International Journal of Scientific Research in Science and Technology
Print ISSN: 2395-6011 | Online ISSN: 2395-602X (www.ijsrst.com)
doi : https://doi.org/10.32628/IJSRST523102103

Face Anti-spoofing and Liveliness detection using Mobile net and Haar cascade Algorithm

T Naveen Prasad[1], Dr. B. Anuradha[2]

*[1]M. Tech student, Department of Electronics and Communication Engineering, Sri Venkateswara University College of Engineering, Tirupati, Andhra Pradesh, India

[2]Professor, Department of Electronics and Communication Engineering, Sri Venkateswara University College of Engineering, Tirupati, Andhra Pradesh, India

ARTICLEINFO

Article History:

Accepted: 01 April 2023
Published: 22 April 2023

Publication Issue
Volume 10, Issue 2
March-April-2023

Page Number
710-723

ABSTRACT

A common biometric approach is another way of face recognition techniques. The method is convenient, easy access to the user and direct comparison with other methods of face recognition due to its rapid development in recent years. Resisting spoofing attacks made on face recognition systems requires Face Anti-Spoofing Systems (FAS) techniques. FAS based on deep learning perform exceptionally well and dominates this area with the emergence of large-scale academic datasets in recent years. The data requirements for training effective anti-spoofing models in this field, however, are large, and there is no way to perform live spoofing. Our paper proposes a combined method of face liveliness detection using Haar-Cascade algorithms and mobile-net classifiers. As a contribution to stimulating future research, we present an overview of technique like deep learning-based FASs. It covers numerous novel and insightful Anti-spoofing approach structured using the modules,1) Eye opening action evaluates with the blinking eye systems and 2) mobile-net classifier module which makes use of a pre-trained version. We wrap up our study with gradually merged these two modules and added them to a basic facial recognition system. Software Python-based results comparisons between classifiers are used to explain efficiency of the suggested approach.

Keywords : Face Anti-Spoofing (FAS), mobile-net classifier, Liveliness detection, Haar-Cascade algorithm deep- learning.

I. INTRODUCTION

There is a critical need for security safeguards against spoof attacks for the general population. The security industry's fastest-growing subset is biometrics. The recognition techniques like facial recognition, handwritingverification, fingerprint recognition, hand geometry, scanning techniques like iris and retinal are

a few of the well-known methods for identification. Face recognition technology is one of these technologies that has advanced most recently and is more straightforward, user-friendly, and practical than other approaches. As a result, it has beenused in many security systems. There is a critical need for security safeguards against spoof attacks for the general population. The area of this security industry that is expanding the fastest is biometrics. The recognition techniques like facial recognition, handwriting verification, fingerprint recognition, hand geometry, scanning techniques like iris and retinal scanners are a few of the well-known methods for identification. Face recognition technology is one of these strategies that has advanced most recently and is more straightforward, user- friendly, and practical than other approaches.

As a result, ithas been used various safety system with security alert. The face recognition algorithms are easy tricked using facial images such as portrait photos. Liveness detection is necessary for a secure system in order to prevent this spoofing.

Currently, liveness detection has seen a large study activity in the communities of iris and fingerprint recognition. However, there are very few solutions available for face recognition to address this issue. The act of liveness depends on the live and non-liveness different feature space for the recognition. A biometric system performance must increase Liveness detection. Determining existing biometric system security against spoofing's dependability is a crucial and difficult topic. The typical assault techniques in face recognition can be divided into a few different groups.

Static and dynamic assaults are terms used to describe face spoofing in software-based systems. While dynamic attacks use video replays or several photographs in a sequence, static 2D demonstration spoofing attacks usephotos or masks. Attacks that use static 3D demonstrations may employ 3D sculptures, prints, or even masks, whilst animated versions employ sophisticated robotics to recreate facial expressions, replete with makeup. The most reliable method of anti-spoofing would be 3D cameras or photoplethysmography. Since we can distinguish between a face and a flat object, specific pixel depth guidance may provide great precision against demonstration attacks.

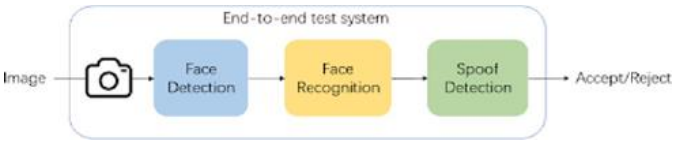


Fig 1: General block diagram of face recognition and face anti-spoofing technique

The facial motion detection and another method facial texture analysis be the two main section among the variousface liveness detection techniques. It is assumed that people will make particular facial gestures in approaches based on facial motion detection, and the detection will determine how live the subject is. Different texture patterns can be seen in real fake facial photos. Reconstructing faces from camera images, it is a basic fact,degrades the clarity of facial expressions and leaves gaps in reflectivity. Using manufactured color texture features, like fluctuations in RGB (Red Green Blue) or LBP, here indicates the anti-spoofing problem and another liveness detection in this article (Local Binary Pattern).

I.1 Local Binary Pattern Histogram (LBPh):

The suggested method uses the Local Binary Pattern histogram (LBP) to find and identify human faces. For face identification, use the Local Binary Patterns histogram algorithm (LBP). The method performs best for texture descriptors which operates on the local binary operator. It is commonly used in facial recognition because of how easily it can be computed and how well it can discriminate. To do this, the following steps are necessary:

- creating datasets
- face acquisition
- feature extraction
- classification

I.2 Mobile net:

Convolutional neural networks of the mobile-net variety were created in applying embedded along with mobile vision applications. An approach based on the simple design, which involves the separable convolution in depth- wise level. The design creates compact neural networks to the deep-learning neural network approach. The feature involved in the deep-learning approach with reduced latency for embedded platform and mobile devices.

