Real-time Liveness Detection: Ensuring the Integrity of Facial Recognition Systems

Course Project

Signals, Image, and Video

Master's degree in Artificial Intelligence Systems

A Course Project Submitted By

Sr #	Student Name	Reg. No.
01	Adnan Irshad	241763
02	Hira Afzal	241351

To

Prof. Francesco De Natale
and **Prof. Andrea Rosani**

UNIVERSITY OF TRENTO

Department of Information Engineering and Computer Science

Table of Contents

Table of Figures

Table of Tables

Real-time Liveness Detection: Ensuring the Integrity of Facial Recognition Systems

Abstract

In the face recognition system, one of the common problems that arise are presentation attacks or spoofing attacks, which raises questions about the performance and reliability of face recognition-based systems for ensuring access & security. To tackle this issue, this project proposes an extra security layer of face anti-spoofing in the face recognition process through the implementation of face liveness detection in real time. The proposed system uses two anti-spoofing techniques: Motion analysis by eyes blinks detection & mouth movement/speaking detection and color texture analysis by color models techniques. Motion analysis approach is suitable for detecting only image-based spoofing attacks, but it fails in video-based spoofing attacks. Therefore, along with motion analysis approach, texture analysis approach by using colorspaces technique is used to ensure more robust anti-spoofing protection against mobile image, mobile replay, and printed image spoofing attacks. The proposed system uses Eyes/Mouth movements detection along with the YCrCb and CIEluv color spaces to detect face liveness and then combines both histograms of these color spaces into a single feature set, which is used as input to a support vector machine (SVM) classifier for liveness prediction.

1. Introduction

In the past two decades, the rise of deep learning has resulted in tremendous improvements in the performance of face recognition systems, such as DeepFace, DeepIDs, VGG Face, FaceNet, SphereFace, and ArcFace. These systems have surpassed human-level accuracy on challenging face benchmarks and are widely used in various applications, such as online payments, e-commerce security, smartphone-based authentication, and border control.

Despite the popularity of face recognition systems, they have become the primary targets of Presentation Attacks (PAs) or Spoofing attacks that affect the performance and reliability of face recognition-based systems. Spoofing attacks can be performed using various methods, such as using a print photo attack, mobile photo attack, replay attack, photo mask attack and 3D mask attack. To ensure the security of face recognition systems, it is essential to implement an anti-spoofing technique.

One of the popular anti-spoofing techniques is face liveness detection. Face liveness detection is the process of detecting whether a face in an image or video being presented for recognition is live or if it is a fake representation such as a mobile picture or a video or a printed picture. This is important in facial recognition systems because it helps to prevent presentation attacks or spoofing, which is when someone tries to bypass the system by presenting a fake face, such as a picture or a video, instead of their real face.

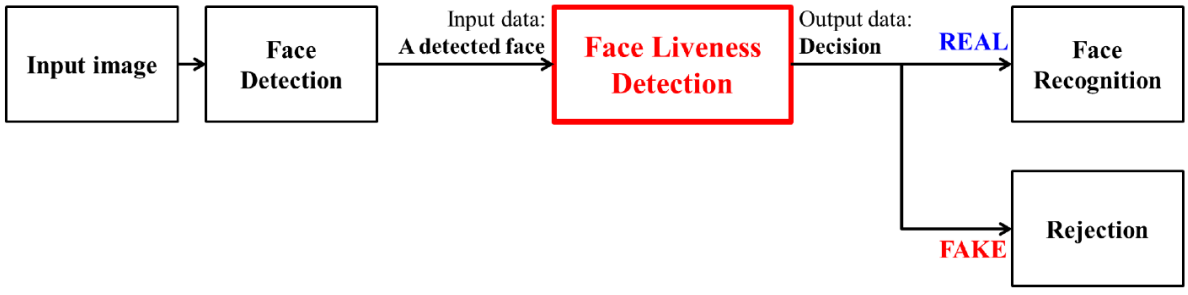


Figure 1 Face Liveness Detection Process

There are several approaches to implement face liveness in facial recognition systems, including:

