MAULANA AZAD
NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL
INDIA, 462003

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

Multi-level Identity Authentication Framework

Major Project Report

Semester VIII

Submitted by:

Ragini Kalvade	181112256
Raj Mehroliya	181112244
Vanshika Agrawal	181112246

Under the Guidance of

Dr. Saritha S. K.

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

Session: 2021-22

MAULANA AZAD
NATIONAL INSTITUTE OF TECHNOLOGY BHOPAL
INDIA, 462003

DEPARTMENT OF COMPUTER SCIENCE & ENGINEERING

CERTIFICATE

This is to certify that the project report carried out on **“Multi-level Identity Authentication Framework”** by the 4th year students:

Ragini Kalvade	181112256
Raj Mehroliya	181112244
Vanshika Agrawal	181112246

Have successfully completed their project in partial fulfilment of their Degree in Bachelor of Technology in Computer Science and Engineering.

Dr. Saritha S. K.
(Major Project Mentor)

DECLARATION

We, hereby declare that the following report which is being presented in the Major Project Documentation Entitled as "**Multi-level Identity Authentication Framework**" is an authentic documentation of our own original work and to best of our knowledge. The following project and its report, in part or whole, has not been presented or submitted by us for any purpose in any other institute or organization. Any contribution made to the research by others, with whom we have worked at Maulana Azad National Institute of Technology, Bhopal or elsewhere, is explicitly acknowledged in the report.

Ragini Kalvade	181112256
Raj Mehroliya	181112244
Vanshika Agrawal	181112246

ACKNOWLEDGEMENT

With due respect, we express our deep sense of gratitude to our respected guide and coordinator Dr. Saritha S. K., for her valuable help and guidance. We are thankful for the encouragement that she has given us in completing this project successfully.

It is imperative for us to mention the fact that the report of major project could not have been accomplished without the periodic suggestions and advice of our project guide Dr. Saritha S. K. and project coordinators Dr. Manasi Gyanchandani and Dr. Bholanath Roy.

We are also grateful to our respected director Dr. N. S. Raghuwanshi for permitting us to utilize all the necessary facilities of the college.

We are also thankful to all the other faculty, staff members and laboratory attendants of our department for their kind cooperation and help. Last but certainly not the least; we would like to express our deep appreciation towards our family members and batch mates for providing the much-needed support and encouragement.

ABSTRACT

Hacking attacks affect approximately one in three Americans every year, and cyber-security has become a trillion-dollar industry. Criminals often target small and mid-size companies or individuals who do not have the latest security systems.

New trends in user authentication seek to balance better cybersecurity with increased convenience. Online security evolves rapidly, along with the evolution of the internet itself and the methods for carrying out malicious attacks. In order to protect your business and offer your users a trustful user journey, you need to implement security procedures that protect your users identity, data and devices.

Biometric authentication uses the unique biological characteristics of an individual to verify their identity. Biometric identification systems can look at someone's fingerprints, voice, retina, or facial features. This is one of the most trusted options for verifying someone's identity because it is almost impossible to fake a fingerprint or fool a retina scanner unless you have specialized knowledge or training.

According to research carried out by Visa, customers are willing to use biometric rather than passwords or PIN: 70 % of users believe it is an easier system, 46 % think it is more secure, and 86 % are interested in using it as a verification system.

Facial verification or fingerprint scans provide a convenient authentication option, but they are best when combined, in view of the fact that after an initial breach, a hacker could gain access to the entire system unless there are multiple layers of security (which would hamper convenience for users).

In this project, a multilevel authentication system is proposed wherein user convenience is factored in while ensuring a secure authentication of valid users. Three levels of authentication - Password protection, Facial Verification, Signature verification have been constructed for this system. By providing multiple highly secure levels of authentication, only validated users can access data and user identity is protected.

TABLE OF CONTENTS

LIST OF FIGURES

INTRODUCTION

1.1 Background

Authentication systems are security measures put in place to secure data and systems by requiring additional input beyond username and password for users to access a system. By providing this additional input, authentication systems help ensure the veracity of the users. Authentication systems can require one other form of user input or more. These systems are sometimes called multiple-factor authentication, or MFA.

Using authentication improves data security and prevents potential breaches. When multi-factor authentication is required to access a system, the system is less vulnerable to security issues like weak passwords or attacks like phishing. Authentication systems are ideal for businesses with sensitive data or systems that require secure user accounts.

Biometric authentication involves using some part of the physical makeup to authenticate. This could be a fingerprint, an iris scan, a retina scan, or some other physical characteristic. A single characteristic or multiple characteristics could be used depending on the infrastructure and the level of security desired.

Convenient biometric authentication options such as facial verification or fingerprint scans are best when combined. Multiple levels of security would hamper hackers after an initial breach of the system and not allow complete access to sensitive files instantly.

A signature is a handwritten representation of the name of a person. Writing a signature is the established method for authentication. As compared to textindependent writer recognition methods, signature/sign recognition goes with shorter handwriting probes, but requires writing the same name or personal sign every time. Handwritten signatures and personal signs belong to the behavioral biometric characteristics as the person must become active for signing.

Online signature recognition systems can be a good component for a multimodal biometric facility. They are comparatively more cost efficient than other biometric systems and ease of use. Also, forgery is comparatively critical as several parameters like x, y coordinates, pen pressure, pen inclination angle, acceleration and velocity, signing duration etc.) can be extracted from an online signature curve. Implementing an online signature system to multi-modal biometric system is easier to implement (as spectacles, beard, nose ring, changed hairstyle or mustache are troublesome in Face Recognition or cold and cough may change voice input for voice biometric.). So, in any multi-modal biometric facility, an online signature recognition authorization can be of utter importance.

Facial authentication is a more secure component of biometrics. The user authenticates with their face as their credential to securely access their online account. To authenticate, the user's image is converted to a 3D map which is compared, one-to-one, with a stored biometric template. The process is a success when a proper match, based on an accuracy score, is achieved.

As a standalone method, it has its own shortcomings and can be the single point of failure in the verification or authentication process. Distinguishing the differences between verifying and authenticating a user - When facial recognition technology verifies a user, it detects an individual's face, analyses it and then compares it against information provided such as an identity card.

After identity has been verified, authentication is when a customer’s identity is confirmed by requesting further credentials to allow access to services. This allows user convenience while providing a secure protected platform.

This multi-factor authentication is designed to decrease uncertainty by increasing the resistance to compromise. It decreases uncertainty by combining “something you know” with “something you are or possess”. The “something you know” is the password and the something you possess is the signature and facial authentication level of the system.

1.2 Problem Definition

Identity theft is real and is increasing at an alarming rate. Fraudsters have been advancing their techniques, relying on a variety of digital fraud strategies such as creating synthetic identities or fake identities that use real information about victims. An increase in the creation and use of ‘deep fakes’ or ‘synthetic media’ is also seen.

It is a growing problem which increases the need for secure, reliable, and trusted digital identity verification systems. Artificial Intelligence (AI) is enabling companies to fight this by analyzing thousands of technological and behavioral data points in a matter of seconds to identify fraudulent and strange behavior.

A Multi-Level Identity authentication system is proposed which determines if the person is who they say they are. Authentication relies on additional data that is difficult to produce, except by that specific person. The problem is solved using a three-level framework, which performs password authentication, signature verification and then cross comparing the real time face to ID photo. In order to achieve this, we employ image analysis and computer vision techniques.

Signature Recognition and Validation is the process of authenticating the person’s identity by checking his signature against samples which are previously collected. Facial comparison is typically used where a trusted source image of the user is available to compare a real-time image against. The trusted source picture is usually extracted from an authenticated identity document such as a passport.

LITERATURE REVIEW AND SURVEY

In offline signature verification, template matching and Hidden Markov model techniques are generally employed. These techniques are based on the structure of the signature. Template matching is a technique in digital image processing for fine small parts of an image which match a template image. When it comes to template matching, metrics like n square error or structural similarity index, or a warping method can be used which warps one curve onto another so that the original shape is maintained.

The use of Deep CNNs to identify who the signature belongs to and whether it is a forgery is also an option. This is done in a two-phase approach - writer-independent feature learning, and writer dependent classification. This approach simplifies it by treating the signatures and their forgeries as separate classes. It is a completely writer independent approach.

Alan McCabe et al. proposed a method for verifying handwritten signatures by using NN architecture. Various static (e.g., height, slant, etc.) and dynamic (e.g., velocity, pen tip pressure, etc.) signature features are extracted and used to train the NN Several Network topologies are tested and their accuracy is compared [3].

Ferrer et al. calculates geometric features of a signature in fixed-point arithmetic for offline verification. The proposed features are then checked with different classifiers, such as Hidden Markov Models, Support Vector Machines etc. [2].

A novel approach to off-line signature verification is proposed by Wei Tian et al. Both static and pseudo dynamic features are extracted as original signals, which can enhance the difference between a genuine signature and its forgery [1].

Eman Alajrami [4] put forth a method that uses CNNs to learn the signatures, though the structure of the fully connected layer is not optimal and may be considered extreme. In the model created in this work, two classes are created for each user (Real and forgery) and best accuracy attained was 99.7%.

A multitude of techniques have evolved for face recognition. The different techniques have been presented below as they were proposed initially and how they have evolved -

PCA (Principal Component Analysis)

This is a standard method for the statistical design in order to reduce dimensionality and used for the feature extraction. This method is used to preserve the important features and neglect or remove the redundant information related to features. A face contains certain features and these features are said to be the principal components or Eigen faces. These features are said to be extracted by the help of principal analysis [5].