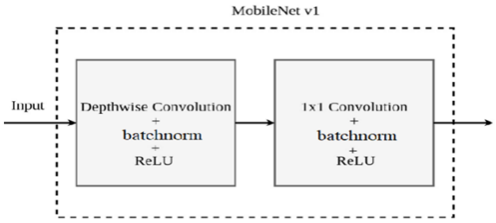


Fig :2 Mobile-Net V1 Block Diagram

Fig. 2 shows how the Mobile-net layers transform the input image's pixels into output features that characterize the image's contents and transmit these features to the other layers. Mobile-nets are largely constructed using the feature component of convolutions with depth-wise separable technique introduced first time in inception models and then utilized to lessen the computation in the first few layers. We can have a smaller network, fewer parameters, and faster performance by employing Mobile net.

I.3 Haar-Cascade:

In machine learning, Haar Cascade techniques involves both the large number of positive and negative images for the trained classifier.

- Positive images attributes – the classifier trained for recognizing these images, which present in these photographs.
- Negative Images – the images or the objects were trying to detect.

The Haar-Cascade algorithm detects features by averaging the total of pixels in lighter and darker regions. The average pixels are then converted into edges using edge detection, and the distance between them is determined. The following is a summary of the most intriguing the technique for face liveness detection, which describe below the section of paper. The background theory for the suggested system is described in the upcoming section. The experimental result for the method and the analytics surrounding them are then highlighted. The references used in this paper are listed after the conclusion, which brings the essay to a close.

I.4 Related Works:

Numerous studies have looked into the ideas behind face spoofing. Some of the more intriguing liveness detection techniques are described in this section.

Using the CNN Classifier technique, Raden Budiarto Hadiprakoso et al. (2020) [1] created a Face Anti-Spoofing. Both the conventional convolutional neural network and deep learning were employed. For detecting lip movement and evaluating if the face detected is real or false, they employed a lip detection module with a basic recurrent neural network. The model was trained after they collected faces from the dataset. To make teaching faces easier, they employed transfer learning to train a new model from a previously learned model. The trained model is given to CNN classifier for face detection after training. When a face is found, it is passed to the lip detection module, which uses variations in lip movement that decides the face is real or not. Following this discovery of spoofing, they ran tests in Python utilizing keras and OpenUPI on an Android platform. They have managed to detect fake photos with an accuracy of 90.39 percent.

A literature review on Anti-spoofing Techniques for Facial Recognition captured with RGB cameras for Generic Consumer Devices as designed by author Zuheng Ming [4] et al. in 2020. The past 20 years have seen a thorough examination of facial Presentation Attack Detection (PAD), which simply needs an RGB camera in common consumer devices. From an experimental standpoint, they have presented a condensed overview of publicly accessible databases and a comparison of findings from several PAD studies.

A study on Face Spoofing detection and a survey of several approaches have been done with the help of author Gency et al. (2020) [3]. They gave an overview of the work that worked in the field of face anti-spoofing over the past ten years, as well as the drawbacks of models.

A Face Anti-Spoofing Detection system utilizing Enhanced Local Binary Patterns was created by Karuna Grover [6] et al. in 2019. They created a geneLBPnet by fusing a modified LBP descriptor with a conventional convolutional neural network. This created an Enhanced Local Binary Pattern. This approach, which makes use of texture analysis, has the issue of being useless in low light. NUAA database was used for the experimental portion. Although they have achieved 98% accuracy and a lower Equal Error Rate, features are challenging to capture in low lighting.

A Face Anti spoofing system was created by Mohammed Rezwan [7] et al. (2019) using texture-based algorithms and filtering techniques. The DoG filtering approach is updated, and a local binary pattern variance (LBPV)-based strategy is employed. Feature vector extraction is done using SVM. Face image input is done using the NUAA database. A measure of false detection parameter like False Acceptance Rate and the rejection rate for False Rejection Rate were used to measure accuracy.

KNN classification was created by Samrity Saini [8] et al. (2019) for face spoof detection. They built the method on extracting eigen features with LBP and classifying data with KNN classifier. For the face image input, they used the AT&T database. When using an SVM classifier in comparison to an existing system, accuracy and execution time are boosted.

Image spoofing method for detection process was developed by Arida Kartika [10] et al. (2018) utilizing a local binary pattern and a local binary pattern variance. They employed the Local Binary Pattern (LBP) and Local Binary Pattern Variance texture analysis approach (LBPV). Here, skewness is calculated. K-Nearest Neighbor has been used for categorization purposes (KNN). From the NUAA impostor dataset, they employed 3362 authentic photos and 5761 fake images. They have achieved an accuracy of 87.22% in identifying fake and authentic photos.

A technique for Face Detection utilizing a KNN classifier system was developed by Sakshi Jha et al. [11] (2018). To assess the textual features pre-set in the test image, they employed the DWT algorithm. KNN classifier is used to categorize the spoofed and non-spoofed characteristics. Here, the KNN classifier uses n-dimensional attributes to represent the training samples. Most training samples are kept in n-dimensional pattern space. The pattern space that is closest to the unknown sample is chosen using the K nearest neighbor matching with the K training samples. The classifier then returns the mean genuine value linked with the k nearest neighbor sample. For the analysis of features related to the test image, the DWT method will be used.

P.Kavitha et al. [12] (2017) conducted research on the various methods for spotting spoofing in facial recognition systems. They have also conducted research on several databases used by researchers. The work already done in the area of face anti-spoofing over the past few decades is thoroughly reviewed in this study.

Additionally, many verification jobs that are carried out with laptop webcams lack these methods entirely. These problems drive the development of a model that

can solve the issue using only RGB photos. However, the method involves few mid-range mobile phones along with formerly out-of-stock gadgets that are devoid of these sensors and computing power.

