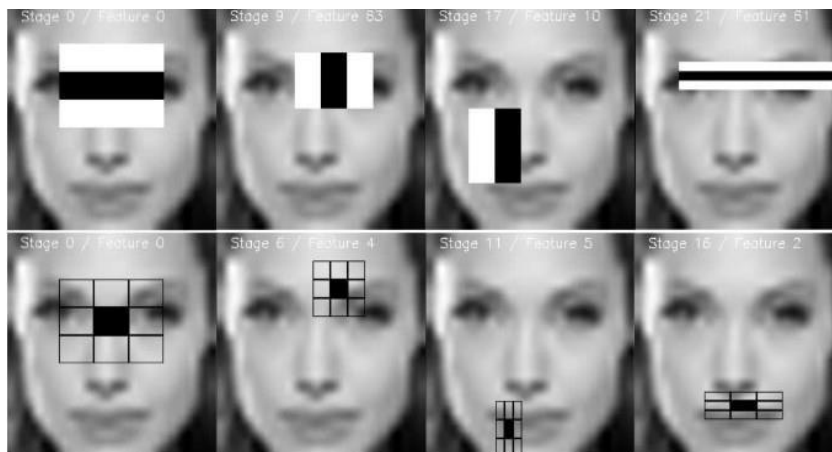


Fig4.1.Haar features

#### Histogram of Oriented Gradients using Dlib

Second method that is quite popular and widely used is under Dlib and uses basic features called Histogram of Oriented Gradients (HOG). This is an application of the authentic paper by Dalal and Triggs.[14]. The main concept underlying HOG is extraction of elements into a vector, and giving it into an algorithm classification like a Support Vector Machine such as the one that will assess if a face (or any other entity trained to identify originally) is present within a region or not. The elements that are derived using the technique are the distribution (histograms) of directions of gradients (oriented gradients) of the image. Gradients help us to detect such regions as they are basically huge around edges.”

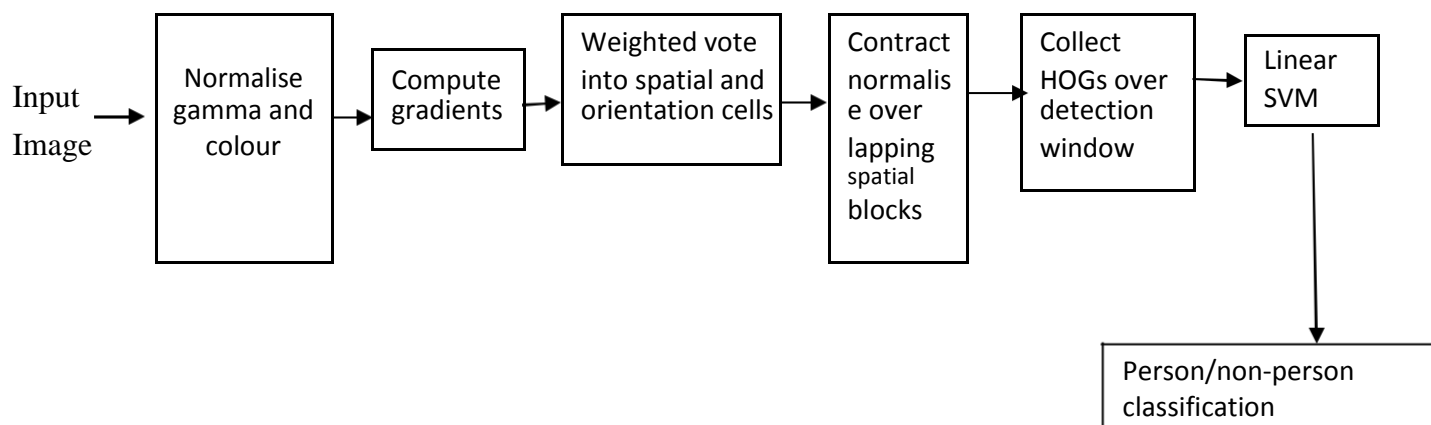


Fig4.2 Convolutional Neural Networks using Dlib/OpenCV

Convolutional Neural Network abbreviated and popularly known as CNN are feed-forward neural network. Their main purpose is in the field of computer vision. They can be used to pre-treat an automated image and also a dense neural network part. CNNs are specially designed types of neural networks that help process data having grid-like topology. Their outlay is based on the visual cortex of animals. The very name of the CNNs resulted from the truth that we convolve the initial image input within a huge set of filters. The parameter to choose remains the number of filters to apply, and the dimension of the filters. The stride length is the dimension the particular filter. The value of strides ranges between 2 and 5, may vary depending on the model being integrated with.”