LDA (Linear Discriminant Analysis)

This method is also called fisher face, which is an appearance based technique used for dimensional reduction. This method is similar to the PCA for the feature extraction. LDA makes use of projection of the training images into sub space (face space) which is elaborated by the fisher face known as fisher space. The conversion of images goes from N-2 dimension to C-dimension space, C, denotes the number of classes of images said to be projected in the fisher space [6].

ICA (Independent Component Analysis)

ICA is a generalized form of PCA. Higher order dependencies are not supported by PCA, because it is based on the second order statistics of the image set. In ICA, inclusion of second order also separates the higher order moments. ICA is said to be laid in two types of architectures, Architecture 1 takes the image of the random variables and provides the outcomes as pixels. Architecture 2 produces the independent coding variables. The factorial facial code is produced by the probability of combination of features, with the individual probabilities.

AAM (Active Appearance Model)

AAM is to be work based on the alignment features. In this model after collecting several images that have different alignment features. This model provides the accurate alignment with pose corrections also.

LBP (Local Binary Patterns)

The Local binary patterns are used to exhibit the texture as well as the shape of the digital image. This is done by dividing the digital image into several parts of small regions in order to extract the features. The region to be examined is called a window and a specific divided part is called a cell. The surrounding of the pixel in the region is provided as the binary patterns that contain these features [6].

The most popular methods are the Haar Classifier[8] and CNN [9] because the Haar Classifier is easy to implement and CNN is quite accurate. Several widely used libraries that employ a Haar classifier include OpenCV and Dlib. On the other hand, Facenet and ArcFace apply MTCNN [10] which is a type of CNN. After the face detection is completed, face comparison can proceed. The approaches which include Dlib, Facenet, and ArcFace are widely accepted

GAPS IDENTIFIED

A sign-on system has several vulnerabilities which can be overcome by using multi-factor authentication systems. A combination of established authentication methods increases security and allows only validated users access. Multi-factor authentication (MFA) reduces the risk of security breaches from occurring and keeps data safe. In the past, requiring a static username and password to access an account seemed sufficient for security. However, weak or stolen passwords can be used to execute fraud attacks and data breaches when they are the only form of authentication required. Using MFA to bolster password security with another form of authentication is proven to keep hackers out of your systems. According to Microsoft, MFA can "prevent 99.9 percent of attacks on your accounts."

In general, single authentication is more concerned with providing access than with restricting it. At a time when malware-based attacks are rampant, more access is not always a good thing. There are quite a few risks that come along with utilizing a single factor access - Instant access to more than just the endpoint, less control once access is granted and little-to-no adherence to the principle of least privilege. In consideration of these shortcomings the implementation of multi-factor authentication systems comes across as a feasible option.

PROPOSED WORK AND METHODOLOGY

4.1 PROPOSED WORK

To overcome the drawbacks of the methods that are reviewed above, a new multilevel framework for identity authentication is proposed. In this system, different techniques are combined to achieve the final goal of identity verification and authentication.

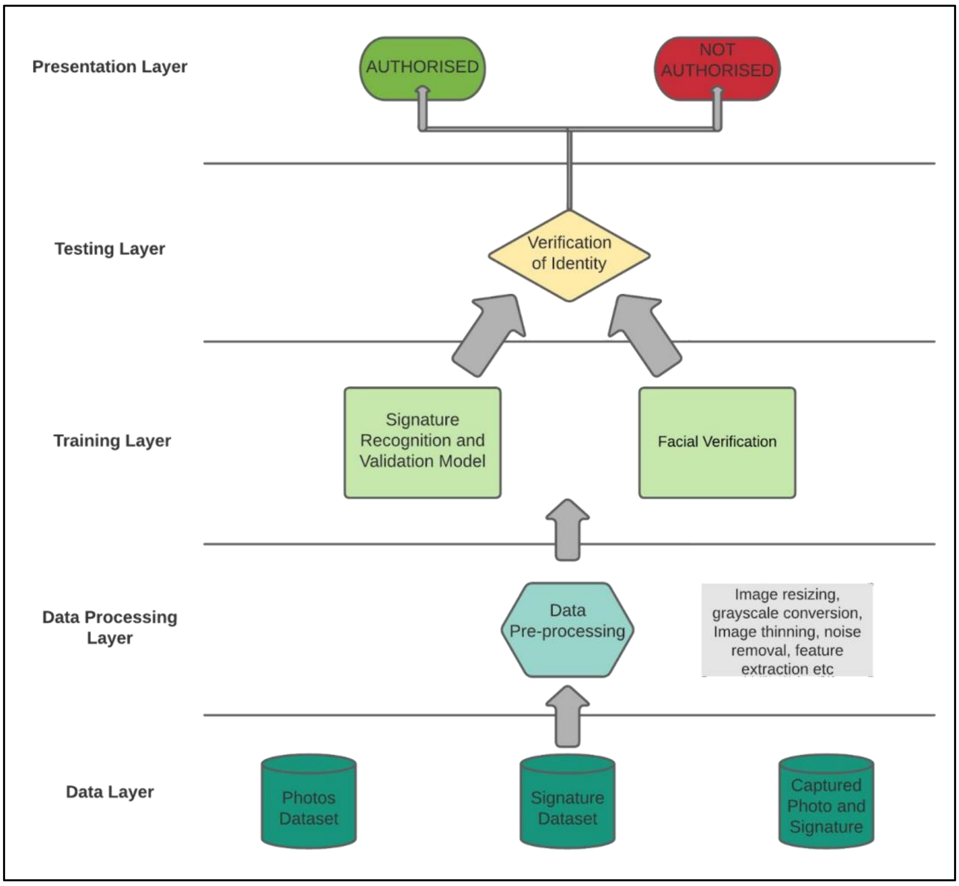


Fig1. Architecture of the project

The steps for the process are documented below -

1. Signature Recognition and Verification - A model based on offline base signature verification is created. Signature verification is the process of authenticating the person’s identity by checking his signature against samples which are previously collected.

2. Facial Verification - A model based on Haar Classifier using OpenCV is proposed for facial detection and verification.

3. Combination of the two to authenticate user - The results yielded by the above two models are combined and, on the basis, result is calculated.

4.2 METHODOLOGY

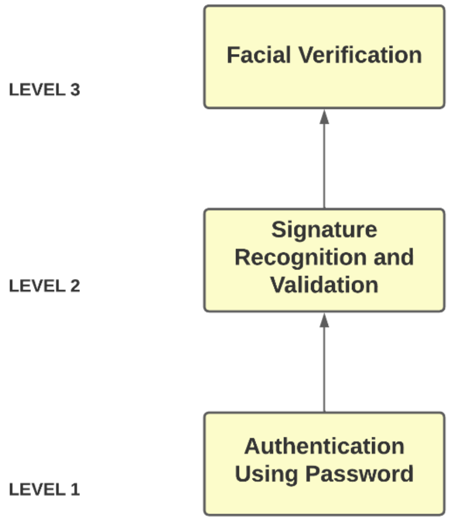

Facial Verification
LEVEL 3
Signature Recognition and Validation
LEVEL 2
Authentication Using Password
LEVEL 1

Fig2. Methodology

4.2.1 Authentication using password

Authentication is the process of verifying who a user claims to be. There are 3 factors of verification -

- What you know — Something you know, such as a password, PIN, personal information like mother's maiden name, etc.
- What you have — A physical item you have, such as a cell phone or a card.
- What you are — Biometric data, such as fingerprint, retina scan, etc.

Password authentication falls into the "what you know" category and is the most common form of authentication. Django comes with a user authentication system. It handles user accounts, groups, permissions and cookie-based user sessions. Django authentication provides both authentication and authorization together and is generally referred to as the authentication system, as these features are somewhat coupled. Authentication verifies a user is who they claim to be, and authorization determines what an authenticated user is allowed to do.

