iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 13 **Issue:** IV **Month of publication:** April 2025

DOI: https://doi.org/10.22214/ijraset.2025.69121

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 13 Issue IV Apr 2025- Available at www.ijraset.com

Email Security using Facial Recognition

Gangalakshmi P[1], Swetha S[2], Sharmila B[3], Mrs. G. Sumathi[4]
[1,2,3]*B.Sc. CS – III, Kalasalingam Academy of Research and Education, TN, India*
[4]*Assistant Professor, Kalasalingam Academy of Research and Education, TN, India*

Abstract: ***The project aims to bolster email security by employing advanced facial recognition technology. Leveraging OpenCV and convolutional neural networks (CNNs), the system verifies the identity of users attempting to access email accounts by analyzing facial biometrics. This approach enhances authentication measures, mitigates the risks of unauthorized access, and strengthens overall email privacy and confidentiality. Through real-time facial recognition algorithms, the system ensures secure access to email accounts, providing a robust defense against potential security breaches.***
Keywords: Face Recognition, OpenCV, Tkinter, Webcam

I. INTRODUCTION

In the digital era, email security is a critical concern, as unauthorized access to personal and corporate emails can lead to data breaches and identity theft. This project aims to enhance email security using face recognition technology, leveraging OpenCV, TensorFlow, and Convolutional Neural Networks (CNNs). The system captures facial images of users and trains a deep learning model to recognize authorized individuals accurately. When a user attempts to access their email, the system detects their face, extracts facial features, and compares them with the trained database. If a match is found, the person's name and an accuracy score are displayed, granting access only if the confidence level meets a predefined threshold. If no match occurs, the system classifies the individual as an unauthorized person, denying access to the email. The face recognition model follows a structured pipeline, including image preprocessing, feature extraction, and classification using CNNs. OpenCV handles real-time image acquisition, while TensorFlow and CNNs perform deep learning-based classification. The trained model continuously improves with additional user data, ensuring higher accuracy over time. The system integrates with email services, restricting access to only verified individuals, thereby preventing unauthorized entry. This project provides a robust, biometric-based security layer for email authentication, enhancing privacy and reducing the risk of email-based cyber threats.

II. THE DEVELOPMENT STAGE OF FACIAL RECOGNITION

A. Early Algorithm Stage

1) OpenCV

OpenCV was started at Intel in the year 1999 by Gary Bradsky. The first release came a little later in the year 2000. OpenCV essentially stands for Open Source Computer Vision Library. Although it is written in optimized C/C++, it has interfaces for Python and Java along with C++. OpenCV boasts of an active user base all over the world with its use increasing day by day due to the surge in computer vision applications.OpenCV-Python is the python API for OpenCV. It is a python wrapper around the C++ implementation of OpenCV. OpenCV-Python is not only fast (since the background consists of code written in C/C++) but is also easy to code and deploy (due to the Python wrapper in foreground). This makes it a great choice to perform computationally intensive programs.

2) Images as Arrays

An image is nothing but a standard NumPy array containing pixels of data points. More the number of pixels in an image, the better is its resolution. When we think of pixels it has to be tiny blocks of information arranged in the form of a 2 D grid, and the depth of a pixel refers to the colour information present in it. In order to be processed by a computer, an image needs to be converted into a binary form. The colour of an image can be calculated as follows:

Number of colours/ shades = 2^bpp where bpp represents bits per pixel.

Naturally, more the number of bits/pixels, more possible colours in the images. The following table shows the relationship more clearly.

a) *Binary Image:* A binary image consists of 1 bit/pixel and so can have only two possible colors, i.e., black or white. Black is represented by the value 0 while 1 represents white.