- **Color and texture analysis:** This approach involves analyzing the color and texture of the face to determine if it is a live face. The idea is that live faces have more nuanced color and texture than fake faces, and these differences can be used to distinguish between the two.
- **Motion analysis:** This approach involves analyzing the motion of the face to determine if it is a live face. For example, a live face will typically exhibit subtle movements such as blinking, facial expressions, and head movements, which can be used to distinguish it from a static picture or video.
- **Infrared analysis:** This approach involves using infrared technology to detect subtle differences in temperature between a live face and a fake representation. A live face will have a higher temperature than a fake face, and this difference can be used to distinguish between the two.
- **Depth analysis:** This approach involves analyzing the depth of the face to determine if it is a live face. For example, a live face will typically have a greater depth than a flat picture or video, and this difference can be used to distinguish between the two.

In this project, face liveness detection is implemented as an extra security layer in the face recognition process by combining motion analysis and color texture analysis approaches. The proposed system uses two anti-spoofing techniques: Eye Blink Detection as part of motion analysis approach and Color Space Histogram Concatenation as part of color texture analysis approach to distinguish between real/live faces and faked/spoofed faces.

2. Proposed System

The problem with the traditional FR system is, they can be cheated by presentation attacks. To minimize such problems an extra security layer of face liveness must be added to the current facial recognition systems.

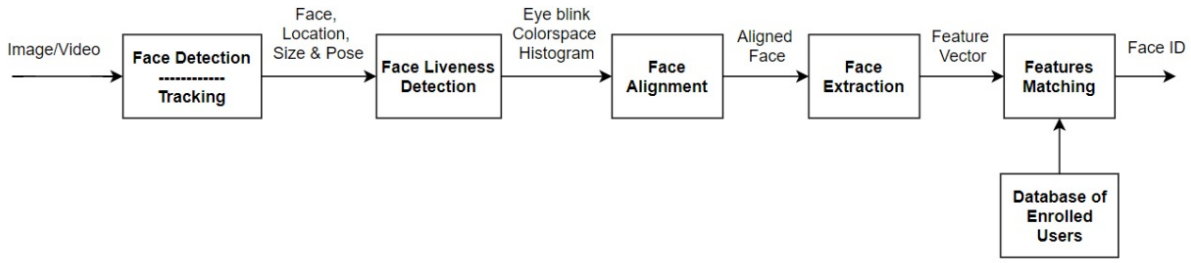


Figure 2 Facial Recognition System with Liveness Detection Layer

In this project we are focusing on liveness detection, we'll be treating liveness detection as a multiclass classification problem. Given an input image/frame from webcam, detect the face, resize it, detect the facial landmarks, convert into color spaces like YCrCb and CIE Luv*, get histograms concatenation and train a classifier capable of distinguishing whether current face is live, or replay attack or printed attack.

2.1 Color Texture Analysis

The Color and Texture Analysis approach for face liveness detection involves analyzing the color and texture properties of a face to determine if it is a live face or a fake representation (such as a photo or a video). This approach is based on the premise that live faces have unique color and texture properties that can be used to differentiate them from fake representations.

In this project, color spaces (YCrCb and CIE Luv*) histograms concatenation approach is used to detect live or spoofed face. In these color spaces, different color information is separated and transformed into a different representation than the RGB color space. The color information in YCrCb is represented in luminance (Y) and chrominance (Cr and Cb) channels, while in CIE Luv*, the color information is represented in a perceptually uniform color space. In both color schemes the components 'Y' and 'l' represent luma component of the colour and other Cr, Cb, u and v represent chroma components. Luma

components always have greater light intensity as compared to other components in printed and mobile images and videos. For real/live faces, the color information remains consistent and uniform, while in a fake face, it contains artifacts or inconsistencies as shown in following figure 3 of live face, printed face and replay face images.

Figure 3 YCrCb and CIE Luv Colorspace Representation of Real, Printed and Replay Face

Further, insights can be get by looking at the histograms graphs of each case, visualizing each component Y, Cr, Cb, L, u and v frequencies in figure 4 and figure 5.