4.2.2 Signature Recognition and Validation

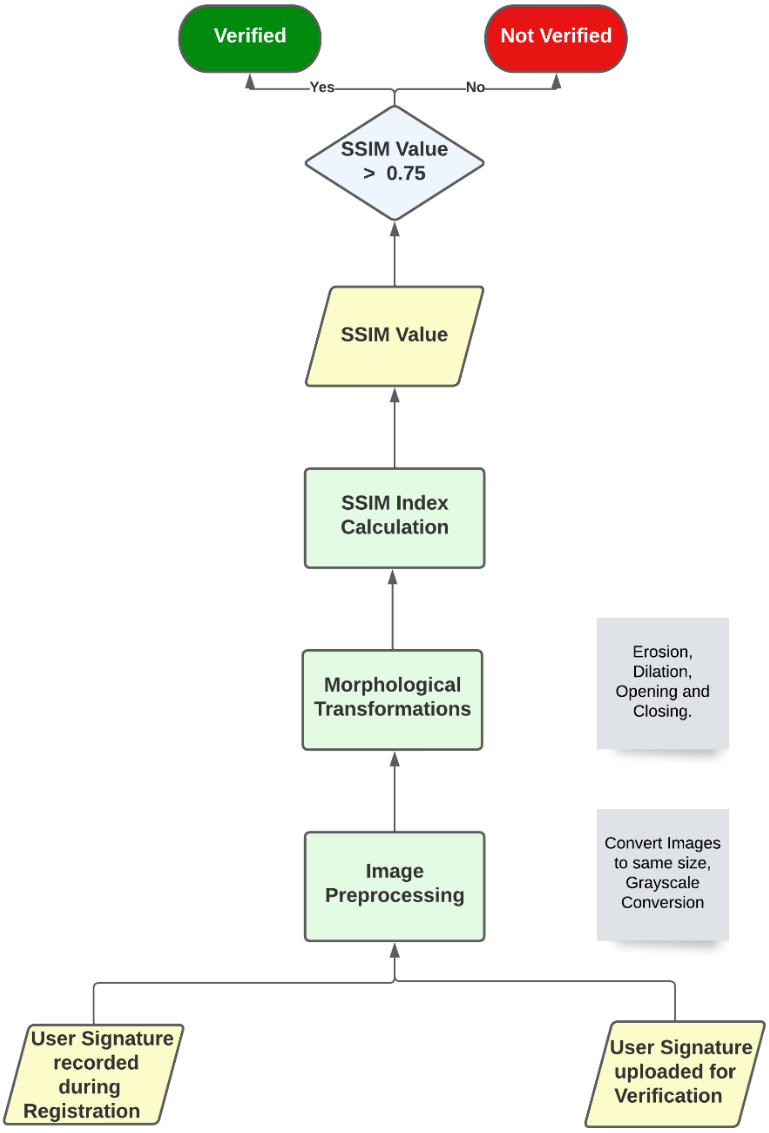

Verified
Not Verified
Yes
No
SSIM Value > 0.75
SSIM Value
SSIM Index Calculation
Morphological Transformations
Erosion, Dilation, Opening and Closing.
Image Preprocessing
Convert Images to same size, Grayscale Conversion
User Signature recorded during Registration
User Signature uploaded for Verification

Fig3. Signature Verification Algorithm

The Structural Similarity Index Measure (SSIM) and detection of special points are based on a skeleton signature representation. It is a perception-based model that considers image degradation as perceived change in structural information. Structural information is a feature that considers pixels which are correlated and spatial near-by in the image, such as directional pixel intensity.

The SSIM values vary between -1 and 1, where the 1 indicates perfect similarity, while 0 shows the opposite, finally -1 is only achieved theoretically. SSIM shows the results of verification and demonstrates low values of False Rejection Rate (FRR) and False Acceptance Rate (FAR).

The Mathematical Formula is –

$$\mathrm{SSIM}(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

Fig4. SSIM Index Formula

The parameters to Equation include the (x, y) location of the N x N window in each image, the mean of the pixel intensities in the x and y direction, the variance of intensities in the x and y direction, along with the covariance.

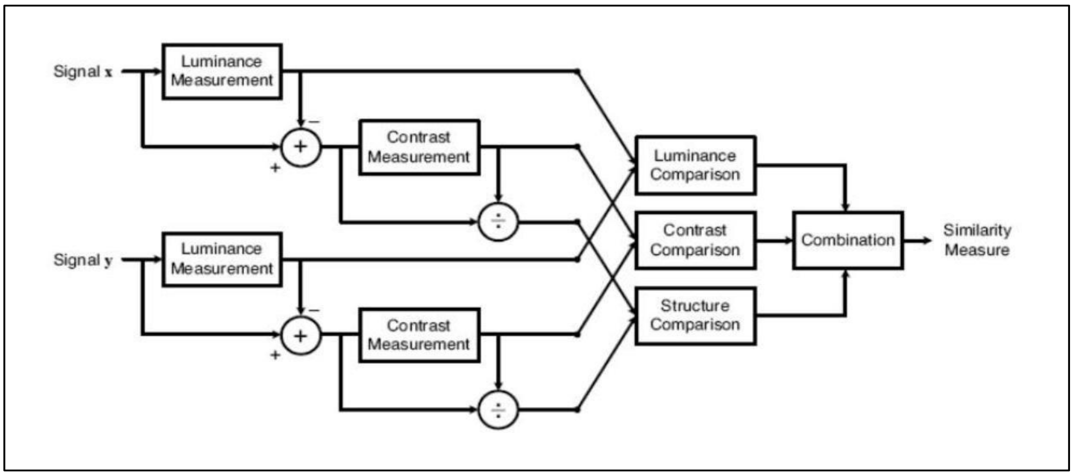
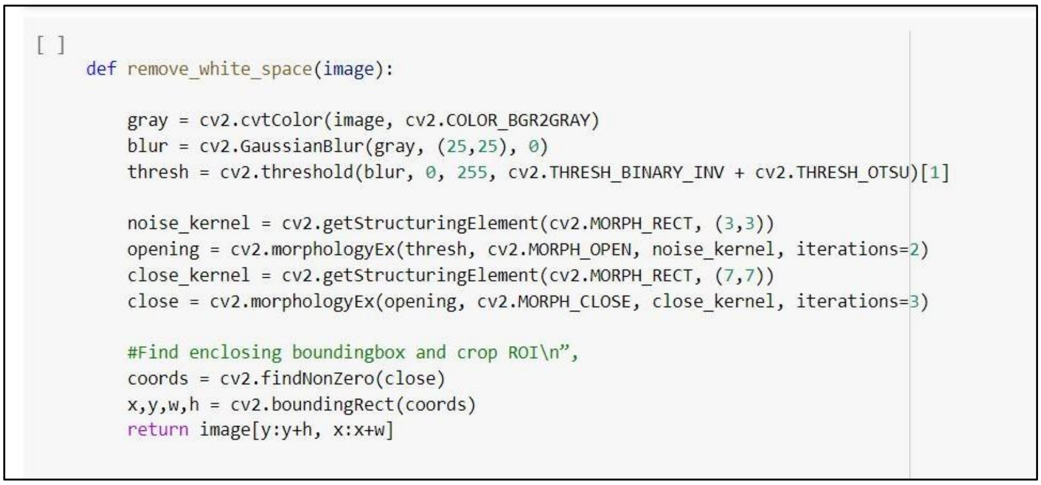


Fig5. SSIM Algorithm

To compare the images using SSIM, we need to pass two gray scale images of same size. Applying Morphological operations to extract the closed region of signature and validate the signature using Structural Similarity Index (SSIM)

- Morphological transformations are image-based operations performed to remove the extra white spaces from an image, or joining the broken images or thickening the characters in an image, etc., In other words we can say morphological operations are performed to remove noise from an image. Basically, it includes Erosion, Dilation, Opening and Closing.



```
[ ]
    def remove_white_space(image):

        gray = cv2.cvtColor(image, cv2.COLOR_BGR2GRAY)
        blur = cv2.GaussianBlur(gray, (25,25), 0)
        thresh = cv2.threshold(blur, 0, 255, cv2.THRESH_BINARY_INV + cv2.THRESH_OTSU)[1]

        noise_kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (3,3))
        opening = cv2.morphologyEx(thresh, cv2.MORPH_OPEN, noise_kernel, iterations=2)
        close_kernel = cv2.getStructuringElement(cv2.MORPH_RECT, (7,7))
        close = cv2.morphologyEx(opening, cv2.MORPH_CLOSE, close_kernel, iterations=3)

        #Find enclosing boundingbox and crop ROI\n”,
        coords = cv2.findNonZero(close)
        x,y,w,h = cv2.boundingRect(coords)
        return image[y:y+h, x:x+w]
```

Fig6. Morphological Transformation Code

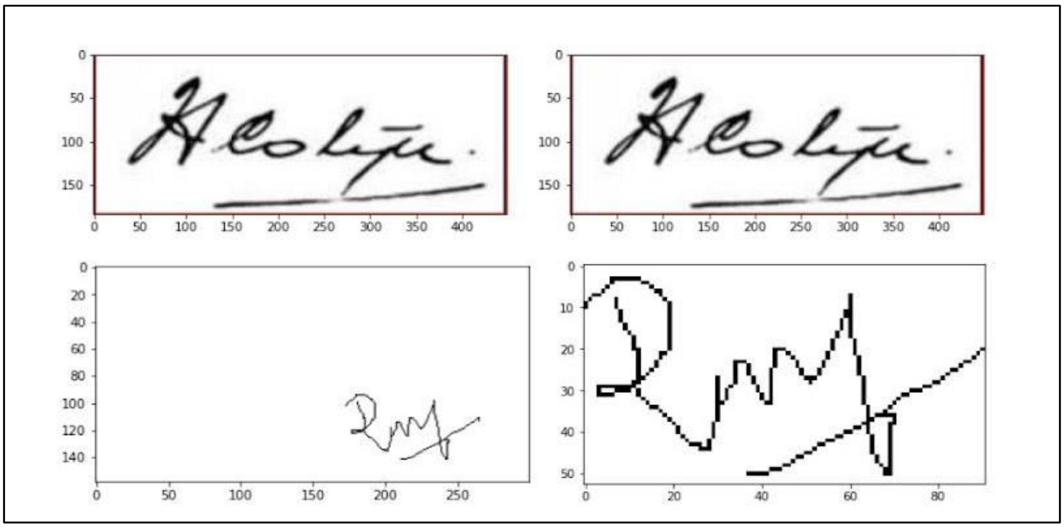


Fig7. Signatures after Transformation

- The correlation coefficient between the resultant image and the images in the database which were also undergone the same process as the input image is calculated. A threshold value is set up to which the correlation coefficient should exist. The image with the value greater than the threshold is displayed with the name of the person. The above whole process is done with the help of a graphical user interface which is user-friendly. Then, compare the signature with database signature. To compare the signature with the sample database signature a simple algorithm has been introduced based on pixel matching concept which is easy to implement and computationally less complex.

4.2.2 Facial Verification

Facial Recognition is the biometric technique used in face detection. The task for validating or recognizing a face from the multi-media photographs is done using facial recognition technique.