II. METHODOLOGY

Face recognition systems contain security flaws, making it possible to go around them or fool them using different face spoofing techniques. Spoofing is the act of mimicking or hiding a genuine person's identity and utilizing it to damage some kind of data. Face spoofing, also known as a attack with face presentation type, which be a crime for a dishonest user fools a system for facial recognition method by pretending to be a user registered over the network in order to acquire unwanted right of entry and constitutional rights. Face spoof attacks that can be divided into two basic categories: the method 2D face attacks and 3D face attacks another way.

II.1 2D Face Attacks:

The attacking method involves the 2D Face Attacks for face display. Face presentation attacks (FPA) using shapes with planar attributes make up this (flat surface), which include the attacks done using prints or photos. Print presentation attacks involve showing a facial recognition system a photo for a genuine user image taken with a camera or still taken as of images belongs to social media that the client has posted online. Since most public enclose online photos that may be viewed without their knowledge and could be used to launch this attack. These are the ways they are displayed.

1.Printed-photo Attack: Authentic user photographs that have been flat-rendered are used in this.

2.Screen-photo Attack: The attack accomplished via showing a genuine user's photo on an electronic device's digital screen (e.g., mobile devices, laptops, etc.).

3.Cut-photo Attack: In order to simulate the features of blinking eyes and a live face movement this assault uses a copy of printed real photos with facial features of face like eye, nose and mouth regions are detected. A second person then hides his face near the paper cuts.

4.Warped-photo Attack: Using a photo with printed that has been bent in various directions to represent facial movements.

Video attack: The term replay attack is often used to describe attacks like this. Photo attacks are not as sophisticated as face recognition systems. A victim's face video is broadcast using a tablet or large smartphone. This method produces additional natural physiological responses. There are physiological signs of life in it, such as facial expressions, blinking of eyes, lip movements, and head movements that are often absent from photographs.

II.2 3D Face Attacks:

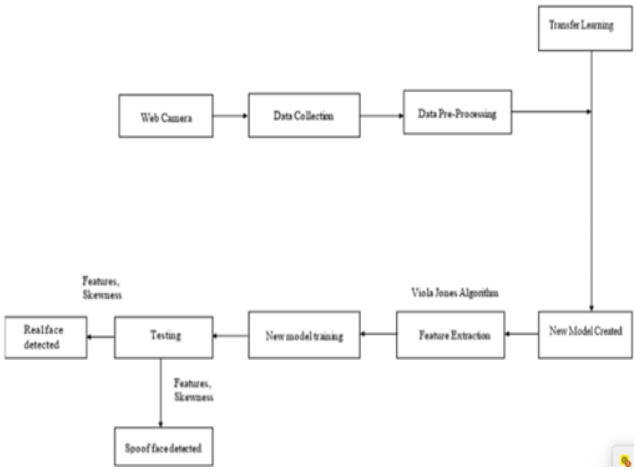
A method of mask 3D techniques of a real user's face action makes finding precise countermeasures more difficult in this case. In 3D attacks, depth cues are ineffective since 3D structures of the human face are replicated, making in use of cues deeply ineffective. Additionally, advanced cosmetics and plastic surgery can be used to attack in 3D.

II.3 Existing System:

The current system desperately needs an anti-spoofing mechanism due to the rise in different spoofing assaults. Face attacks of spoofing types executed using face display on a display device, like a smartphone or tablet. Poor facial textures caused by attacks like these are easy to identify by assessing the feel and image quality of HSV. The color spectrum in screen material, such movies or photographs, is probably going to be less varied than it is in a real person's face. The pictured faces may also have variations in regional tones. The color gamut is influenced with the display device medium, and neighboring variations in chroma, which can be understood through observing color feature of the chroma present in the channel. The method

examined whether color model, which gives the nearly everyone representation with micro-texture involves extraction of LBP (Local Binary Pattern) data through various color spaces. The development of low- cost, highly accurate based on software anti-spoofing attack detection methods that accelerated during the past few decades. Numerous anti-spoofing techniques have been created as a result of the quick growth of technology. The main purpose of using deep learning techniques is to simplify the system.

Due to its simplicity, convenience of use while training numerous face photos, and promising results, the Convolutional Neural Network (CNN) is the deep learning method that is used commonly.



Fig,3: Block Diagram of Existing system

A conventional convolutional neural network (CNN) can be used to train a face recognition system effectively, but the results are subpar. Therefore, face liveliness detection and a CNN might be utilized in implementation an efficient face anti-spoofing system. In order to differentiate between actual and spoofed faces, a traditional convolutional neural network is combined with face liveliness detection, such as detecting human facial features, tracking eye blinking, or tracking lip movement. There are some disadvantages as well which is shown below:

- In the existing system, a single dataset is trained using a convolutional neural network more frequently, which increases training time.
- This system's accuracy is poor. To attain high accuracy rates, more datasets must be trained.

Slow progress toward anti-spoofing.

II.4 Proposed System:

The proposed work implemented using Python programming that enables large toolbox in constructing solutions faster, to implement liveliness in face recognition. We examined the Object Detection algorithm Haar-Cascade, which uses edge or line detection features. Normalizing the current model, here gathered, processed, with the training of proposed framework using more than 2000 photographs of different people's eyes. This dataset also includes 1909 images of closed and open eyes, as well as 1799 images of different people's eyes with and without glasses. The implementation of eye-blink movement tracking uses this dataset. Eighty percent of the data were used for training, while twenty percent were used for validation. We implemented the concept of transfer learning to reduce training time. We employed a lightweight Mobile Net model for quicker training and inference. We have gathered a sizable dataset from open sources. Web cameras can also be used to collect real-time face photos, which can then be analyzed to create a classifier that explains the difference between the actual and fake faces.