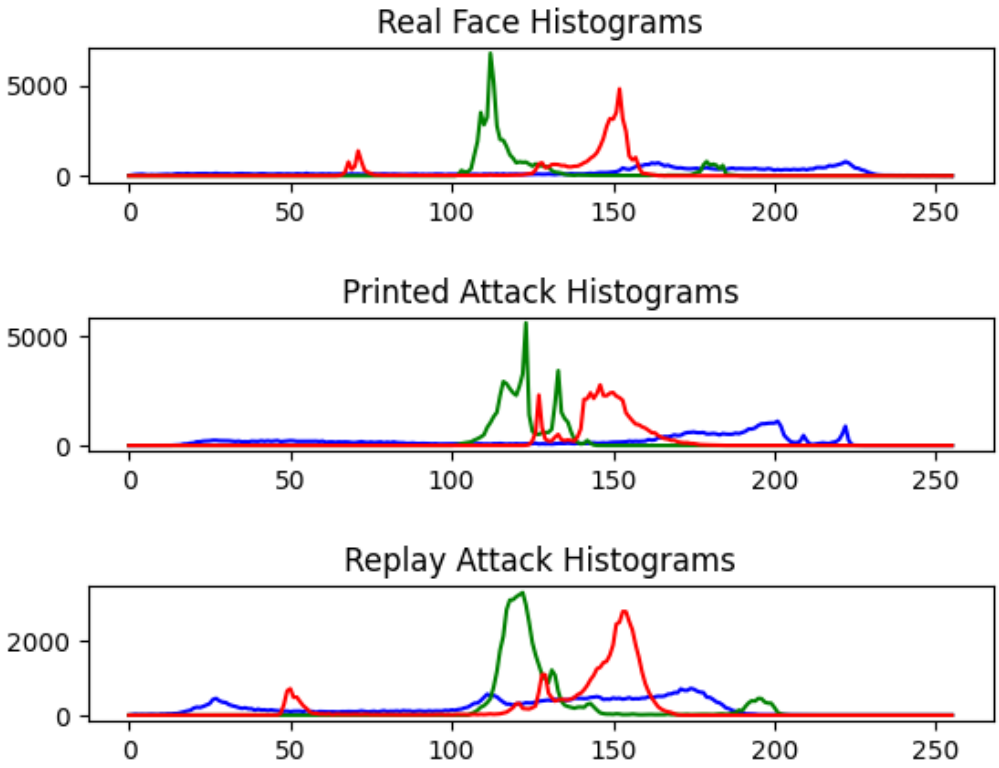


Figure 4 YCrCb Colorspace representation of Live, Printed and Replay Face

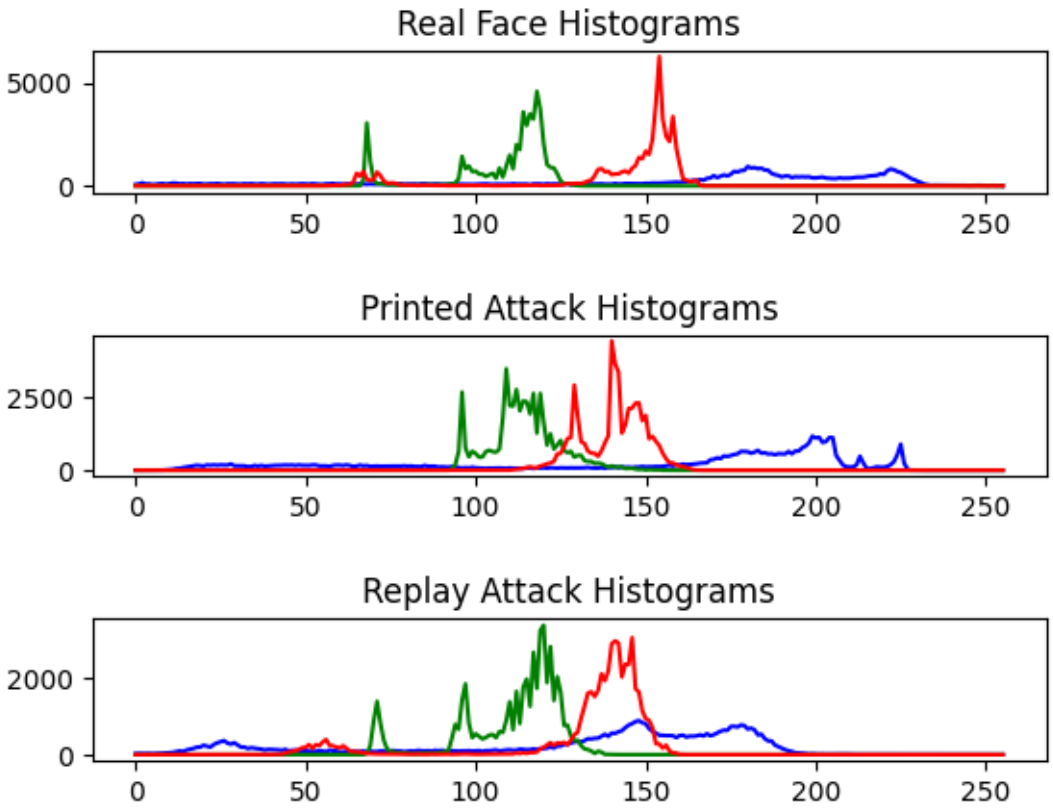
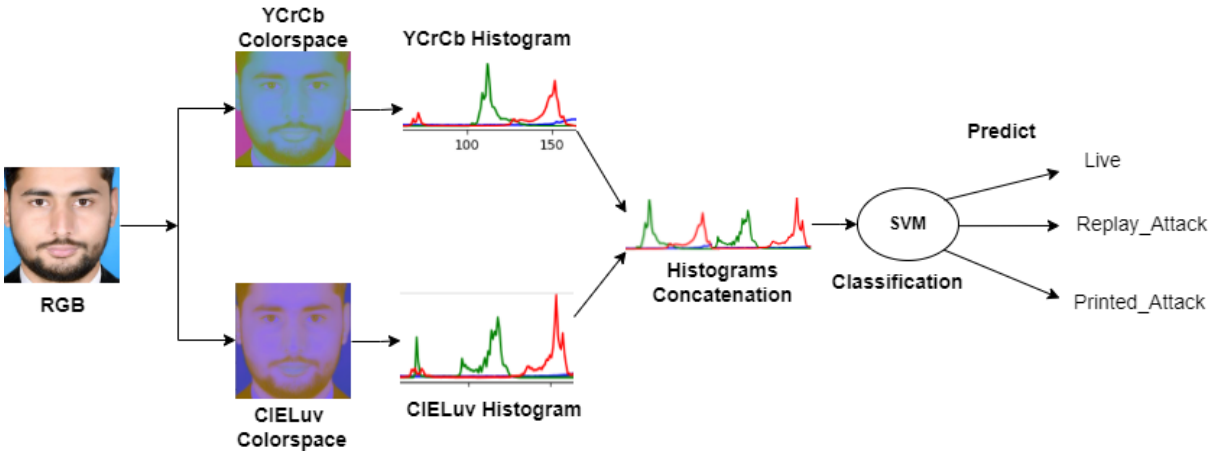


Figure 5 CIE Luv Colorspace representation of Live, Printed and Replay Face

By observing color analysis for real and fake images, created a custom dataset, converted the images to the YCrCb and CIELuv color spaces, calculated histograms (1 for each component of YCrCb and

CIELuv color spaces) and concatenated them into a single feature vector of size 1536. Finally used SVM Classifier to train the classifier on the feature vectors extracted from the images.



2.2 Motion Analysis

In the motion analysis approach, eye blink and mouth/jaws open/close movements are used as cues to determine whether a face is live or not. The idea behind this approach is that a live face should show dynamic movements, such as eye blinks or mouth open/close/speaking movements, while a fake face, such as a photograph or a mask, should remain static. To detect eye blinks, the system can analyze changes in eye shape, eye closure duration, and the frequency of eye blinks. Similarly, the system can analyze changes in the shape of the mouth region, such as lip movements, to detect mouth open/close movements.