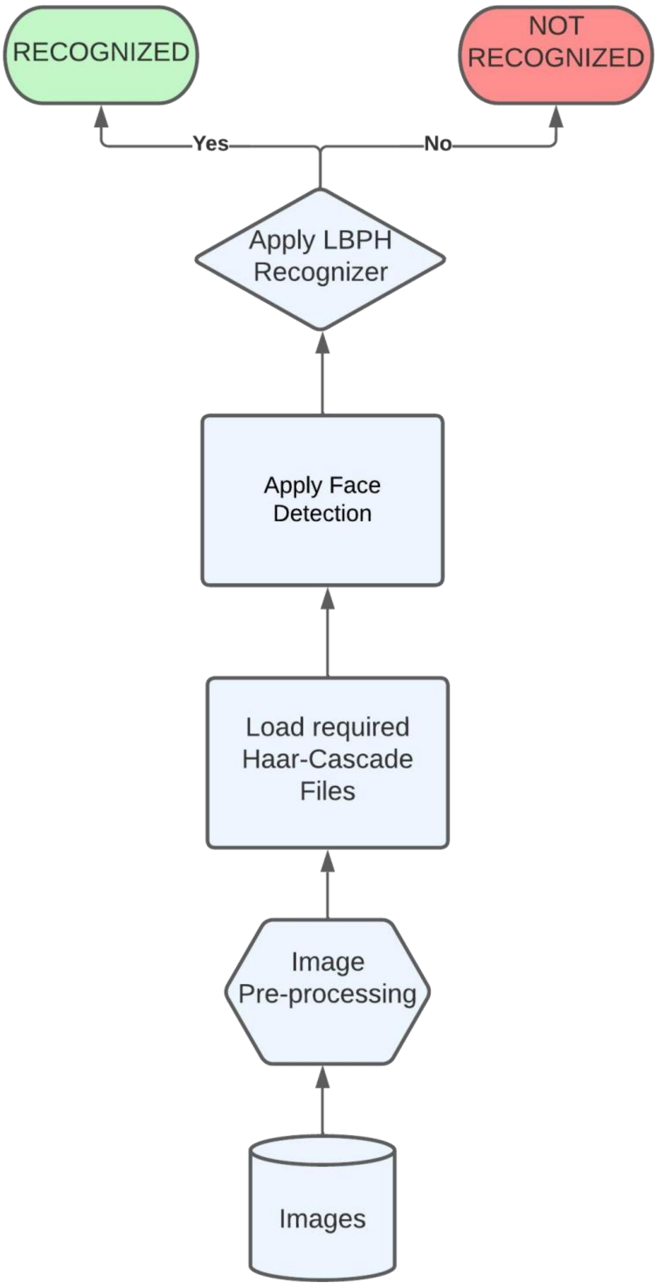


Fig8. Facial Verification Algorithm

The 2 phases of facial verification are -

Face detection and gathering

“Haarcascade” is a popular algorithm for facial detection. It is computationally less expensive, a fast algorithm, and gives high accuracy. It works in four stages -

- Calculating Haar features - A Haar feature is essentially calculations that are performed on adjacent rectangular regions at a specific location in a detection window. The calculation involves summing the pixel intensities in each region and calculating the differences between the sums.

- Creation of Integral Images - Integral images essentially speed up the calculation of these Haar features. Instead of computing at every pixel, it instead creates sub-rectangles and creates array references for each of those sub-rectangles. These are then used to compute the Haar features.

- Adaboost Training - Adaboost essentially chooses the best features and trains the classifiers to use them. It uses a combination of “weak classifiers” to create a “strong classifier” that the algorithm can use to detect objects.

- Cascade Classifier - The cascade classifier is made up of a series of stages, where each stage is a collection of weak learners. Weak learners are trained using boosting, which allows for a highly accurate classifier from the mean prediction of all weak learners. Based on this prediction, the classifier either decides to indicate an object was found (positive) or move on to the next region (negative). Stages are designed to reject negative samples as fast as possible, because a majority of the windows do not contain anything of interest.

OpenCV comes with lots of pre-trained classifiers. Those XML files can be loaded by the cascadeClassifier method of the cv2 module.

Training the recognizer and face recognition

The first computational step of the LBPH is to create an intermediate image that describes the original image in a better way, by highlighting the facial characteristics. To do so, the algorithm uses a concept of a sliding window, based on the parameters radius and neighbors.

The LBPH algorithm typically makes use of 4 parameters -

- Radius - The distance of the circular local binary pattern from the center pixel to its circumference and usually takes a value of 1.
- Neighbors - The number of data points within a circular local binary pattern. Usually, the value of 8.
- Grid X - The number of cells in the horizontal plane, is usually a value of 8.
- Grid Y - The number of cells in the vertical plane is usually a value of 8.

A central value of the matrix is created by the conversion of the binary number to a decimal value which corresponds to the pixels of the original image. For a better representation of the characteristics of the original image.

Each one made a histogram for an image in the training data set. Two histograms are compared to output the image with the closest histogram matches to an input image. This output is the ID or name of the image. This algorithm also returns a confidence' measurement which is the calculated distance. The correctness of the algorithm in recognizing the image is estimated automatically by the confidence and the threshold.