The face anti-spoofing system can be implemented using liveliness detection and conventional Convolutional Neural Network (CNN). By utilizing efficient classifiers and other feature extraction approaches, existing systems' output can be improved. This study designs and evaluates a system for face anti-spoofing system, which uses Mobile-net while a classifier and liveliness detection. The classifier must be trained with the help of a number of steps in this process. These steps include (1) data collection (2) data pre-processing (3) model evaluation (4) model training (5) testing. There are still certain issues related to illumination, image quality, the quantity of people in the frame, etc. even after applying efficient classifiers. A liveliness module is additionally included in order to reduce these problems. We suggest eye blink

movement monitoring for the liveliness module since it is simple to calculate and more accurate at telling real faces from false ones.

Fig 4: Block Diagram of Proposed system

Data Pre-processing involves changing the face image's size, cropping it, flipping it, rotating it, changing the contrast and brightness, and more. Mobile-net learns face features using a pretrained model and processes the data gathered as illustrated in fig. 5. A new model with those attributes is produced when mobile-net learns from the previously trained model. These features are taken from the newly developed model and are retrieved using the Local Binary Pattern Histogram (LBPH) technique form methods like face recognition. Every region in a 224x224 image is divided by LBPH belongs the region for the same height and width. Every area of the 224x224 facial image uses LBPH. Each zone is given a window, which makes each one the same height and width.

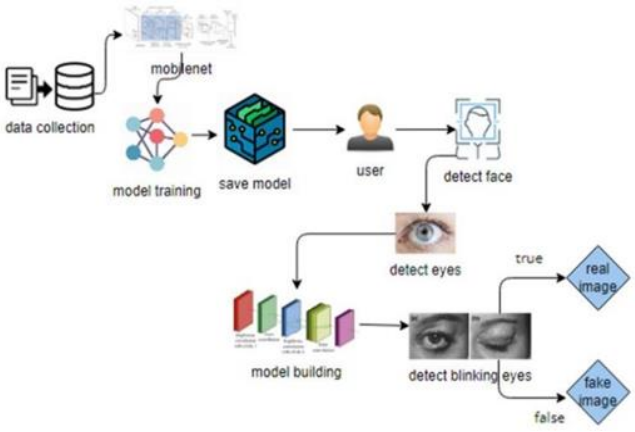


Fig 5: Architecture of Proposed system

The process of finding human faces in a scene is called face detection. The algorithmic steps help in identifying the faces that present in the frame is one of the Haar cascade classifier, as shown in figure 6. The model is initially trained using the positive (facial) and negative (non-face) images. Then, using a convolutional mask, features were retrieved.

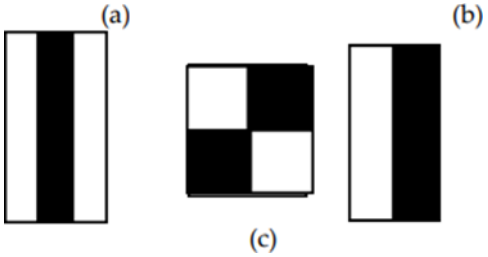


Fig. 6. Haar Feature mask used to detect face in frame.

The sum of the pixels beneath the bright pixels and the pixels under the black rectangle is subtracted from each kernel to provide a single value for each feature.

Fig.7 illustrates how the LBPH algorithm works. The method used to extract color and statistical information is called preprocessing. Therefore, we needed both a color and a grayscale image. HSV color space*6+ is used to extract the color characteristics. Grayscale color space is created from RGB color for statistical feature extraction. In mathematics, it is written as:

$Gray = ((0.3 * R) + (0.59 * G) + (0.11 * B) \ldots \ldots (1).$

Extraction of Features

An important picture parameter, such as the means of the red color, green color, and blue color and also the

mean for the grayscale, as well as the standard deviation and variances, as well as the number of data points, are used in liveness discovery to make the framework more precise and productive. These features differ for counterfeit images, offering several threshold finalization possibilities.

A) Luminance: It is a particular direction of light density travelling through a surface. It gives the frequency of light reflected from a certain surface. The surface affects the luminous characteristics. The brightness of a real face is different from a fake one. The brightness factor in a real face is arbitrary, while it is nearly same in a phony face due to the three-dimensional impact of the nose and eyes. Red, green, and blue colors in the image can be used to determine the illumination. Using the equation, it is determined:

Luminance = (0.299*R + 0.587*G + 0.114*B) …… (2)

B) Variance: The degree to which the black-and-white (grey) level deviates from the typical grey level is:

$$\mu_2(Z) = \sum_{l=0}^{L-1}(z_l - m)^2 p(z_l) \ldots\ldots\ldots \quad (3)$$

C) Mean of RGB and Mean or Gray scale: A feature is the average of all the pixel values in the R, G, and B color space. To do this, first divide each color channel apart, and then compute the means of each channel independently. The sum of the number of pixels divided by the total number of pixels equals the mean of each color.

D) Data point count: The elements in grayscale photographs that are not zero are called data points. In essence, fake photos have less data points than real images.

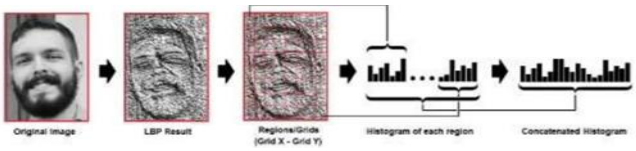
$$LBP(x_c, y_c) = \sum_{p=0}^{P-1} 2^p s(i_p - i_c)$$

Assume that the centre pixel value (xc, yc) has the intensity "Ic". And taking into account "In" as the brightness of the adjacent pixel. With the help of median pixel value act as threshold. The LPBH compares the pixel with the eight closest neighbors. The value of neighbor set to 1, when large or equal to central value and 0 otherwise. As a result, gather than 8binary values among neighbors. The obtained values are combined to create an 8-bit binary number that expediently converted to a decimal number.