To detect eyes blinks & jaws movements in proposed system, the system detects the human face and then uses mediapipe to obtain the facial landmarks. Then using the ayes 6 landmarks position, and calculates the eye aspect ratio (EAR) by dividing the distance between the eyes by the width of the face using the formula written below and image shown below:

$$\text{EAR} = \frac{||p2-p6|| + ||p3-p5||}{2||p1-p4||}$$

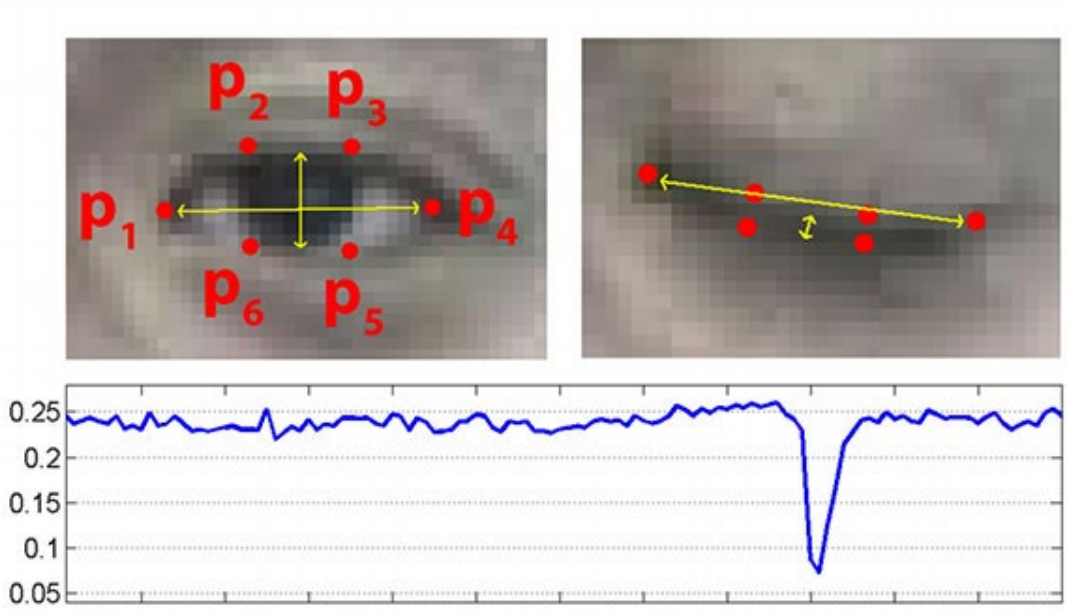


Figure 6 Eye Aspect Ratio Formula

While the mouth aspect ratio (MAR), like EAR goes up when there are movements in jaws like open the mouth or speak.

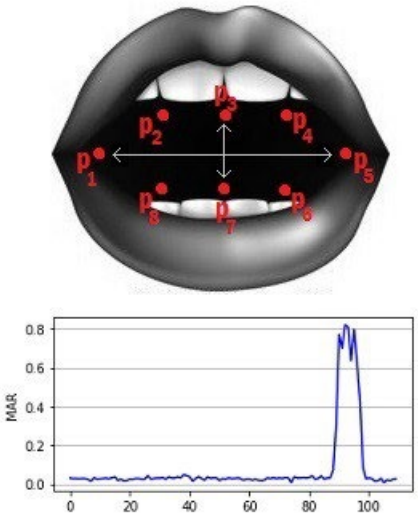


Figure 7 Mouth Aspect Ratio Formula

The system then uses these ratios to determine if the eyes are closed or open and if the mouth is open or closed. If the eye aspect ratio is lower than a certain threshold, it indicates that the eyes are closed, and if the mouth aspect ratio is higher than a certain threshold, it indicates that the mouth is open. These movements, along with color information from the color spaces, are then used to determine the liveness of the face.