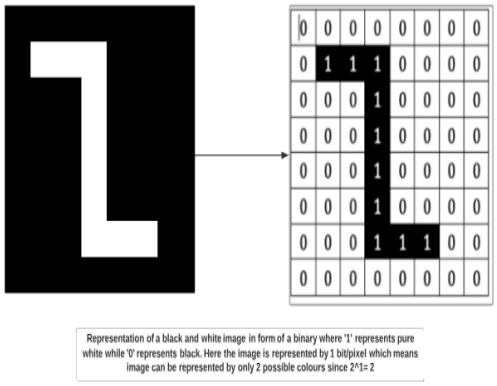


Figure 1-Binary Image

3) *Grayscale Image:* A grayscale image consists of 8 bits per pixel. This means it can have 256 different shades where 0 pixels will represent black color while 255 denotes white. For example, the image below shows a grayscale image represented in the form of an array. A grayscale image has only 1 channel where the channel represents dimension.

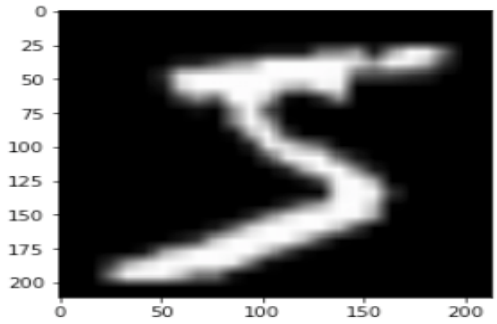


Figure 2-Grayscale Image

4) *Colored Image:* Colored images are represented as a combination of Red, Blue, and Green, and all the other colors can be achieved by mixing these primary colors in correct proportions.

B. Artificial Features and Classifier Stage

1) 'Haar features' extraction

After the tremendous amount of training data (in the form of images) is fed into the system, the classifier begins by extracting Haar features from each image. Haar Features are kind of convolution kernels which primarily detect whether a suitable feature is present on an image or not.

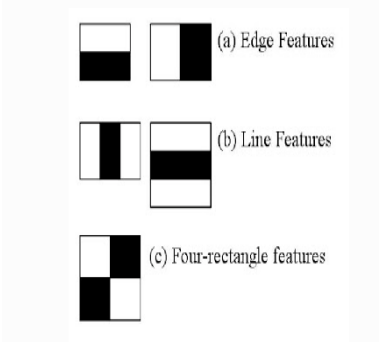


Figure 3-Haar Features

These Haar Features are like windows and are placed upon images to compute a single feature. The feature is essentially a single value obtained by subtracting the sum of the pixels under the white region and that under the black.

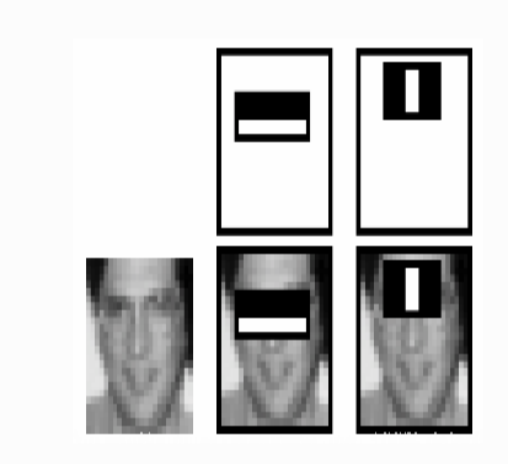


Figure 4-Haar Process on Image

For demonstration purpose, let us say we are only extracting two features, hence we have only two windows here. The first feature relies on the point that the eye region is darker than the adjacent cheeks and nose region. The second feature focuses on the fact that eyes are kind of darker as compared to the bridge of the nose. Thus, when the feature window moves over the eyes, it will calculate a single value. This value will then be compared to some threshold and if it passes that it will conclude that there is an edge here or some positive feature.

2) 'Integral Images' Concept: The algorithm proposed by Viola Jones uses a 24X24 base window size, and that would result in more than 180,000 features being calculated in this window. Imagine calculating the pixel difference for all the features? The solution devised for this computationally intensive process is to go for the Integral Image concept. The integral image means that to find the sum of all pixels under any rectangle, we simply need the four corner values.