The obtained decimal number, which ranges from 0-255, is referred to as the pixel LBP value. After the creation of the LBP value histogram,[10] the count of related LBP values in each part of the image is counted to construct the histogram for that region. The result of histogram obtained by merging of all the histogram values, which calculated in each region's formation called Image feature vector. The comparison of test image and database histograms to take the closest histogram for the programmed [9] Here, we compare the test image results with the dataset's photos using Euclidean distance.

$$d(a,b) = \sqrt{\sum_{i=1}^{n} |a_i - b_i|^2}$$

A simple understanding of LBPH algorithm visually can be shown in Fig 7.



Fig,7: Working structure of face recognition system using LBPH algorithm

The constructed model is forwarded to mobile-net for training after its features are retrieved. The classifier is trained to discover the same features in the photos in the dataset and in the web camera-captured images using the features that were retrieved from the model during training. In other words, feature matching is carried out, and real and fake photos are differentiated based on the features that match in both the dataset and the real-time images taken using a web camera. The function involved to categorize and train images, Mobile-net employs depth- wise separable convolution. Hence an individual input channel receives filter through single channel filter combines with depth-

wise convolution. Finally, the outputs are combines the input that creates new set of outputs. The method is divided into two layers using depth-wise separable convolution, one layer for combining and another layer for filtering. The computation processes and the model size are gradually decreases as a result of the fraction. For both layers, Mobile-net employs batch normalization and ReLU nonlinearities.

$$\hat{\mathbf{G}}_{k,l,m} = \sum_{i,j} \hat{\mathbf{K}}_{i,j,m} \cdot \mathbf{F}_{k+i-1,l+j-1,m}$$

Where, K is the depth-wise convolutional kernel of dimension DK × DK × M the place mth filter in ˆK is utilized to the mth channel in F to produce the mth channel of the filtered output function map. The model trained is then compared with those captured from a web camera as well as those from the dataset after training. Haar-cascade filter is applied on the images captured through web camera. Images taken with a web camera are processed using the Haar-cascade filter. Edge detection is used by the Haar-cascade filter to utilize Haar characteristics. The distance between edges formed from a region is calculated. The data points with positive values are taken as the part of the objects are predicted and classified by the Haar cascade algorithm as positive, and negative data points as negative.

II.5 Working of Haar-cascade Algorithm:

By dividing the regions in an image of the face into edges and measuring the distances between them, Haar-Cascade can identify characteristics. If A and B are equal, then A is lighter than B and the B region of the face could be an eyebrow. If B > A, B is lighter than A, and A region of the face might be a brow bone. The formula X=Sum of all pixels in the darker region-Sum of all pixels in the brighter region can be used to determine the value of a single feature. You can determine the separation between two features using,

(Sum of values in darker region)/2 – (Sum of values in lighter region)/2

Four stages can be used to explain the Haar Cascade method: 1. Calculating Haar Features, 2. Creating Integral Images, 3. Using Adaboost, 4. Implementing Cascading Classifiers.

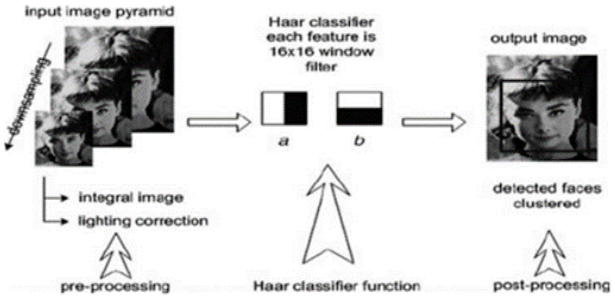


Fig 8: Working Flow of Haar-cascade

On the photos of the collected faces, a 16x16 rectangular window is applied, processed, and transformed into edges. Each image's integral factor is computed when the haar features are determined. The adaboost activation function is used to improve the image quality and strength. To create a clustered image, classifiers are finally cascaded. For detection, the clustered image is clipped into [224,224] a grayscale image. The liveliness detection phase follows the detection and recognition phase. Real and fake faces can be distinguished via liveliness detection. Using a recurrent neural network, the eye-blinking module is created in the Python programming language. The eye-blinking module [5] makes use of data gathered from earlier studies on eyes that were closed, open, wearing glasses, and not wearing spectacles. Each image is transmitted for liveliness detection after post-processing. The eye-blinking module determines the blinking rate for 30 seconds in both a video and a still image. If the blink rate is 0, the module will assume that the eyes are closed, and if it is 1, it will assume that the eyes are open. The system recognizes a face image as a real face image if the rate of blinking or blinking id is greater than the user-defined value of 20, and it recognizes a face image as a faked image if the rate of blinking is less than the user-defined value of 20. 2000 test faces from a dataset downloaded from GitHub and more than 25 distinct people have been used to test the proposed system in real time. In a comparison of the proposed system and the existing system, many parameters including accuracy, precision, recall, and f1

score have been determined. Some findings suggest that the system with proposed idea produces better outcomes than the existing system.

II.6 Research Opportunities:

Some unresolved challenges and potential research avenues for face anti-spoofing are listed.

1.Generalization to Unknown Attacks:

Numerous visual cues have already been investigated for non-intrusive spoofing detection, with outstanding results on specific databases. However, it is not possible to foresee for a single anti-spoofing technique, such as facial texture analysis, can generalize the issue in practical applications due to the diversity of spoofing attempts and acquisition settings.