An approach neural networks based is chosen, focusing on convolution neural networks along with OpenCV in python.

## FINAL RESULTS

The results plotted by the model show that it obtained 99% liveness detection accuracy on the validation set put to test.

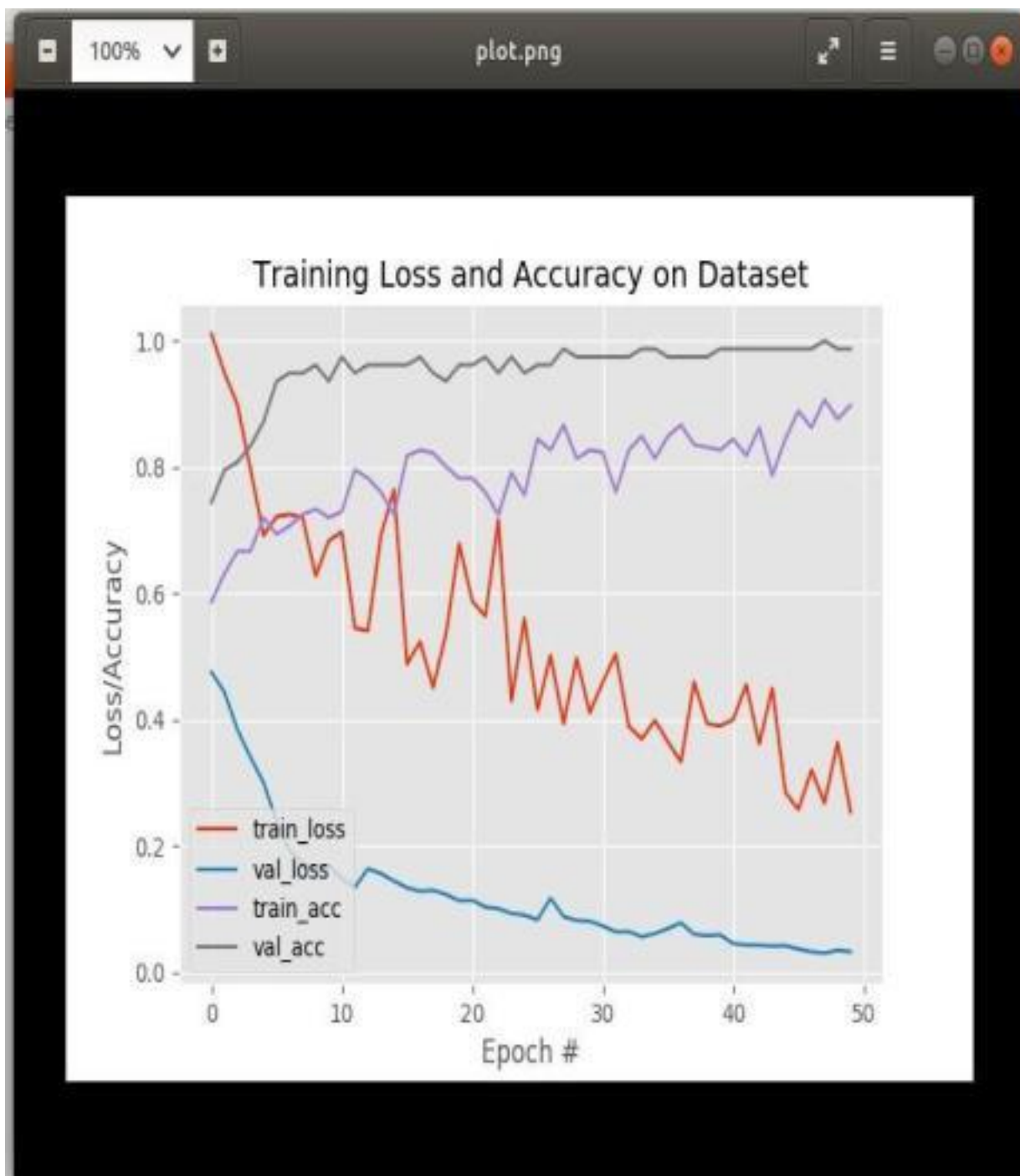


Fig4.3. Plot.png depicting statistical data about the network

## PURPOSE

Systems of facial recognition have become quite popular over the last few years ,causing them to being included in our daily lives as an important inseparable digital enhancement.The existing examples involve mobile phones,building entries,security systems,crypts ad locking systems using facial recognition for authentication,widely.The benefits of the technology are also visible in industrial sector.The globally spread mobile industry is using facial recognition as an authentication system to provide an extra layer of biometric security.It is being used to make communities more secure by law enforcement agencies.Travellers' safety and comfort have benefitted at airports.The technology has also helped keep an eye on crimes and violence.