3. Dataset

Since majority databases like CASIA Face Anti-Spoofing and Replay-Attack datasets were private and needed access to train & evaluate classifier, we created our own custom dataset by taking 15 to 20 seconds of videos with different poses and lightening conditions of each case including live face, live replay attack, mobile photo attack, printed image and photo of printed image. Apart from this we also gathered some publicly available datasets as well to prevent bias datasets. After gathering dataset and recording our own videos, we performed preprocessing operations including converting videos into frames, extracting face from frames & cropping faces, and splitting into training and testing datasets. We trained our face liveness classifier on 5957 training images and 1491 testing images. There were a total of three classes including 2555 images of printed attack, 757 images of replay attack and 2645 images of real/live faces.

Figure 8 Custom Dataset Structure

4. Implementation Methodology

The proposed system for liveness detection is implemented in Python3. After gathering & preparing dataset, its development divided into two parts, training liveness detection model and testing model in real time webcam.

In training the liveness classifier, following steps were performed:

1. Loading the Dataset
2. Preprocessing: Preprocesses dataset by extracting, cropping, and resizing face into 224 x 224.
3. Color space conversion: Converted each face image from the RGB color space to both the Y CrCb and CIE Luv* color spaces.
4. Feature extraction: Calculated six histograms corresponding to each component of the Y CrCb and CIE Luv* color spaces and concatenate them into a feature vector.
5. Model training: Trained an SVM classifier using the extracted features and labels.
6. Model evaluation: Evaluate the performance of the model using metrics such as accuracy, precision, recall, and F1-score, confusion matrix and classification report.
7. Saved the classifier in pickle format.

After training and evaluating classifier for liveness detection, used the trained classifier and tested it in real time via web cam streaming.

1. Captured web cam live streaming using OpenCV.
2. Detect face using Mediapipe library.
3. Check Face Liveness by eye & mouth movement using facial landmarks.
    a. Get facial landmarks of detected face.
    b. Get eyes landmarks indexes and calculate the EAR.
    c. If EAR is less than a certain threshold point count this as eyes blinking.
    d. Get lips landmarks indexes and calculate the MAR.
    e. If MAR is greater than certain threshold point, count this as mouth open.
    f. If either of the eyes blinking or mouth movements occurs, then go to next liveness check if no movements occur then mark the current face as spoofed attack in step 5.
4. If liveness occurred in step 4 then check liveness using color texture analysis approach.
    a. Convert face into Y CrCb.
    b. Convert face into CIE luv.
    c. Calculate histogram of Y CrCb face
    d. Calculate histogram of CIE luv face
    e. Concatenate calculated histograms & convert into features.
    f. Predict liveness class from trained classifier as Live, Printed Attack or Replay Attack
5. Show the liveness results.

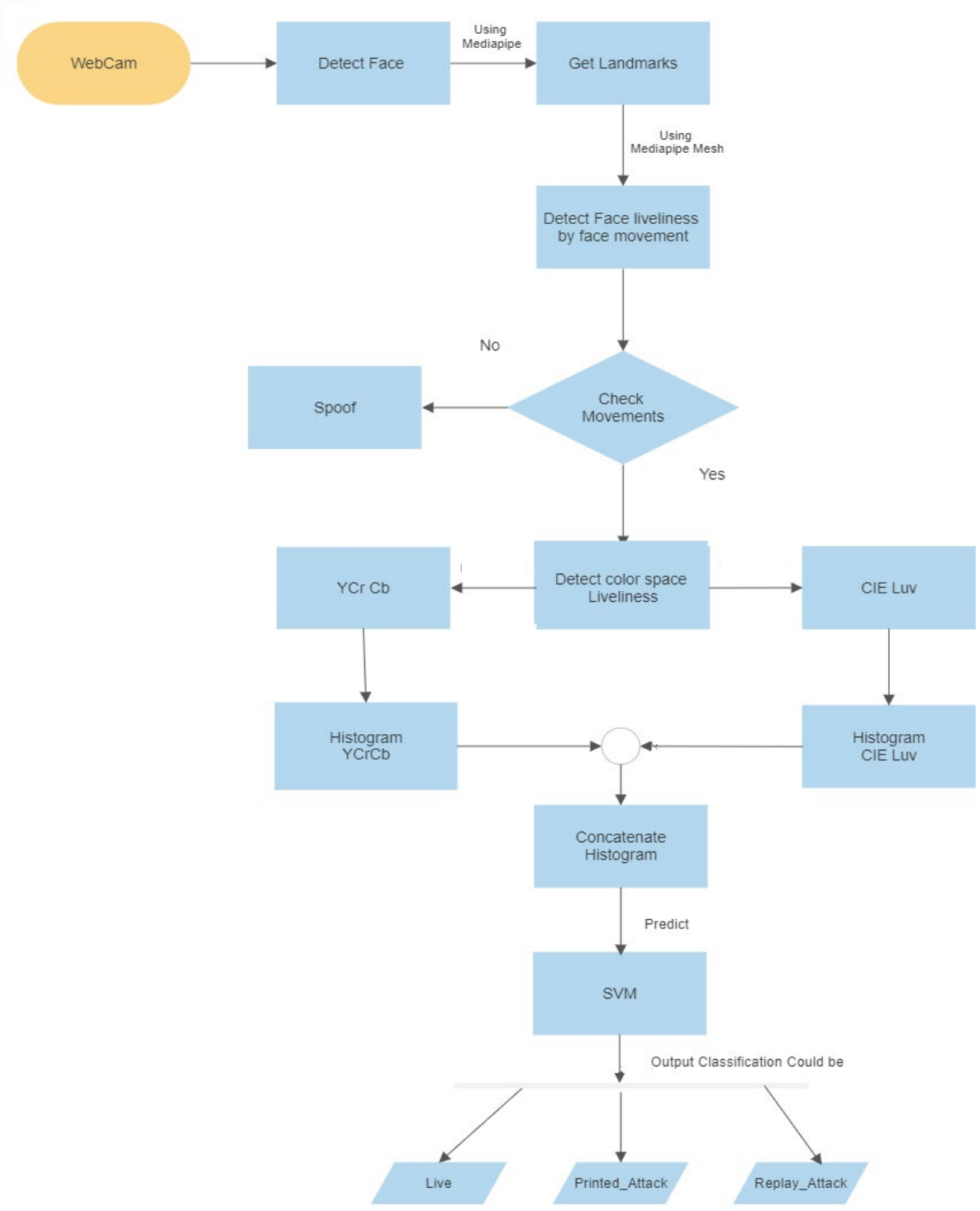


Figure 9 Proposed Face Liveness Detection Flow Diagram

5. Results

As above (at dataset section) we let you that we created our own dataset i.e., 15 to 20 seconds videos with different poses, 2555 printed images, 757 replay attack images and 2645 real\live faces. After training the model we performed the test cases on the model and evaluated its results by using the metrics parameters.ie., accuracy, precision, recall, and F1-score, confusion matrix, and classification report, results are mentioned in the below figures and table.

Classification Report	Precision	Recall	F1_score
Live	0.93	0.89	0.91
Printed_Attack	0.86	0.94	0.90
Replay_Attack	0.97	0.85	0.90
Accuracy	-	-	0.90
Macro_avg	0.92	0.89	0.90
Weight_avg	0.91	0.90	0.90