2.Fusion of Countermeasures:

It is real looking to expect that no single most efficient method is in a position to discover all known, let alone unseen, spoofing assault. Hence, the hassle for spoofing that assaults ought to be damaged down for attack specifically and sub problems that are solvable acceptable mixture for complementary usage of countermeasures.

3.Biometric System and Countermeasures:

A spoofing countermeasure is typically intended to work in tandem with a recognition system rather than as a stand- alone technique. The majority of anti-spoofing research, however, tends to concentrate primarily on the spoofing detection aspect, neglecting to incorporate the countermeasures for an identified system. Effectiveness for a recognition system become an impact in practice implementing the countermeasures. Although it will make it less susceptible that may cause recognition performance to suffer.

4.Contextual Information:

Photographs of faces taken through face spoofs may visually resemble images of real faces quite closely. As a result, face spoofing detection depend on a single face image or a brief video sequence may be challenging to execute. Without any scene information, abnormal motion, or may be patterns in the facial texture, it can be very difficult, only for humans, the system able to differentiate between a real human face and a fake face that may depend on the imaging and the quality of false face.

5.Challenge-Response:

IA challenge – Response can be challenging to detect spoofing based on Approach liveness and motion analysis when all you have is a few brief video clips to go on. This issue can be made simpler by asking the user to complete a particular task or random action (such as a smiling and moving the head to the right). Evidence of liveness will be provided by the response for the users, if any. This method of spoofing detection is known as the challenge- response strategy.

III. RESULTS AND DISCUSSION

We constructed and trained a liveness model using the Haar Cascade technique. We employed random scaling, random rotation, random brightness, and random contrast augmentations during training. Webcam data is gathered, and the Mobile-net and LBPH algorithms are used to extract features.

Fig 8: Detection and Recognition of Real and Spoofed face images of person 1 with glasses.

Features of records gathered are in comparison with functions of records trained in dataset and if there's a fantastic match, then detects as actual and if don`t match, then it detects as spoofed face.

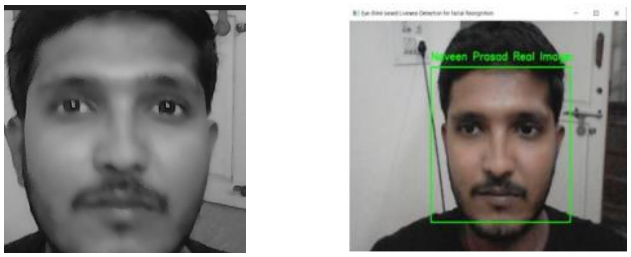


Fig 9 (a) **Fig 9 (b)**

Fig 9 (a) and 9 (b) shows the trained dataset and detected face image are a match and the same is detected as real face image.

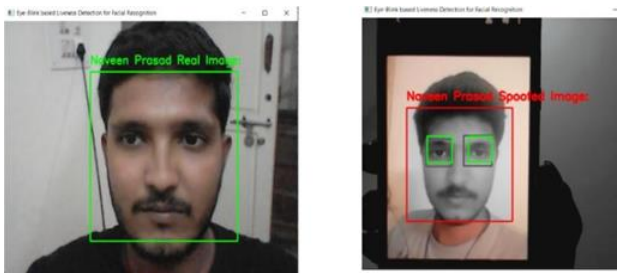


Fig 10: Detection and Recognition of Real and Spoofed face images of person 1 without glasses

Different character faces with and without glasses are gathered via web digital digicam and examined as shown in Fig 10.

Sample face images captured through web camera for training

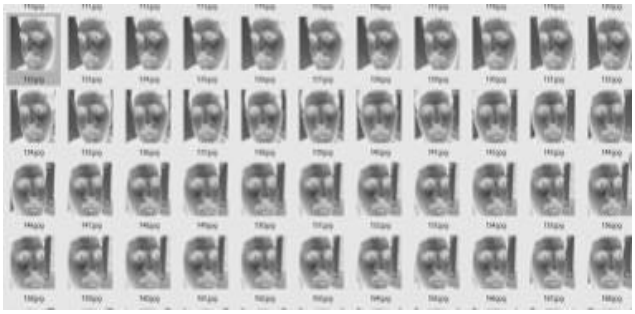


Fig 11: Sample face images captured through web-camera undergoing pre-processing

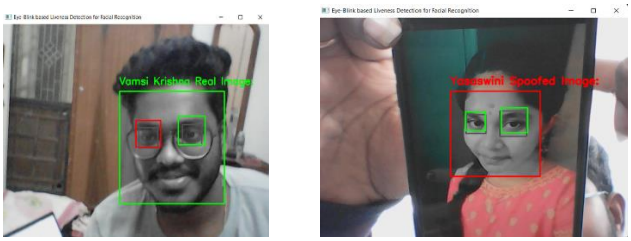


Fig 12: Various real time face images of different persons detected and recognized

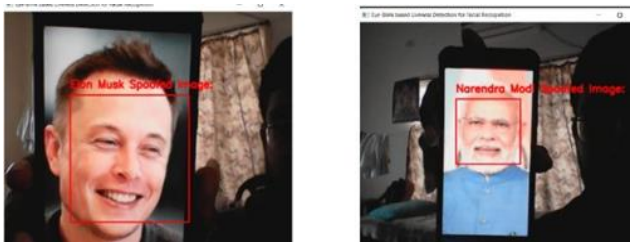


Fig 13: Detection and recognition of Spoofed images collected from internet

In Fig. 13, a few celebrities face images from the two most popular fields were collected from the internet and tested are shown.

III.1 Evaluation metrics:

The major way that motion analysis distinguishes between 2D and 3D faces is by their motion patterns. It makes use of planar things becomes difficulty from 3-D items for real human faces. Optical flow that is estimated from video sequences is typically the basis for analysis for motion. When employing motion analysis, the method exceedingly not easy for fake a 2D facial image because it is independent of texture and does not require user involvement. However, motion analysis requires video, and using motion analysis where the video has little motion activity is exceedingly challenging. The current method needs high-quality pictures and can be faked by 3D sculptures.

Texture analysis method primarily uses observable texture patterns, like print errors and general blur image, help to identify attacks. The proposed method is based on the idea where faces with fake images are printed as output on paper, and that a texture feature generated by the printing process. The face structure present in the paper able to distinguish between the printed image and real face images. To verify or identify the user, along with their user's face verified with on printed paper and placed in front of the camera. In these circumstances, texture analysis can be helpful to determine which human faces are real because printing process and paper typically method have high texture qualities. A technique based on texture analysis is simple to apply and does not require user participation. However, there might be a wide range of paper method and printing textures analysis, and systems based on texture analysis need to be resilient to various texture of patterns, which calls for the development of a wide range of datasets. Another possibility is that the attack will be conducted using a screen-displayed image, which will generate relatively little texture data.In order to overcome the dependence

on specific texture patterns, motion analysis will be helpful. However, when there is little motion data, motion analysis may run into issues. This may occur as a result of the user's potential for varied behavior, very noisy photos, and resolution with low levels. A motion analysis may be unsuccessful when spoof assaults are carried out with the use of more sophisticated techniques, such as 3D sculptural face models.

There are two different ways to detect life signs. One first expects the user will interact in a given way. When confirm the liveness for his own facial image in this scenario, then the user must complete a specific activity. This activity could be a specific motion that serves as a password or a challenge response. It is thought that real users are those who would complete their tasks accurately. The second category does not rely on user involvement and instead focuses on certain facial motions, such as blinking of eyes, and interprets these movements act as evidence of life and a real face. It is exceedingly difficult to fake a life sign-based liveness detection-based approach using 2D and 3D sculptures. This method is likewise independent of texturing; however, user participation may be required. The primary method used in this approach is face part detection. Here the formulation for calculating the detections is given:

Accuracy:

Accuracy = TP+TN / TP+FN+TN+FP

Precision:

Precision = TP/TP+FP

Recall:

Recall = TP/TP+FN

F1 score:

F1 = 2* precision* recall / precision+ recall

Classifier	Dataset	Accuracy	Precision	Recall	F1 Score
Mobile net	GitHub and Real time	98.62	98.61	98.40	98.50

Table 1: Different metrics obtained using training data in Mobile net

Classifier	Dataset	Accuracy	Precision	Recall	F1 Score
Mobile net	GitHub and Real time	96.25	95.31	96.97	96.11

Table 2: Different metrics obtained using validation data in Mobile net

CLASSIFIER	DATASET	TRAINING (%)				VALIDATION (%)			
		ACC	PRE	REC	F1	ACC	PRE	REC	F1
CNN	GITHUB REAL TIME	93.86	93.80	93.65	93.58	95.41	94.29	95.91	95.32
MOBILE-NET	GITHUB REAL TIME	98.62	98.61	98.40	98.50	96.25	95.31	96.97	96.11

Table 3: Comparison of different metrics obtained between existing system and proposed system.

We can identify live/spoof in face recognition systems using this liveness detection, which also allows us to detect live faces in photos, videos, and real people. With the help of our Haar Cascade method, the system's performance was enhanced, and the validation set's observed liveness detector achieved 99% overall accuracy. The graph below displays the evaluation of the findings based on our dataset.

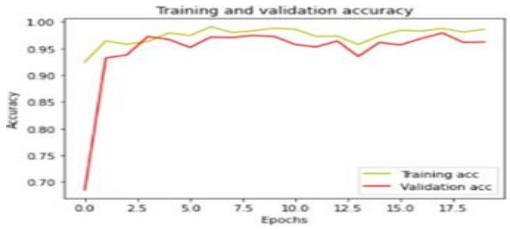


Fig 14: Training parameter and Validation of Accuracy observed in Mobile-net model

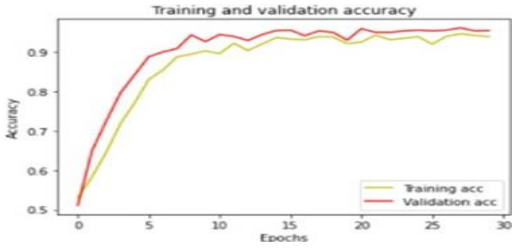


Fig 15: Training and Validation Accuracy observed in CNN analysis model.

Fig 13: Training and Validation loss observed in Mobile- net model

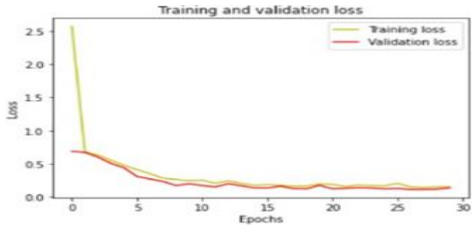


Fig 14: Training and Validation loss observed in CNN analysis model

A combination process for datasets results in increased parameter like variance value in both the genuine and fake classes because of distinctive properties observed in each dataset, including lighting, about the spoofing medium analysis, the setting, and the recording equipment.

A greater variety of features can be learned by the model as a result, enabling it to differentiate between actual and attack classes. The generalization potential of handcrafted features was assessed in the context of Face Presentation Attack Detection by combining additional datasets (FPAD). According to the study's findings, cutting-edge techniques with outstanding performance of intra-dataset are less in generalizable form, where in cross-dataset evaluation method that combined with sources of heterogeneous. Their performance is influenced by a number of variables, including the image quality, display settings, and capture equipment. The tests in this study, in contrast to that evaluation, binary classification uses with four previously trained method for deep neural networks, that identify PAs. Yet deep learning frameworks cannot generalize to varied distributions, as was seen from the analysis of experimental results.