But the major feature here is being able to distinguish between fake faces and real faces.This will provide an extra boom to this popular facial recognition system.

This an important aspect to ponder and work over because spoofing could destroy the very virtue of the technology by allowing spoofers to hack into and get through authentication,leaving data vulnerable to misuse and heavy losses.Thus,if an unauthorised person is able to get through the facial authentication step, the whole model becomes useless.

“

Thus,alot of efforts are being to made to make the system hack proof .This can be done by integrating eye-ball movement or light reflection or lip movement features.Technological advances can help make the sytem more and more secure .

## ADVANTAGES OF THE MODEL

The network is quite shallow and has minimum possible parameters for two reasons:

To avoid the probability of over-fitting on a small dataset.

To ensure this liveness detector is quick, capable of running in real-time (even on resource-constrained devices, such as the Raspberry Pi).

Overall, the liveness detector was able to obtain 99% accuracy on the validation set.

The algorithm can very easily be expanded and integrated to other types of spoofed faces, including print outs, high-resolution prints, etc.”

Collecting the dataset is feasible as per the requirements of the specific model.



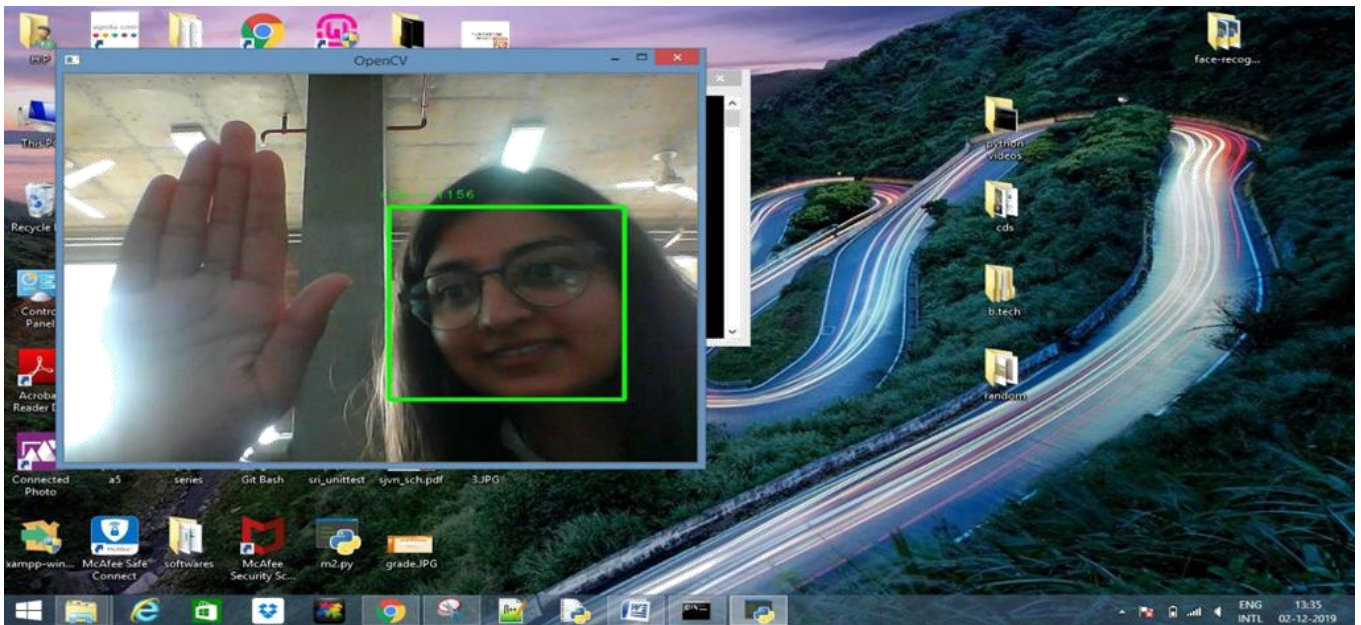


Fig 4.4 Also the model depicts faces correctly

## SHORTCOMINGS OF THE MODEL

With continued testing it was noticed that the model is slightly biased towards the faces in the dataset which is obvious as that was all the model was trained on.

The basic limitation of this liveness detector is the limited small dataset.

The liveness detector was only trained on spoof attacks from holding up a screen — it was not trained on any images or photos that were printed out.”