Table 1 Classifier Evaluation Results

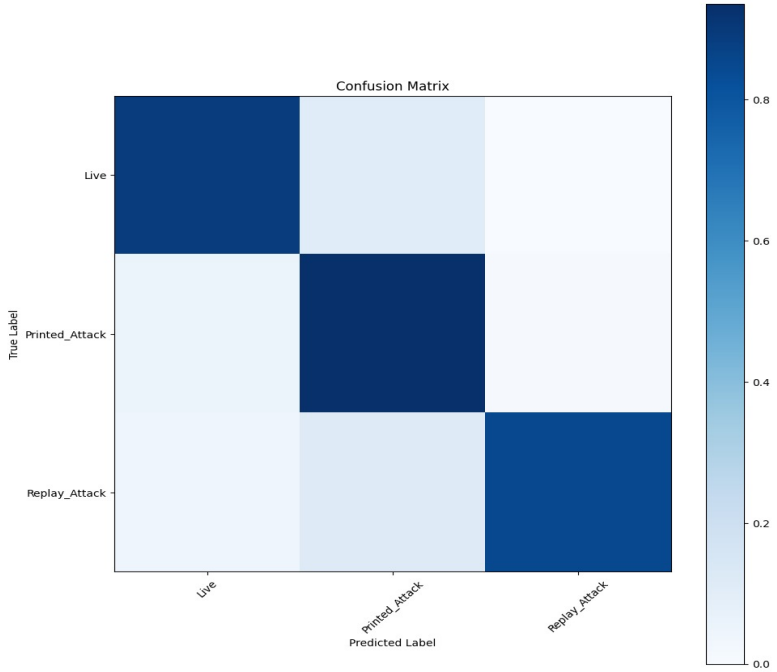


Figure 10 Classification Confusion Matrix

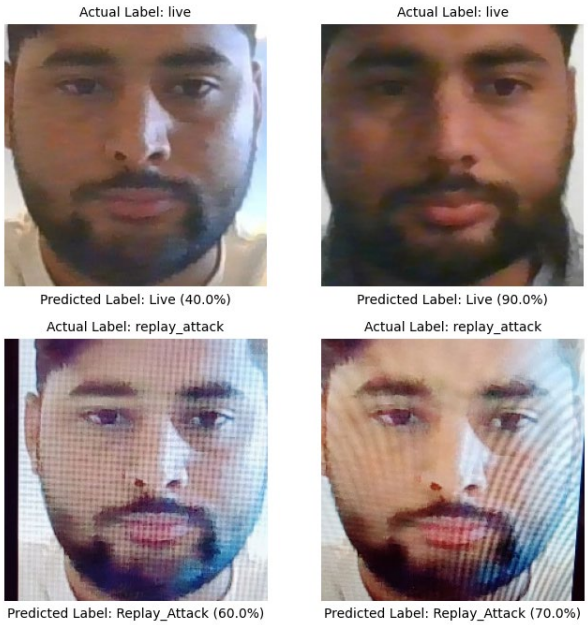


Figure 11 Predictions on Unseen Faces

6. Conclusion

Our Project aims to address the problem of spoofing attacks in face recognition systems by proposing an extra security layer of face anti-spoofing through the implementation of real-time face liveness detection. This model uses two anti-spoofing techniques: Eye Blink Detection and Color Space Histogram Concatenation, which are combined to provide robust anti-spoofing protection. The performance of the model is evaluated using metrics such as accuracy, precision, recall, and F1-score, confusion matrix, and classification report. The model tested in real-time using a webcam, having a limitation, its accuracy could be vary according to the lighting condition. The model can detect both image and video-based spoofing attacks. Overall, this project provides a practical solution to the problem of face anti-spoofing in face recognition systems, which is essential for ensuring access and security.

References

[1] I. R. a. S. K. Youngjun Moon, "Face Antispoofing Method Using Color Texture," *Security and Communication Networks.*

[2] a. K. N. R. F Sthevanie, "Spoofing detection on facial images recognition using LBP and," in *International Conference on Data and Information Science*.

[3] a. P. Budigem Sailavanya, "Face Recognition and Spoofing Detection System Adapted," *IJRECE VOL. 6 ISSUE 3 JULY-SEPT 2018.*

[4] R. A. G. S. C.Anuradha, "Liveness Detection with Opencv," *International Journal of Innovative Technology and Exploring Engineering (IJITEE).*

[5] J. K. A. H. Zinelabidine Boulkenafet, "FACE ANTI-SPOOFING BASED ON COLOR TEXTURE ANALYSIS," *IEEE.*

[6] X. Z. Y. Z. H. W. F. Y. Jingjing Li, "FACE LIVENESS DETECTION BASED ON MULTIPLE FEATURE DESCRIPTORS," *IEEE,* 2019.

[7] A. S. ,. a. A. R. Valter Costa, "Image-Based Object Spoofing Detection," *Springer,* 2018.

[8] Mouse Cursor Control Using Facial Movements — An HCI Application | by Akshay L Chandra | Towards Data Science

[9] CelebA Spoof For Face AntiSpoofing | Kaggle Dataset

[10] Anti-Spoofing dataset: live, replay, printouts | Kaggle Dataset