IV. CONCLUSION AND FUTURE SCOPE

This work provides details on face anti-spoofing utilizing deep learning methodology. Our goal is to demonstrate that deep learning technique is an effective technique to identify between real and spoofed faces with better results than any other techniques utilized, despite the fact that there are several research and strategies employed in the face anti-spoofing field. In the realm of computer vision, the conventional convolutional neural network (CNN) is heavily utilized for its output. But as the use of advanced technology increases, hackers are become more difficult to control. Traditional CNN is insufficient in such situations to increase security. Therefore, we present mobile net, a modernized version of CNN. Mobile-net analysis model and a liveliness module work together to identify, recognize, and differentiate between authentic and fake facial photos in order to block unwanted access. Mobile-net employs depth-wise separable convolutions and point-wise convolutions to boost system efficiency and boost precision. By assigning a unique identifier to each face in the dataset, the Local Binary Pattern histogram (LBP) is utilized to recognize faces. Every time a face is photographed, LBPH assigns it an ID. For quicker classification of various faces with fewer parameters and a smaller network, mobile-net analysis is utilized. Mobile-net model and liveliness module are integrated for anti-spoofing. We utilize the eye-blinking module since it works well and is easy to use. Python is used to create the eye-blinking module. Faces that have been analyzed by the mobile-net model are transferred to the liveliness module, where the amount of eye blinking indicates if the face displayed is authentic or a fake in order to prevent unauthorized access.

V. FUTURE SCOPE

There is research being done on novel methods. However, by incorporating any new anti- spoofing approaches, it is feasible to create better and more effective outcomes. The suggested work uses deep learning techniques, which have a great deal of potential to detect, recognize, and distinguish between authentic and spoofed facial photographs. Parallel programming approaches that speed up face

recognition and anti- spoofing systems can be explored in more detail.

VI. REFERENCES

[1]. R. B. Hadiprakoso, H. Setiawan and Girinoto ,"Face Anti-Spoofing Using CNN Classifier", 2020 3rd International Conference on Information and Communications Technology (ICOIACT),pp. 143-147, 2020, doi: 10.1109/ICOIACT50329.2020.9331977.

[2]. Raden Budiarto Hadiprakoso, "Face Anti-Spoofing Method with Blinking Eye and HSV Texture Analysis", 2020 IOP Conference Series: Materials Science and Engineering, Volume1007,3rd Tarumanagara International Conference of the Applications of Technology and Engineering (TICATE) 2020 Jakarta, Indonesia.

[3]. Gency V, Mrs. Chaithanya C, Mrs. Aysha Fymin Majeed, "Face Spoofing Detection: A Survey on Different Methodologies" 2020 International Research Journal of Engineering and Technology (IRJET); Volume: 07 Issue: 12

[4]. Zuheng Ming, Muriel Visani,Muhammad Muzzamil Luqman,Jean- Christophe Burie, "A Survey on Anti-Spoofing Methods for Facial Recognition with RGB Cameras of Generic Consumer Devices", 2020 Journal of Imaging .

[5]. Hsueh-Yi Sean L "Convolutional Neural Networks for Face Anti- Spoofing and Liveness Detection", 2019 6th International Conference on Systems and Informatics (ICSAI), 2019, doi: 10.1109/ICSAI48974.2019.9010495.

[6]. Karuna Grover, Rajesh Mehra, "Face Spoofing Detection using Enhanced Local Binary Pattern", 2019 International Journal of Engineering and Advanced Technology (IJEAT) ISSN: 2249-8958 (Online), Volume-9 Issue-2.

[7]. Md Rezwan Hasan University of kent; S M Hasan Mahmud, "Face Anti-Spoofing Using Texture-Based Techniques and Filtering Methods", 2019 IEEE,Conference: 3rd International Conference on Machine Vision and Information Technology (CMVIT 2019) At: Guangzhou, China.

[8]. Samrity Saini; Kiranpreet Kaur, "KNN Classification for the Face Spoof Detection", 2019 International Journal of Scientific & Engineering Research Volume 10, Issue 3, March-2019; ISSN 2229-5518;

[9]. Shaimaa Mohamed; Amr S. Ghoneim, "Visible/Infrared face spoofing detection using texture descriptors", 2019 MATEC Web of Conferences 292(2):04006 DOI:10.1051/matecconf/201929204006

[10]. Arida Kartika , Indra Bayu Kusuma et al, "Image Spoofing Detection Using Local Binary Pattern and Local Binary Pattern Variance",2018 International Journal on ICT Vol. 4, Issue. 2, December 2018. pp. 11-18,doi:10.21108/indojc.2018.42.13

[11]. Sakshi Jha, Mtech Scholar; Dr. Neetu Sharma Associate Professor, Computer Science & Engineering, Ganga Institute of Technology and Management Kablana,Jhajjar, Haryana., "Approach for Face Spoof Detection Using KNNclassifier", 2018 International Journal of Creative Research Thoughts (IJCRT)

[12]. P.Kavitha et al, "A Study on Spoofing Face Detection System", 2017 International Journal of Pure and Applied Mathematics Volume 117 No. 22 , 205-208,ISSN: 1311-8080 (printed version) ISSN: 1314-3395

Cite this article as :

T Naveen Prasad, Dr. B. Anuradha, "Face Anti-spoofing and Liveliness detection using Mobile net and Haar cascade Algorithm", International Journal of Scientific Research in Science and Technology (IJSRST), Online ISSN : 2395-602X, Print ISSN : 2395-6011, Volume 10 Issue 2, pp. 710-723, March -April 2023. Available at doi : https://doi.org/10.32628/IJSRST523102103

Journal URL : https://ijsrst.com/IJSRST523102103