In case of showing 2(or more) faces, the model either didn't recognize it as a face or couldn't classify it correctly as depicted in the figure below.

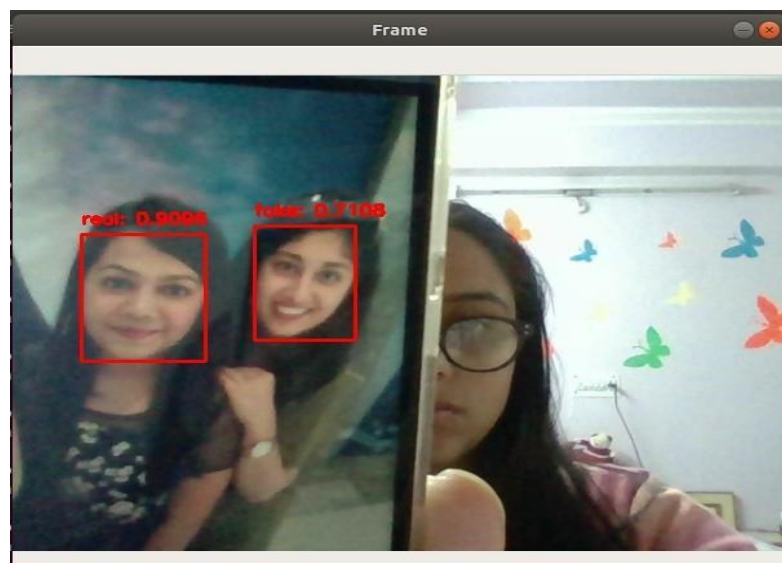


Fig4.5. Frame window screenshot depicting an error in the model

## CONSTRAINTS FACED

There always exist some restraints and barriers in the designing of model. Presented below are some of such constraints faced in the development on the "face liveness detection using python" model.

- Dataset: "The problem associated with dataset is that the model requires a very specific and typical type and the ones readily available in the market are all mostly based on Caucasian or Southeast Asian. Thus, for testing the model on people with Indian features data set requires to have such features for proper training and results. Also, if model is not provided a huge, diverse database, the system may not work properly and efficiently
- Specific orientation: The datasets also contain videos of some individual faces for better training of the model. However, the specificity of the trained model doesn't pick up people of different regions with characteristically different features. Limiting its overall performance.
- Additional image/face sources: "The time constraint doesn't allow one to integrate high quality printed pictures to be differentiated against as fake ones from the real ones. Thus, the model will always require better advances and technological integration.
- Pose variations: Changing one's alignment with respect to the camera or making different kinds of faces by yawning or covering some part of the face make it difficult to match the features in the existing set to features on which the model is trained, as doing so changes the face features to a large extent.
- Varying illumination conditions: "Sitting in varied types of lighting systems directly affect the model by bringing variations of illuminations. Just as in low levels of illumination it becomes very difficult to find coordinates of features, same is true for an extremely illuminated seating area. Just as in a completely dark room, there will be no illumination so there will be no face detection leaving feature recognition far behind. Thus, illumination can affect the accuracy of the model. Also, some lighting patterns affect facial features to be appearing different what they originally are, and again affecting the performance of the model.



## CHAPTER 5 CONCLUSION

### 5.1 CONCLUSION

So as to create security frameworks for character confirmation, face acknowledgment (FR) innovation has been applied. One among the most issues of applying FR innovation is that the frameworks are especially in danger of assaults with satirizing faces (e.g., second pictures). To shield from these assaults and to fortify the duty of FR frameworks, a few enemy of parodying approaches are as of late created.

It's been inferred that face parody identification is the procedure that is been applied to help security of the bio-decimal standard for measuring. Anti-caricaturing is changing into a critical issue in biometric validation frameworks. It's incredibly pivotal for a system to appropriately find and stop assailants particularly with the shifted variety of assaults.

The support and assistance of physical properties combined with biological properties of human beings is what makes a biometric system that is capable of offering much more safe environment for a system of security. Facial recognition may be a terribly effective tool which will facilitate law and its enforcers acknowledge violators and package firms are investing the very technology helping users to be able to reach their technology. This very technology can be further developed to be capable of being employed and used in different environments like ATMs, using confidential data files, or maybe different sensitive materials. This will formulate different safety measures like passwords and keys to become a thing of the past. Another way that innovators are wanting to implement face recognition is at intervals subways and different transportation retailers. They're wanting to leverage this technology to use faces as credit cards to pay money for your transportation fee, rather than having to travel to a booth to shop for a price ticket for a fare, the face recognition would take your face, run it through a system, and charge the account that you've created. This might probably contour the method and optimize the flow of traffic drastically.

## 5.3 APPLICATIONS AND CONTRIBUTIONS

### MANUAL MEASUREMENTS BY BLEDSOE (1960S)

Woodrow Wilson Bledsoe is popularly famous for his contribution in the field of facial recognition systems. In the 1960's, he developed a device which with the help of a RAND tablet could help classify by hand of facial photographs. RAND tablet was to help input, with a stylus that used electromagnetic pulses, into a grid horizontal coordinates and vertical coordinates. This in turn allowed the system as a whole to track the coordinate points and locations of varied features of the face such as ears, eyes, nose, etc.

Thus, a database was created. The database could be further used to retrieve similar images in the system by matching facial features, closely, whenever a photograph of a new individual was given to it. The advances were however limited due to under developed facial recognition systems and comparatively lower processing power of the computers at the time of his research. But the advances however limited, cannot be ignored and can be considered as an important and progressive step in the field of facial recognitions and biometric systems.

### INCREASED ACCURACY WITH 21 FACIAL MARKERS (1970S)

Goldstein, Harmon, and Lesk in the 1970's, added extraordinary efficiency to the facial recognition systems that were manual in nature. The main aspect they brought about was the computerised or automatic recognition of faces by using prominent facial features such as thickness of lips and color of hair. In total 21 detailed features were used in the process.

### EIGENFACES (LATE 1980S-EARLY 1990S)

Sirovich and Kirby experimented with application of linear algebra into the system of facial recognition, in 1980. A way for low-dimension representation of the images of faces was being searched upon. Sirovich and Kirby's work showed that a proper facial feature analysis done by collecting data of facial images could result in the culmination of basic features. They also successfully practically proved the usage of over a hundred values could result in coding of a simple face image efficiently.

Further research and technological advances led to recognition of faces from a photograph by Turk and Pentland in 1991. This was a major achievement in the field of automated facial recognition systems however there were various technological and economic restrictions. This automated facial recognition was first of its kind.

### FERET PROGRAM (1993-2000S)

The Facial Recognition Technology (FERET) program was brought about by Defense Advanced Research Projects Agency (DARPA) and the National Institute of Standards and Technology beginning in the 1990s. This was done to boost the facial recognition system market. The creation of the database was also a part of the program. Updates were made in the database from time to time. The very first update helped inclusion of high-resolution 24-bit color versions of images.

The test set constituted non-moving 2413 facial images of around 856 persons. A large database was created with a hope that it may bring about new advances and further make the system better than before. SUPER BOWL XXXV (2002)

Facial recognition was used by the law and order enforcement officials at the Super Bowl, 2002. This was seen as a major test of the facial recognition system. Although, the overall test declared the system as a failure but very few defaulters could be detected through the system. Facial recognition was not considered fit to be deployed into authentication systems without further advances, updates and critical testing. This was due to critical comments on false positives being proved by the system. Another limitation was that the system did not show good results when put in test in large crowds. Thus, the system was deemed unfit for purposes like event security.

#### FACE RECOGNITION VENDOR TESTS (2000S)

The Face Recognition Vendor Tests (FRVT) were started by the National Institute of Standards and Technology (NIST) in the early 2000's. FRVT was a further advancement of the Facial Recognition Technology (FERET). The major aim of FRVTs was to be able to provide relevant information to designated authorities such as law and order enforcement agencies for the application of facial recognition systems. This was done through repeated evaluations to make the system commercially viable and secure.

#### LAW ENFORCEMENT FORENSIC DATABASE (2009)

The Pinellas County Sheriff's Office made a forensic database which helped designated persons to be able to access the pictures archive section of the state's Department of Highway Safety and Motor Vehicles (DHSMV), in 2009. By 2011, around a hundred and seventy authorised personnel had been fitted and given cams that had the feature of taking clicking photos of suspects for further cross-checking and verification amongst the ones in the existing database. This affected the policing by helping in more arrests and violation investigations that was not seen before without the technology,

#### SOCIAL MEDIA (2010-PRESENT)

A very popular social media platform, Facebook also tried its hands on facial recognition application and functionalities, in 2010. This proved useful for Facebook as it helped recognise users whose faces or features appeared frequently in the updated statuses of Facebook users.

This feature did not come without shortcomings, a lot of negative comments were seen soaring around which questioned the privacy breach by the feature. However, this negativity did not affect Facebook user base much as it still has millions of users and the tagging feature being used quite frequently.