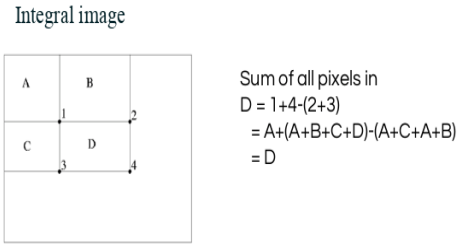


Figure 5- Integral Image

This means, to calculate the sum of pixels in any feature window, we do not need to sum them up individually. All we need is to calculate the integral image using the 4 corner values. The example below will make the process transparent.

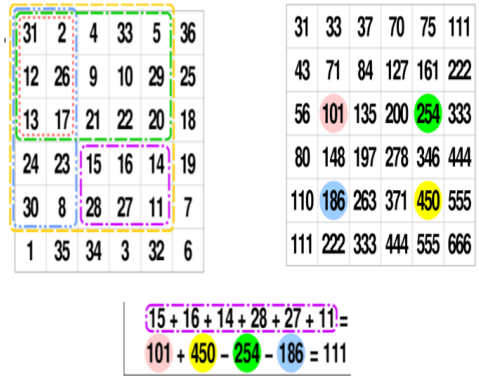


Figure 6-Sum of Pixels

3) 'Adaboost' : to improve classifier accuracy

As pointed out above, more than 180,000 features values result within a 24X24 window. However, not all features are useful for identifying a face. To only select the best feature out of the entire chunk, a machine learning algorithm called Adaboost is used. What it essentially does is that it selects only those features that help to improve the classifier accuracy. It does so by constructing a strong classifier which is a linear combination of a number of weak classifiers. This reduces the number of features drastically to around 6000 from around 180,000.

4) Using 'Cascade of Classifiers'

Another way by which Viola Jones ensured that the algorithm performs fast is by employing a cascade of classifiers. The cascade classifier essentially consists of stages where each stage consists of a strong classifier. This is beneficial since it eliminates the need to apply all features at once on a window. Rather, it groups the features into separate sub-windows and the classifier at each stage determines whether or not the sub-window is a face. In case it is not, the sub-window is disLockerded along with the features in that window. If the sub-window moves past the classifier, it continues to the next stage where the second stage of features is applied. The process can be understood with the help of the diagram below.