The stepwise implementation of the facial verification model is as below -

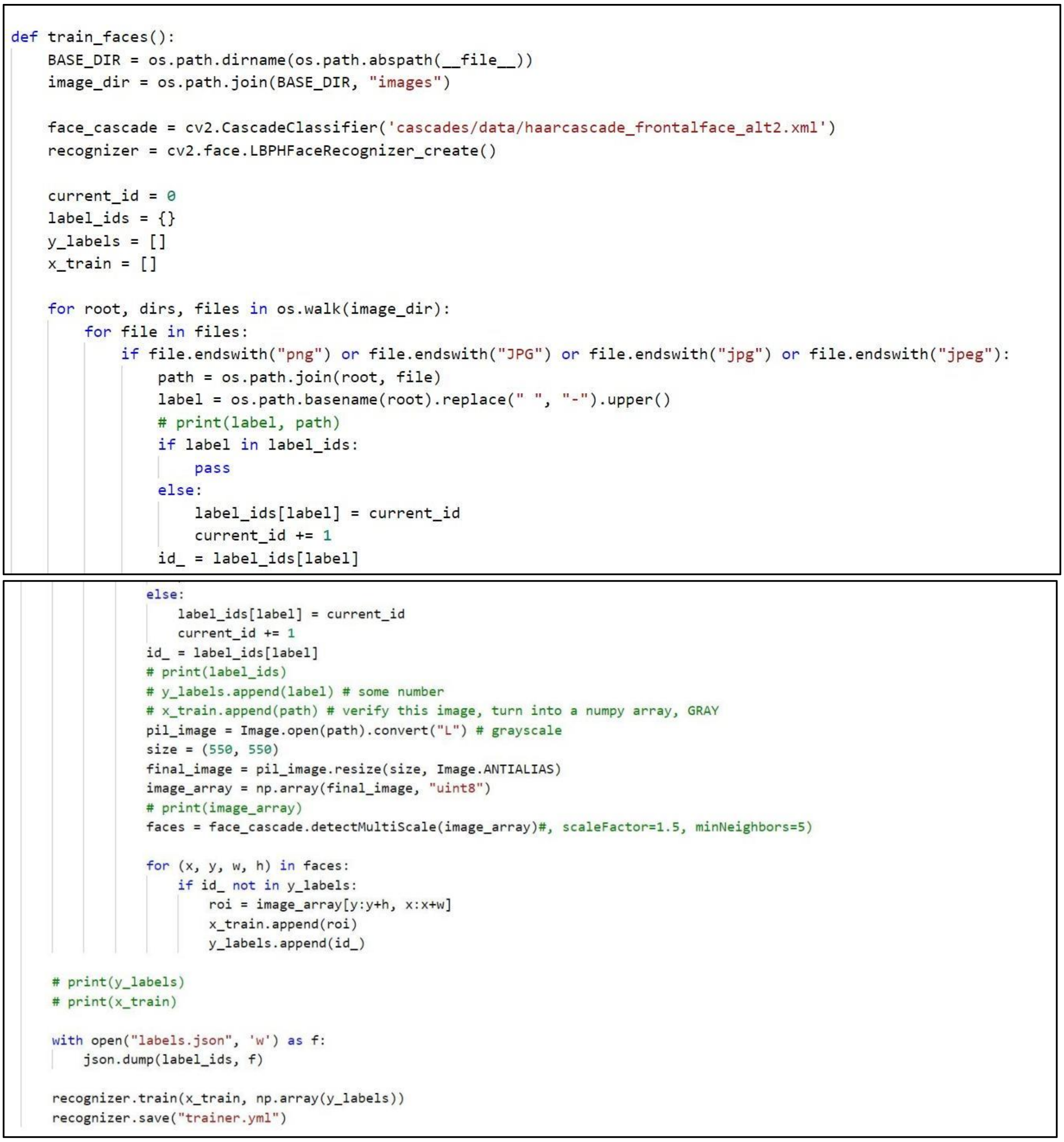
Step 1 - Load the image

Step 2 - Converting the image to grayscale

Step 3 - Loading the required haar-cascade XML classifier file

Step 4 - Applying the face detection method on the grayscale image

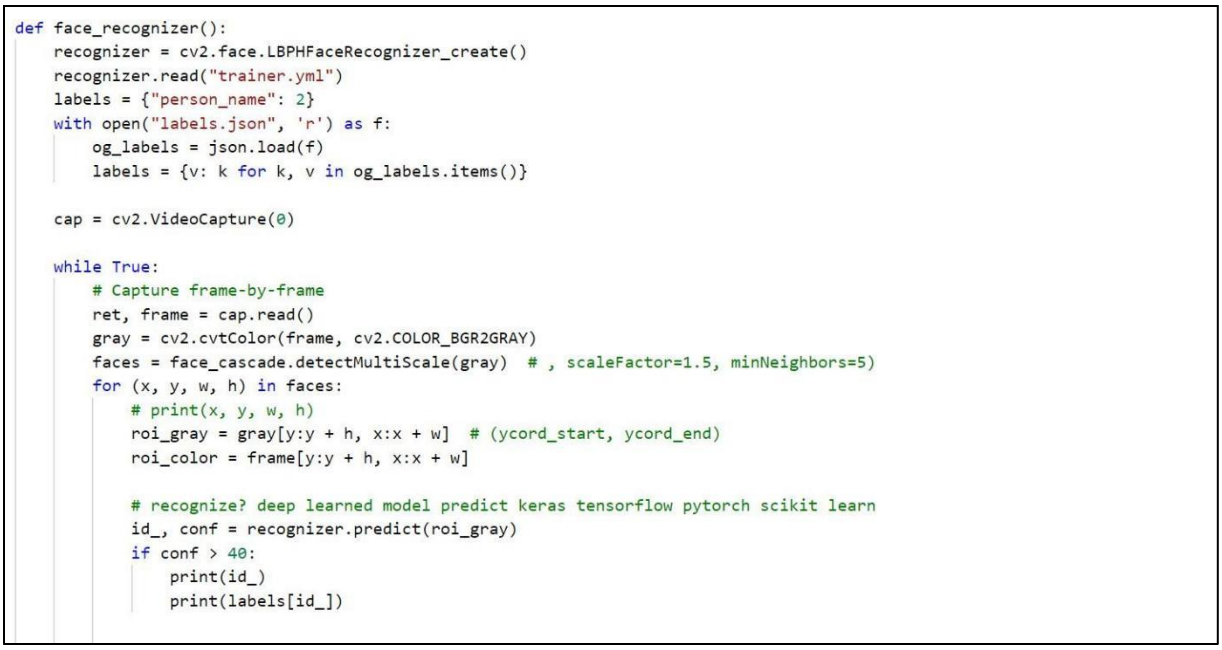
Step 5 - Iterating through rectangles of detected faces



```
def train_faces():
    BASE_DIR = os.path.dirname(os.path.abspath(__file__))
    image_dir = os.path.join(BASE_DIR, "images")

    face_cascade = cv2.CascadeClassifier('cascades/data/haarcascade_frontalface_alt2.xml')
    recognizer = cv2.face.LBPHFaceRecognizer_create()

    current_id = 0
    label_ids = {}
    y_labels = []
    x_train = []

    for root, dirs, files in os.walk(image_dir):
        for file in files:
            if file.endswith("png") or file.endswith("JPG") or file.endswith("jpg") or file.endswith("jpeg"):
                path = os.path.join(root, file)
                label = os.path.basename(root).replace(" ", "-").upper()
                # print(label, path)
                if label in label_ids:
                    pass
                else:
                    label_ids[label] = current_id
                    current_id += 1
                id_ = label_ids[label]
```

```
                else:
                    label_ids[label] = current_id
                    current_id += 1
                id_ = label_ids[label]
                # print(label_ids)
                # y_labels.append(label) # some number
                # x_train.append(path) # verify this image, turn into a numpy array, GRAY
                pil_image = Image.open(path).convert("L") # grayscale
                size = (550, 550)
                final_image = pil_image.resize(size, Image.ANTIALIAS)
                image_array = np.array(final_image, "uint8")
                # print(image_array)
                faces = face_cascade.detectMultiScale(image_array)#, scaleFactor=1.5, minNeighbors=5)

                for (x, y, w, h) in faces:
                    if id_ not in y_labels:
                        roi = image_array[y:y+h, x:x+w]
                        x_train.append(roi)
                        y_labels.append(id_)

    # print(y_labels)
    # print(x_train)

    with open("labels.json", 'w') as f:
        json.dump(label_ids, f)

    recognizer.train(x_train, np.array(y_labels))
    recognizer.save("trainer.yml")
```

Fig9. Face Detection Training Model Code

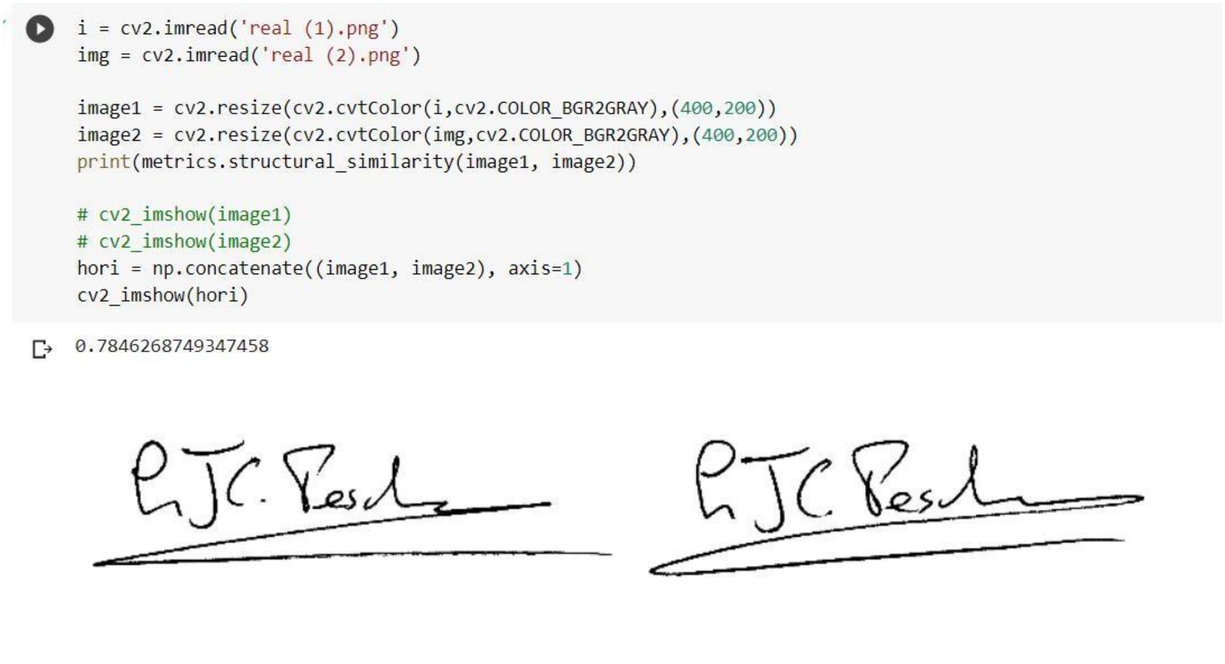


```
def face_recognizer():
    recognizer = cv2.face.LBPHFaceRecognizer_create()
    recognizer.read("trainer.yml")
    labels = {"person_name": 2}
    with open("labels.json", 'r') as f:
        og_labels = json.load(f)
        labels = {v: k for k, v in og_labels.items()}

    cap = cv2.VideoCapture(0)

    while True:
        # Capture frame-by-frame
        ret, frame = cap.read()
        gray = cv2.cvtColor(frame, cv2.COLOR_BGR2GRAY)
        faces = face_cascade.detectMultiScale(gray)  # , scaleFactor=1.5, minNeighbors=5)
        for (x, y, w, h) in faces:
            # print(x, y, w, h)
            roi_gray = gray[y:y + h, x:x + w]  # (ycord_start, ycord_end)
            roi_color = frame[y:y + h, x:x + w]

            # recognize? deep learned model predict keras tensorflow pytorch scikit learn
            id_, conf = recognizer.predict(roi_gray)
            if conf > 40:
                print(id_)
                print(labels[id_])
```

Fig10. Face Verification Model Code

RESULTS AND DISCUSSION

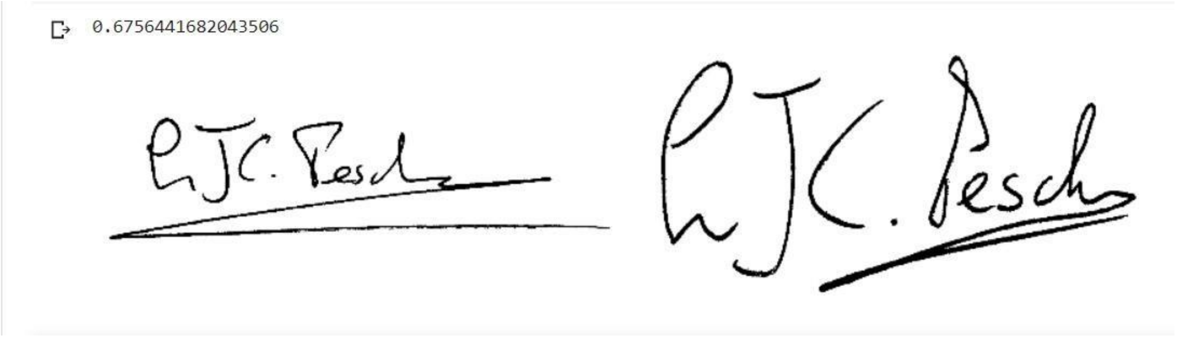
The employment of SSIM for signature verification produced the following results. A database of 10 users with real and forged signatures was tested against the model which led to the determination of the threshold value for SSIM index. According to the results obtained, a set of signatures with an SSIM value of **less than 7.5** will be forged, whereas anything with an SSIM index of **7.5 or above** will be characterized as matching and hence the user will be verified.



```
i = cv2.imread('real (1).png')
img = cv2.imread('real (2).png')

image1 = cv2.resize(cv2.cvtColor(i,cv2.COLOR_BGR2GRAY),(400,200))
image2 = cv2.resize(cv2.cvtColor(img,cv2.COLOR_BGR2GRAY),(400,200))
print(metrics.structural_similarity(image1, image2))

# cv2_imshow(image1)
# cv2_imshow(image2)
hori = np.concatenate((image1, image2), axis=1)
cv2_imshow(hori)
```