#### FIRST PROMINENT USAGE OF FACE RECOGNITION IN AN AIRPORT (2011)

The government of Panama, together with then-U.S. Secretary of Homeland Security Janet Napolitano, initiated an experimental programme of FaceFirst's system of facial recognition in the year 2011, to help reduce on illegal actions in the airport of Panama's Tocumen (infamously named the hub for smuggling of drugs and organized crime).

This experimental program turned to be a huge success and was heavily appreciated as it helped the system led to the apprehension of many suspects at the Interpol. The facial recognition system deployment

at Tocumen's airport remain the largest of its kind up to date.

## OSAMA BIN LADEN IDENTIFICATION (2011)

The areas of application of face recognition has been widespread, some of them being for forensics by law and order enforcement agencies and military related professionals. It is often quite efficient in identification of dead bodies. The fact is that the recognition based on facial features was put to help verify the identity of Osama bin Laden after his killing in a raid by the U.S.

## LAW ENFORCEMENT AGENCIES ADOPT MOBILE FACE RECOGNITION (2014)

The Automated Regional Justice Information System (ARJIS), was the one to start supply of FaceFirst mobile technology to its consumers to use facial recognition technology for enforcing of law and security terms. ARJIS, a composite justice enterprise of criminal in a network promoting sharing of valuable files and data with the local, state and federal law and order agencies of enforcement, wanted to resolve a composite issue: real time recognition for persons who possessed no ID or did not will to be recognised. San Diego police, DOJ, FBI, DEA, CBP and U.S. Marshalls are examples of various agencies that implemented the facial recognition model for authentication and identification purposes.

## FACE RECOGNITION "INEVITABLE" FOR RETAIL (2017)

There was an observation that fast growing retail industry has adopted and implemented facial recognition into its system faster than any other industry. In a quite recently held webinar, D&D Daily Publisher and Editor Gus Downing presented that facial feature recognition is on an "inevitable path to retail adoption." Downing, a loss prevention thought leader, sees a lot of potential in facial recognition system for retailers and also believes that its adoption will be highly advantageous.

## IPHONE X (2017)

A global brand, Apple released the iPhone X in 2017, which had face recognition as one of its new and attractive characteristics of the new model. This characteristic feature was basically used for lock screen authentication, providing an added security layer. It became popular among the masses within no time, setting new global security standards.

Face ID was launched by the very famous and valuable company Apple, on the launching event of iPhone X as a successor to the old and common biometric system, a fingerprint primarily based authentication service system. This module included a system that used face recognition as an element of sensing which further combined two important and reliable parts: a "Romeo" module [31] The design pattern that was spread and distributed to an area called "Secure Enclave" inside the device's most important unit that is the central process unit (CPU) to validate and make a comparison with the mobile's owner's facial features. The facial pattern wasn't available to use and spoof for the employees and database of Apple. The

technique did not work with closed that is eyes, in an attempt to stop spoofing and hacking. The methodology learnt from variations and changes that came within a customer's features, and thus it worked with accessories such as caps, glasses, spectacles, etc.

## WATCHLIST AS A SERVICE (2017)

Facial recognition technology has benefitted people in all spheres. FaceFirst launched WatchList as a Service (WaaS) at the conference of the NRF Protect. WaaS will serve as a unique and emerging facial recognition information base specifically made to assist avoid events such as shoplifting and serious violations. WatchList also constitutes a proper database of verified violators that pose security threats, theft or violent crime issues. The database works in tandem with the FaceFirst biometric surveillance platform, that takes into account feature matching technology to raise safety alarm about real-time threats.

## CUSTOMER SERVICE

Banks in Malaysia installed provisions which used "Face Recognition" to verify valuable users of the banking system such that can that the bank provided the personalized and authentic customized service. This is how, banking systems were able to produce more reliable revenues by keeping such customers and keeping them happiest with the service.

## HYDERABAD's AIRPORT EXPERIMENT.

The Rajiv Gandhi International Airport in its complex has initialised Face Recognition (FR) system on a pilot basis for passengers for entry permission into the aerodrome. The automatic procession of passenger's would be initiated based on the system of facial recognition at various points of checking namely entry point check, entry in to Security Check, aircraft boarding, adding to this this would also be the facilitation self-bag drop and check-in, which will be done using system of facial recognition to recognise and authenticate the customers and data recall. Digi-Yatra would help in the facility of non-paper travel and helping in turn avoiding the check of identity at various points, this is what has been said by an official release earlier.

## RUSSIAN FEDERATION



The system of CCTV in the capital of the nation will help in recognising the faces by using the exploitative formula of supported neural networks. Town camera recordings region of unit analysed in dynamic time. Faces which appear on the screen region unit scanned might and might be checked against multiple bases of information , some of the popular being the police information base, to spot a suspect. The analytical system can even facilitate police recreate a suspect's movements round the town. The system searches for connected recordings from varied CCTV cameras and identifies constant face from many sightings.

About 16,000 users within the police, and federal and regional security agencies area unit connected to the city's security system. Every jurisdiction has its own access level, that helps maintain info confidentiality

## THE US DEPARTMENT

The Federal Bureau of Investigation has conjointly formulated its Next Generation Identification program to amalgamate face recognition, also as a lot of ancient life science for example fingerprints and even iris scans, which have the tendency to pull from each and every criminal and also civil databases. The federal General answerableness workplace critically analysed and disliked the Federal Bureau of Investigation for not addressing numerous issues associated with accurateness and individual security. It was in 2019, researchers rumored that Immigration and Customs social control uses automatic face recognition computer code was totally not in favour of state license databases, together with fore a few of states that give licenses to unregistered immigrants.

## CHINA

It is as recently as in 2017, biometric identification and computer science technology has been deployed by the Chinese in the state. Reporters who were on a visit to the region discovered police investigation cameras put in each hundreds metres just about in many cities, similarly as biometric identification as in checkpoints at various regions like gas stations, looking centers, and house of prayer entrances.

## References

- [1] Wei Bao, Hong Li, Nan Li and Wei Jiang, "A liveness detection method for face recognition based on optical flow field," *2016 International Conference on Image Analysis and Signal Processing*, Taizhou, 2009, pp. 233-236. doi: 10.1109/IASP.2009.5054589  
URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5054589&isnumber=5054562>
- [2] R. Tronci *et al.*, "Fusion of multiple clues for photo-attack detection in face recognition systems," *2011 International Joint Conference on Biometrics (IJCB)*, Washington, DC, 2011, pp. 1-6.  
doi: 10.1109/IJCB.2011.6117522  
URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6117522&isnumber=6117470>
- [3] G. Kim, S. Eum, J. K. Suhr, D. I. Kim, K. R. Park and J. Kim, "Face liveness detection based on texture and frequency analyses," *2012 5th IAPR International Conference on Biometrics (ICB)*, New Delhi, 2012, pp. 67-72.  
doi: 10.1109/ICB.2012.6199760  
URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6199760&isnumber=6199747>
- [4] J. Määttä, A. Hadid and M. Pietikäinen, "Face spoofing detection from single images using micro-texture analysis," *2018 International Joint Conference on Biometrics (IJCB)*, Washington, DC, 2018, pp. 1-7.  
doi: 10.1109/IJCB.2018.6117510  
URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6117510&isnumber=6117470>
- [5] B. Peixoto, C. Michelassi and A. Rocha, "Face liveness detection under bad illumination conditions," *2015 18th IEEE International Conference on Image Processing*, Brussels, 2015, pp. 3557-3560.  
doi: 10.1109/ICIP.2015.6116484  
URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6116484&isnumber=6115588>
- [6] R. Tronci *et al.*, "Fusion of multiple clues for photo-attack detection in face recognition systems," *2014 International Joint Conference on Biometrics (IJCB)*, Washington, DC, 2014, pp. 1-6.  
doi: 10.1109/IJCB.2014.6117522  
URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6117522&isnumber=6117470>
- [7] Mengyang Liu, Hong Fu, Ying Wei, Yasar Abbas Ur Rehman, Lai-Man Po, and Wai Lun Lo "Light field-based face liveness detection with convolutional neural networks," *Journal of Electronic Imaging* 28(1), 013003 (8 January 2019). <https://doi.org/10.1117/1.JEI.28.1.013003>
- [8] Jourabloo, A., Liu, Y. & Liu, X. (2018). Face De-Spoofing: Anti-Spoofing via Noise Modeling. In Proceedings of the European Conference on Computer Vision (ECCV), Munich, Germany (pp. 88-93), DOI:10.1007/978-3-030-01261-8\_18
- [9] Pan, G., Sun, L., Wu, Z. et al. (2007). Eyeblick-based Anti-Spoofing in Face Recognition from a Generic Webcam. In IEEE International Conference on Computer Vision (pp. 144-147).
- [10] Pereira, T., Komulainen, J., Anjos, A., De Martino, J. M., Hadid, A., Pietikäinen, M. & Marcel, S. (2014). Face liveness detection using dynamic texture, *EURASIP Journal of Image and Video Processing*, 2014(1):2, 1-15.
- [11] Wen, D., Han, H. & Jain, A. K. (2015). Face Spoof Detection with Image Distortion Analysis, *IEEE Transactions on Information Forensics & Security*, 10(4), 746-761, DOI: 10.1109/TIFS.2015.2400395