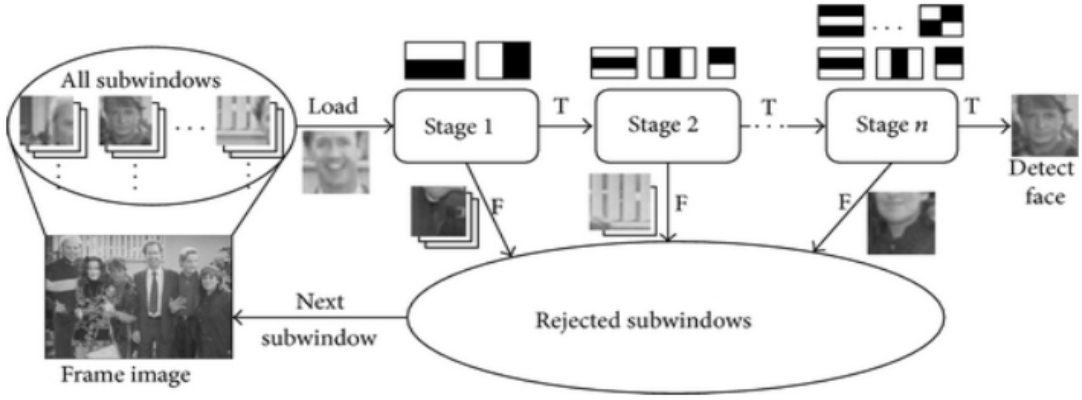


Figure 7-Cascade of Classifiers

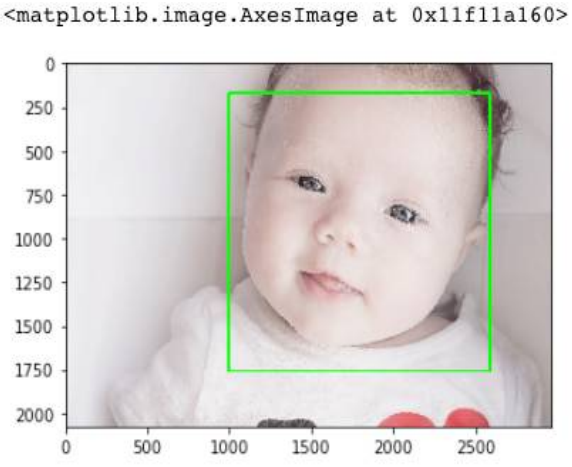


Figure 8-Face Detection using Frame

III. EASE OF USE

The purpose of this project is to overcome the increasing threats of cyberattacks and unauthorized access to personal and corporate emails, biometric authentication methods offer a more secure alternative to traditional password-based logins. This project proposes an advanced email security system using face recognition, integrating OpenCV, TensorFlow, and Convolutional Neural Networks (CNNs).

The system captures facial images, processes them using deep learning techniques, and trains a model to identify authorized users accurately. When a user attempts to access their email, the system detects and matches their face with the stored database, displaying the person's name and an accuracy score. If the detected face meets the predefined accuracy threshold, access is granted. Otherwise, the system flags the user as an unauthorized person and restricts email access, ensuring a secure authentication process.

The project follows a structured workflow, including image preprocessing, feature extraction, model training, and real-time face detection for authentication. OpenCV is used for image acquisition and preprocessing, while TensorFlow and CNNs handle deep learning-based classification. The system continuously improves its accuracy with more training data, making it adaptive to different lighting conditions, facial expressions, and variations. This face recognition-based email security system provides a robust, biometric-driven authentication mechanism, reducing the risks associated with stolen passwords and unauthorized access, ultimately enhancing data privacy and cybersecurity

IV. METHODOLOGY

A. *Hardware Specification*

Processor	Intel Core i3
RAM	8 GB
Hard Disk	500 GB

B. *Software Specification*

Front End	PYTHON, OpenCV
Back End	TensorFlow
Algorithm	CNN Algorithm

C. *About Algorithm*

CNN are made up of a large number of interconnected neurons that have learnable weights and biases. In the architecture of CNN the neurons are organized as layers. It consist of an input layer, many hidden layers and an output layer. If the network has a large number of hidden layers the same are generally referred as deep neural networks. The neurons in the hidden layers of CNN are connected to a small region (receptive field) of the input space generated from the previous layer instead of connecting to all, as in the fully connected network like Multi Layered Perceptron (MLP) networks. This approach reduces the number of connection weights (parameters) in CNN compared to MLP. As a consequence, CNN takes less time to train for the networks of similar size. The input to the typical CNN are two dimensions (2D) array of data such as images. Unlike the regular neural network the layers of a CNN are arranged in three dimensions (width, height and depth).

The followings are the type of layers which as commonly found in the CNNs.

1) Input Layer is basically a buffer to hold the input and pass it on to the next layer.

2) Convolution Layer performs the core operation of feature extraction. It performs Convolution operation of the input data. The convolution operation is executed by sliding the kernel over the input, and performs sum of the product at every location. The step size with which the kernel slides are known as stride. Numerous convolution operations are performed on the input by using different kernel, which results in different feature maps, the number of feature map produced in a convolutional layer is also known as the depth of the layer.

3) Rectified Linear Unit (ReLU) is an activation function used to introduce non linearity. It replaces negative value with zero, which can speed up the learning process. The output of every convolution layer is passed through the activation function.

4) Pooling layer reduces the spatial size of each feature map, which in turn reduces the computation in the network. Pooling also uses a sliding window that moves in stride across the feature map and transform it into representative values. Min pooling, average pooling and max pooling are commonly used.

5) Fully connected layer connects every neuron in the layer to all the neurons in the previous layer. It learns the non-linear combination of the features and used for classifying or predicting the output. For classification problems, the fully connected layer is generally followed by a soft-max layer, it produces the probability of each class for the given input. And for regression problems, it is followed by a regression layer to predict the output.