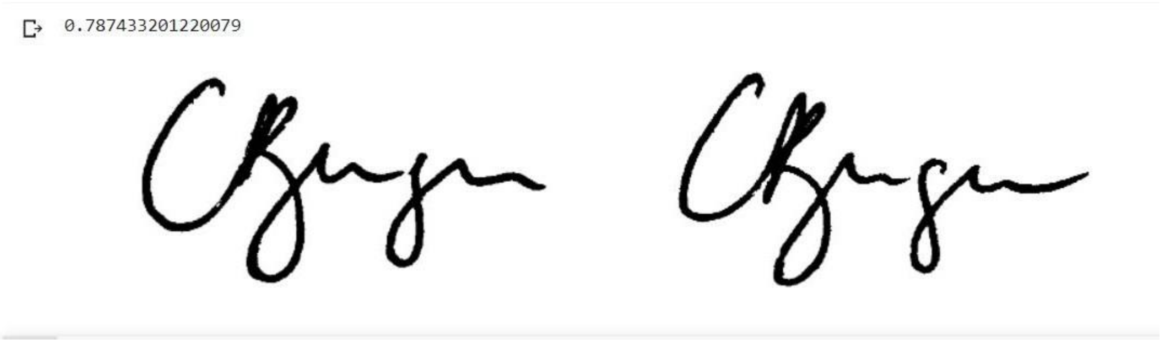
```
0.7846268749347458
```

Fig11. Comparison of Real Signatures of User 1



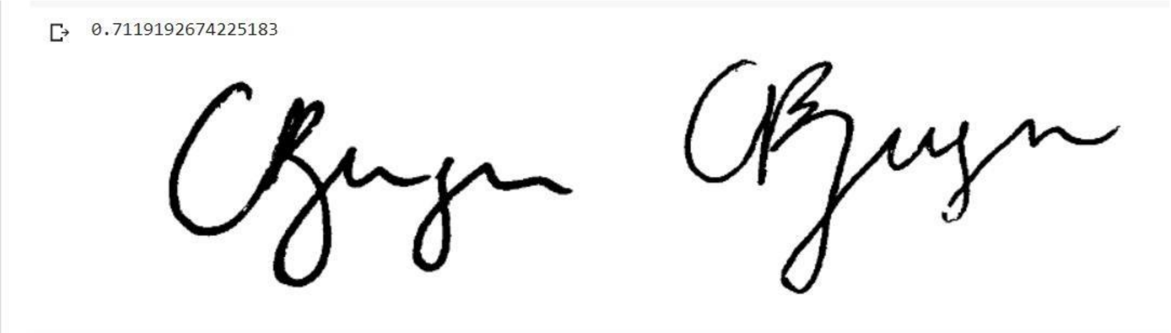
```
0.6756441682043506
```

Fig12. Comparison of Real & Forged Signatures of User 1



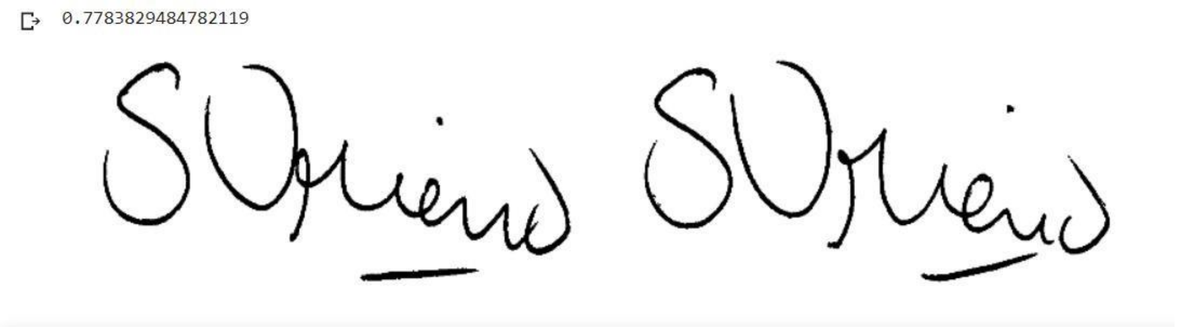
```
0.787433201220079
```

Fig13. Comparison of Real Signatures of User 2



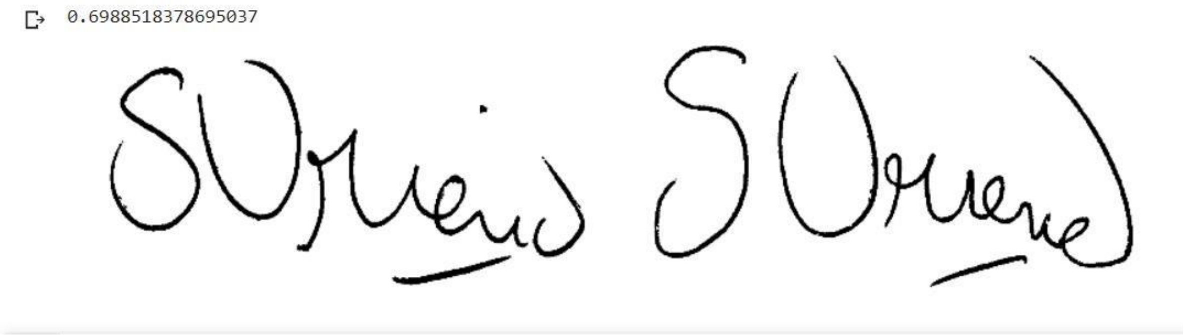
```
0.7119192674225183
```

Fig14. Comparison of Real & Forged Signatures of User 2



```
0.7783829484782119
```

Fig15. Comparison of Real Signatures of User 3



```
0.6988518378695037
```

Fig16. Comparison of Real & Forged Signatures of User 3

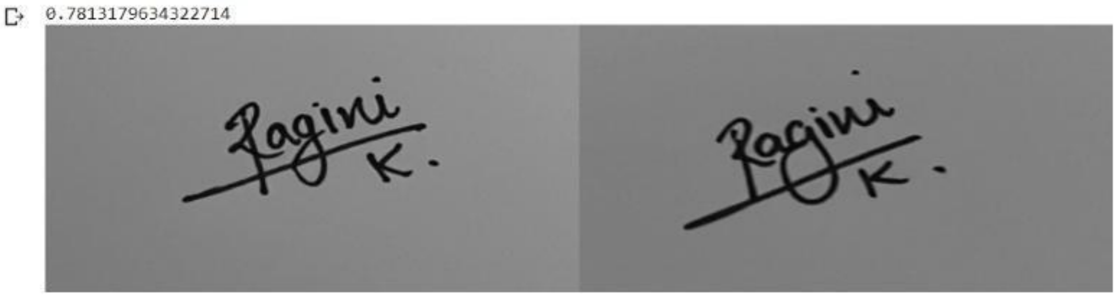

```
0.7813179634322714
```

Fig17. Comparison of Real Signatures of User 4

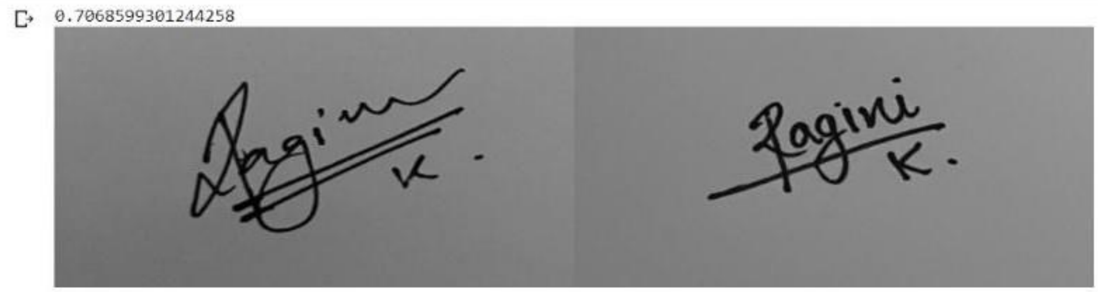

```
0.7068599301244258
```

Fig18. Comparison of Real & Forged Signatures of User 4

The Facial Recognition model upon execution opens up a frame which recognizes a user based on their username if they are present in the database otherwise gives no result.

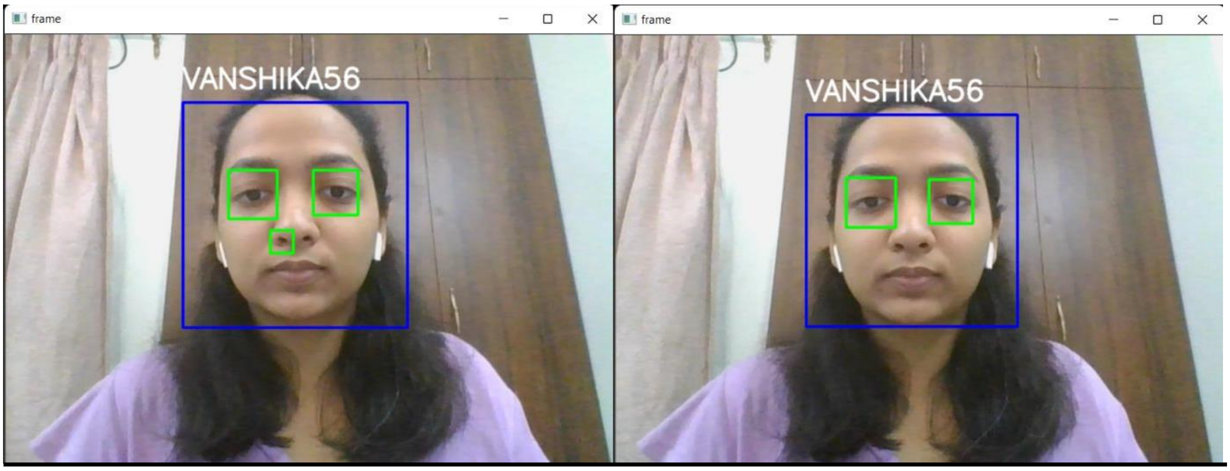
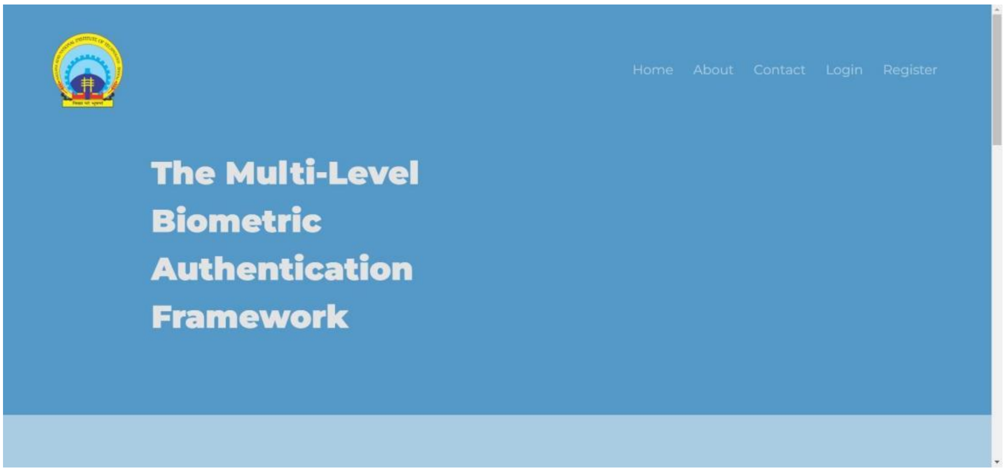


Fig19. Facial Recognition Result for a user in the database

WORKING MODEL

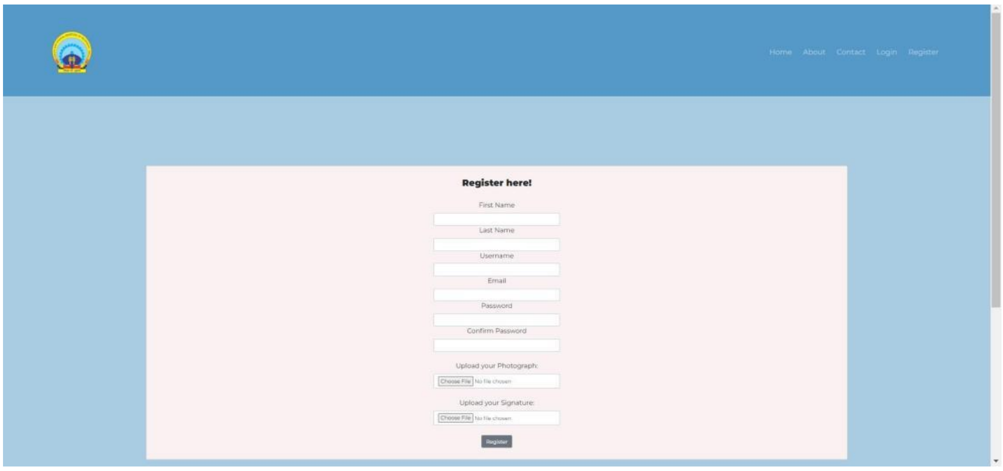
In this project, a website employing various authentication factors has been developed. It represents the possibilities a corporation can establish in order to progress the user protection and security of sensitive information. A triple layer of security has been set in motion - the first being a pass code, second signature validation and final layer of facial verification.

All developed models have a fairly high accuracy and users may access different features based on the authentication level cleared. This allows a principle of privilege to be established while maintaining user convenience.



The Multi-Level Biometric Authentication Framework

Fig20. Home Page of the Website



Register here!

First Name

Last Name

Username

Email

Password

Confirm Password

Upload your Photograph:

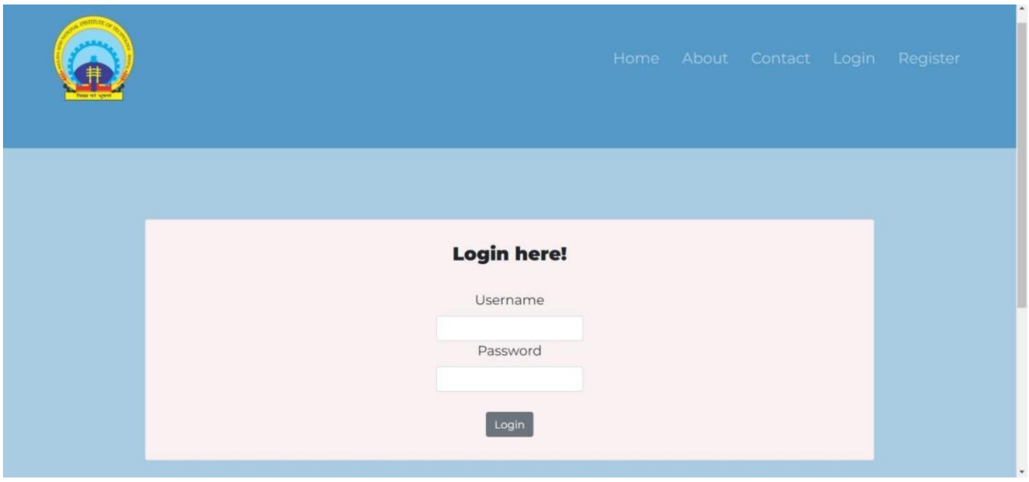
Choose File No file chosen

Upload your Signature:

Choose File No file chosen

Register

Fig21. Registration Page of the Website

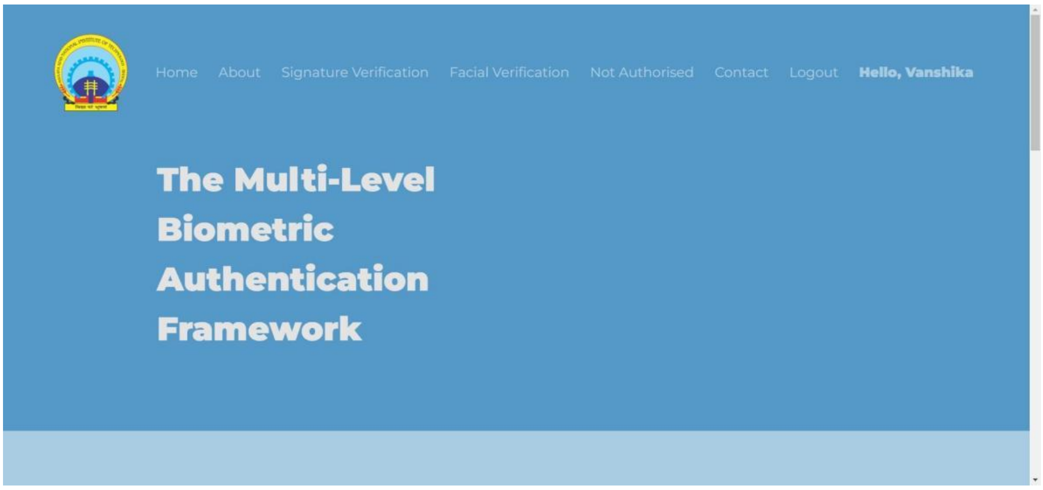


Login here!

Username

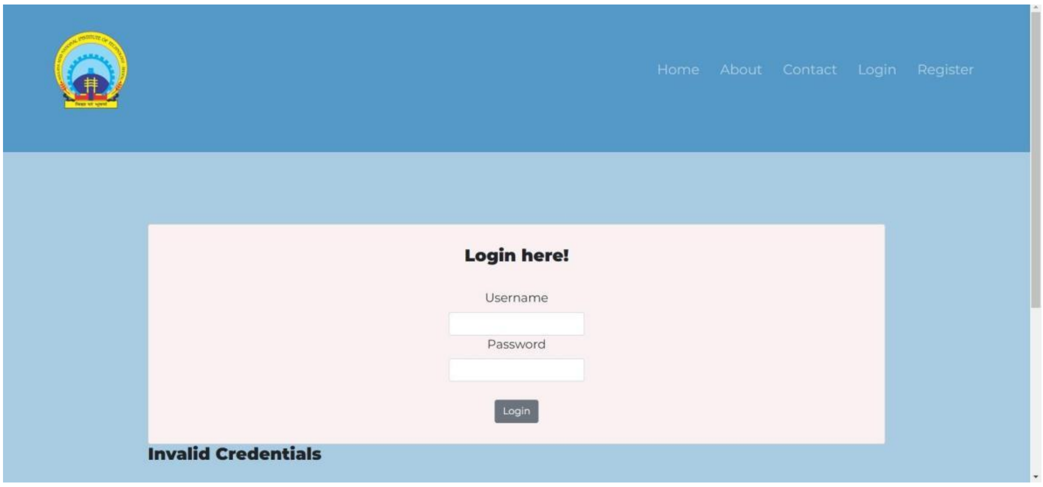
Password

Fig22. Login Page of the Website



The Multi-Level Biometric Authentication Framework

Fig23. Landing Page of the Website after login



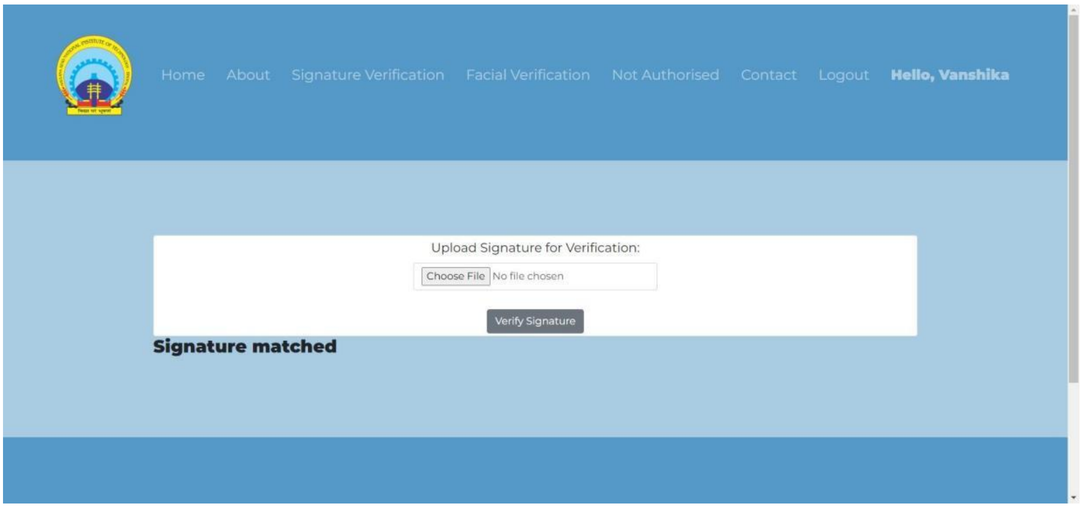
Login here!

Username

Password

Invalid Credentials

Fig24. Error Page after entering wrong login information



Upload Signature for Verification:

No file chosen

Signature matched

Fig25. Signature Verification result when it matches

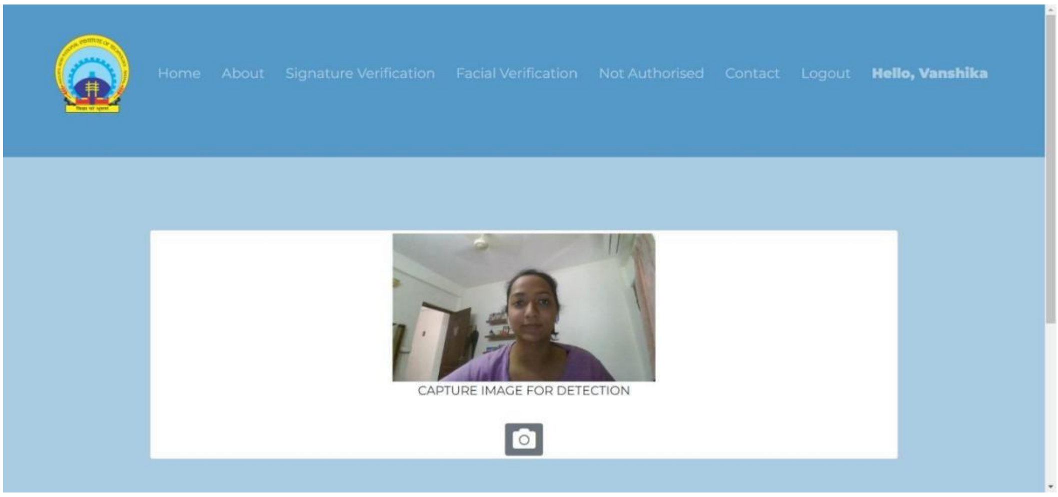
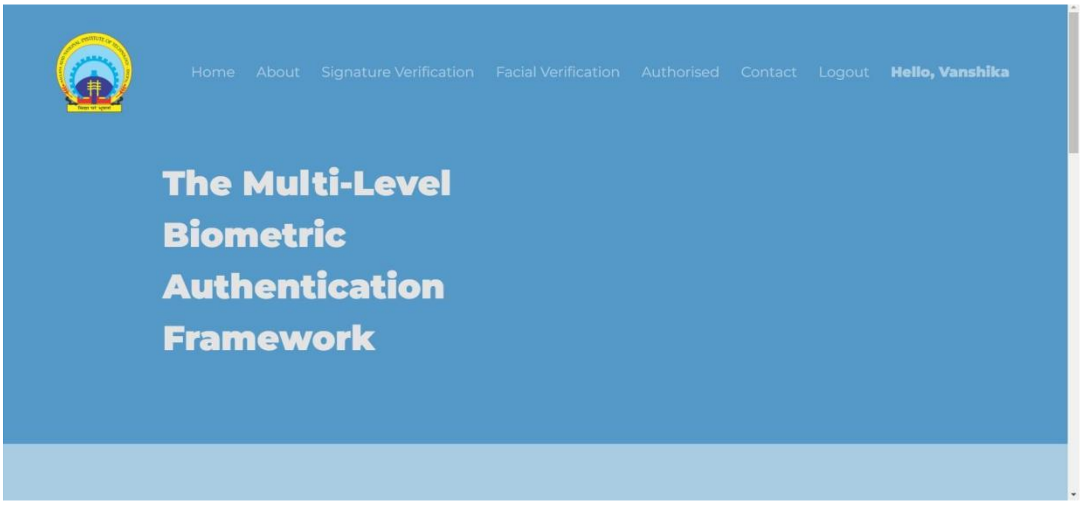


Fig26. Facial Verification page of the website



The Multi-Level Biometric Authentication Framework

Fig27. Homepage of Website if the user is verified

CONCLUSION AND FUTURE SCOPE

From the research done, the conclusion is security will remain a dominant feature in this digital age. A combination of the selected features provides us with a well-rounded authentication system which is highly secure and has control over access to privileged data.

The possibility to improve upon security is omnipresent. In this digital age, improvement of security has to be one step ahead of hackers. This authentication system has two factors which require personal knowledge from the user while the next factor is based on a feature that the user possesses and requires the physical presence for authentication.

To sum up this authentication system provides a secure level of validation by employing various biometric authentication methods. Over time, the model improves upon itself so as the database increases over time, simultaneously increasing sensitive data, the validation also becomes more unassailable.

The future scope of the project would be to solicit this biometric verification to protect particular information from user's access. Ease of user convenience can also be improved upon alongside a more robust database system. In-depth research for new features of security can be pursued to rival advances made by hackers.

REFERENCES

[1] Wei Tian, YizhengQiao and Zhiqiang Ma, “A New Scheme for Off-line Signature Verification Using DWT and Fuzzy Net”, 8th ACIS International Conference on Software Engineering, Artificial Intelligence, Networking, and parallel/Distributed Computing.

[2] Miguel A.Ferrer, Jesu’s B. Alonso, and Carlos M. Travieso, Offline Geometric Parameters for Automatic Signature Verification Using Fixed-Point Arithmetic IEEE transactions On Pattern Analysis And Machine Intelligence, vol. 27, No. 6, June 2005

[3] Alan McCabe, Jarrod Trevathan and Wayne Read, Neural Network-based Handwritten Signature Verification, Journal of computers, vol. 3, no. 8, August 2008.

[4] Eman Alajrami , Belal A. M. Ashqar, Bassem S. Abu-Nasser , Ahmed J. Khalil, Musleh M. Musleh, Alaa M. Barhoom, Samy S. Abu-Naser, Handwritten Signature Verification using Deep Learning, International Journal of Academic Multidisciplinary Research (IJAMR) Vol. 3 Issue 12, December – 2019, Pages: 39-44

[5] R. Kaur and E. Himanshi, "Face recognition using Principal Component Analysis*,"* 2015 IEEE International Advance Computing Conference (IACC)*,* Banglore*,* 2015*,* pp*.* 585-589*.*

[6] F. Z. Chelali, A. Djeradi and R. Djeradi, "Linear discriminant analysis for face recognition*,"* 2009 International Conference on Multimedia Computing and Systems, Ouarzazate, 2009, pp. 1-10.

[7] M. S. Bartlett, J. R. Movellan and T. J. Sejnowski, "Face recognition by independent component analysis," in IEEE Transactions on Neural Networks, vol. 13, no. 6, pp. 1450-1464, Nov. 2002.

[8] P. Viola and M. Jones, “Rapid object detection using a boosted cascade of simple features,” *Comput. Vis. Pattern Recognit.*, vol. 1, pp. 511-518, 2001.

[9] A. Krizhevsky, I. Sutskever, and G. E. Hinton, “ImageNet classification with deep convolutional Neural networks,” *Communications of the ACM*, vol. 60, issue 6, pp. 84-90, June 2017.

[10] K. Zhang, Z. Zhang, Z. Li *et al*., “Joint face detection and alignment using multi-task cascaded convolutional networks,” *Spl*, no. 1, pp. 1–5, 2016.