- [12] Zhang, Z., Yan, J., Liu, S., Lei, Z., Yi, D. & Li, S. Z. (2012). A face anti-spoofing database with diverse attacks. In *IEEE International Conference of the Biometrics* (pp. 26-31).
- [13] Erdogmus, N. & Marcel, S. (2013). Spoofing attacks to 2D face recognition systems with 3D masks. In *International Conference of the Biometrics Special Interest Group, (EPFLCONF-192407)* (pp. 184-189).
1. Wei Bao, Hong Li, Nan Li and Wei Jiang, "A liveness detection method for face recognition based on optical flow field," *2016 International Conference on Image Analysis and Signal Processing*, Taizhou, 2009, pp. 233-236. doi: 10.1109/IASP.2009.5054589 URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5054589&isnumber=5054562>
  2. R. Tronci *et al.*, "Fusion of multiple clues for photo-attack detection in face recognition systems," *2011 International Joint Conference on Biometrics (IJCB)*, Washington, DC, 2011, pp. 1-6. doi: 10.1109/IJCB.2011.6117522 URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6117522&isnumber=6117470>
  3. G. Kim, S. Eum, J. K. Suhr, D. I. Kim, K. R. Park and J. Kim, "Face liveness detection based on texture and frequency analyses," *2012 5th IAPR International Conference on Biometrics (ICB)*, New Delhi, 2012, pp. 67-72. doi: 10.1109/ICB.2012.6199760 URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6199760&isnumber=6199747>
  4. J. Määttä, A. Hadid and M. Pietikäinen, "Face spoofing detection from single images using micro-texture analysis," *2018 International Joint Conference on Biometrics (IJCB)*, Washington, DC, 2018, pp. 1-7. doi: 10.1109/IJCB.2018.6117510 URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6117510&isnumber=6117470>
  5. B. Peixoto, C. Michelassi and A. Rocha, "Face liveness detection under bad illumination conditions," *2015 18th IEEE International Conference on Image Processing*, Brussels, 2015, pp. 3557-3560. doi: 10.1109/ICIP.2015.6116484 URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6116484&isnumber=6115588>
  6. R. Tronci *et al.*, "Fusion of multiple clues for photo-attack detection in face recognition systems," *2014 International Joint Conference on Biometrics (IJCB)*, Washington, DC, 2014, pp. 1-6. doi: 10.1109/IJCB.2014.6117522 URL: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=6117522&isnumber=6117470>
  7. Mengyang Liu, Hong Fu, Ying Wei, Yasar Abbas Ur Rehman, Lai-Man Po, and Wai Lun Lo "Light field-based face liveness detection with convolutional neural networks," *Journal of Electronic Imaging* 28(1), 013003 (8 January 2019). <https://doi.org/10.1117/1.JEI.28.1.013003>
  8. Jourabloo, A., Liu, Y. & Liu, X. (2018). Face De-Spoofing: Anti-Spoofing via Noise Modeling. In *Proceedings of the European Conference on Computer Vision (ECCV)*, Munich, Germany (pp. 88-93), DOI:10.1007/978-3-030-01261-8\_18
  9. Pan, G., Sun, L., Wu, Z. et al. (2007). Eyeblick-based Anti-Spoofing in Face Recognition from a Generic

Webcamera. In IEEE International Conference on Computer Vision (pp. 144-147).

10. Pereira, T., Komulainen, J., Anjos, A., De Martino, J. M., Hadid, A., Pietikainen, M. & Marcel, S. (2014). Face liveness detection using dynamic texture, EURASIP Journal of Image and Video Processing, 2014(1):2, 1-15.
11. Wen, D., Han, H. & Jain, A. K. (2015). Face Spoof Detection with Image Distortion Analysis, IEEE Transactions on Information Forensics & Security, 10(4), 746-761, DOI: 10.1109/TIFS.2015.2400395
12. Zhang, Z., Yan, J., Liu, S., Lei, Z., Yi, D. & Li, S. Z. (2012). A face anti-spoofing database with diverse attacks. In IEEE International Conference of the Biometrics (pp. 26-31).
13. Erdogmus, N. & Marcel, S. (2013). Spoofing attacks to 2D face recognition systems with 3D masks. In International Conference of the Biometrics Special Interest Group, (EPFLCONF-192407) (pp. 184-189).
14. N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05), San Diego, CA, USA, 2005, pp. 886-893 vol. 1.

# PLAGIARISM REPORT