6) The architecture of the CNN is organized according to the problem to be addressed. Like any other neural networks the CNN is intelligent and they can learn. Learning is achieved through training (supervised). The CNN are feedforward networks and uses back-propagation training algorithm. The training is performed in two passes; forward and backward pass. In the forward pass the network weight and bias are initialized with small random numbers and compute the network output by using training input. The error is computed by comparing the network output with the desired training output. In the backward pass the error propagates backward and all the weights and bias are adjusted to minimize the error. The process is repeated until the desired result is obtained. Once the network is trained with a suitable dataset.

Steps of Training Process

a) Train a person face and capture image.
b) Collect dataset
c) Compare the image with collected dataset and the lively face
d) Find solution if a face matches with collected dataset it gives access
e) If its unmatched it will alert the admin by sending the mail
f) Authorized person can only access the email

V. RESULT AND DISCUSSION

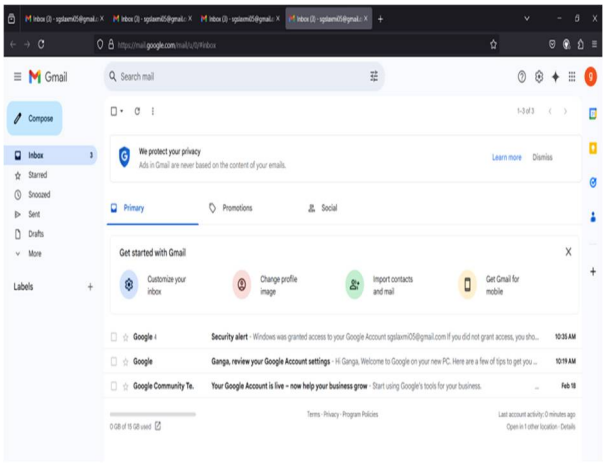


Figure 9-Access Email for Authorized Person

The project is to develop a product for locker security using face recognition. The implementation helps to restrict the unauthorized person to enter to open the locker. The captured face images can be trained and built as a model. The OpenCV computer vision library helps to perform face detection and recognition. The captured image will compare with the images in the database. If the comparison said to be true, the system predicts the authorized person name. If it fails to recognize, it will intimate with unknown label. The comparison works are handled by using CNN algorithm. For unknown person, the alert intimation will be sent to the user mail by using SMTP mail protocol. In proposed system for security purposes, image processing was implemented which helps to find the authorized person by face recognition concept. The camera placed in the door helps to capture the face and as per the matches, the locker door opens for authorized person. Here instead of storing one image, a group of images of various angle of users are stored and trained. Therefore the user can stand at any angle which get captured and compared in accurate manner. Any number of users image can be stored and trained which can be predicted in accurate manner. Only for those trained users, the locker triggers to open. The SMTP protocol is used to send mail to the authorized user during the time of unknown person prediction

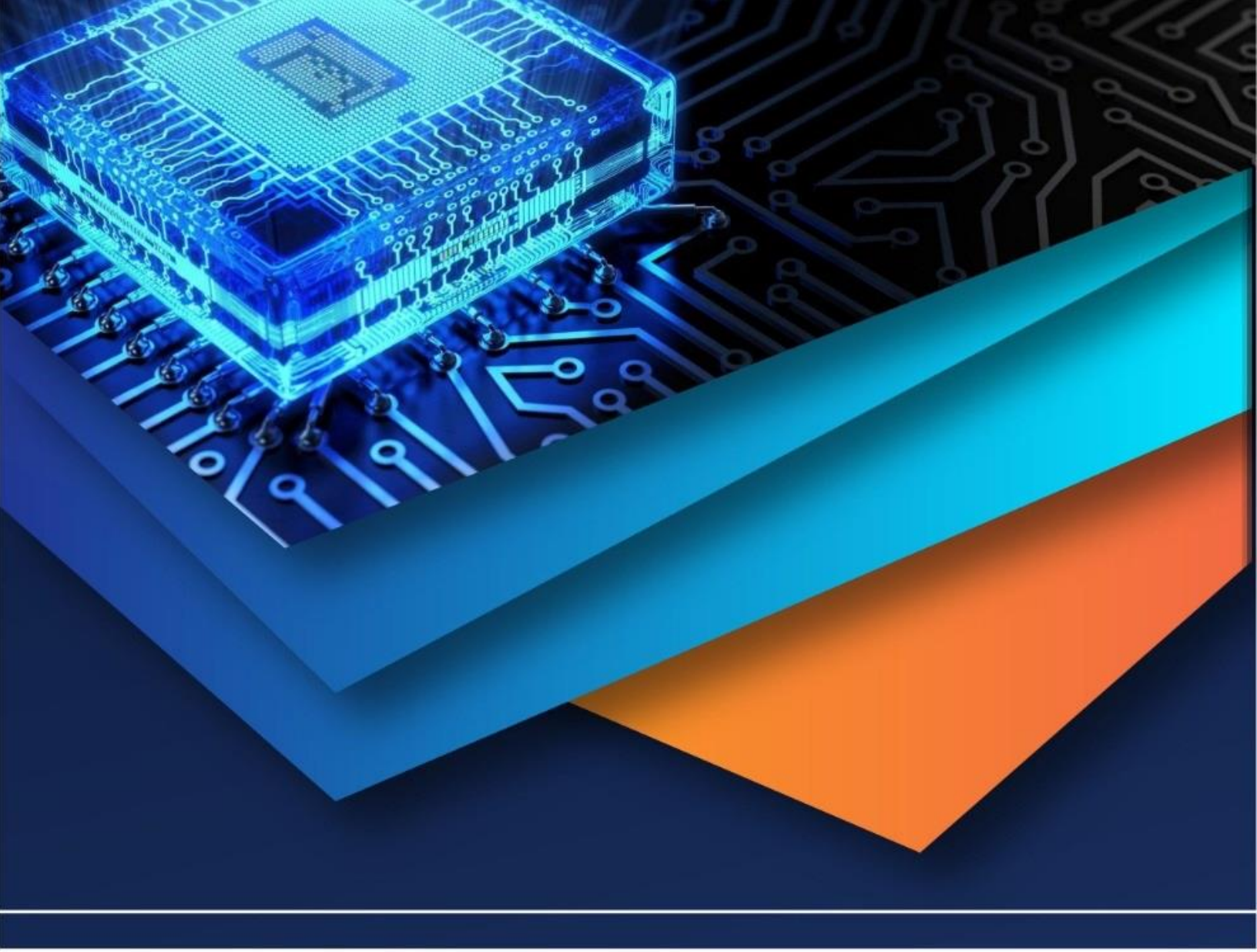
VI. CONCLUSION

In this project, I extend the concept of Multiview Face Detection using Convolution Neural Networks to provide an automatic tagging system. Once all the faces are detected, they are tagged using Local Binary Patterns Histograms (LBPH) method. Furthermore, Precision, Recall and F-measure calculations show an accuracy of 85% for tagging the faces which are successfully detected.

The live face detection and prediction can be implemented in a successive manner by using OpenCV and Numpy Library. This implementation helps the users to maintain the locker in safe and secure manner. Any unauthorized person access will be intimated to the account user by sending a mail. The proposed work and model creation also be extended to use it with any CCTV footage system for identifying people in case of theft, murder, etc. More specifically, it could even be used by the law enforcement agencies to identify absconding criminals in case they are spotted at the airport, railway station or bus-station by these surveillance cameras.

REFERENCES

[1] ”Automate The Boring Stuff With Python, 2Nd Edition: Practical Programming For Total Beginner”, SL Swegart, 2020.
[2] “Learning Python 5ed: Powerful Object-Oriented Programming” O’Relly – 2014.
[3] “Elements of Programming Interviews in Python: The Insiders' Guide”, Adan Aziz, Tsung-Hsen Lee – 2021.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY