

Privacy-Preserving Iris Based Authentication System

by

Radha Agrawal

202111078

A Thesis Submitted in Partial Fulfilment of the Requirements for the Degree of

MASTER OF TECHNOLOGY

in

INFORMATION AND COMMUNICATION TECHNOLOGY

to

DHIRUBHAI AMBANI INSTITUTE OF INFORMATION AND COMMUNICATION TECHNOLOGY



May, 2023

Declaration

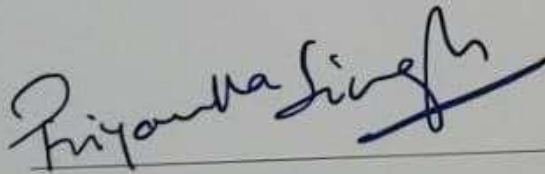
I hereby declare that

- i) the thesis comprises of my original work towards the degree of Master of Technology in Information and Communication Technology at Dhirubhai Ambani Institute of Information and Communication Technology and has not been submitted elsewhere for a degree,
- ii) due acknowledgment has been made in the text to all the reference material used.


Radha Agrawal

Certificate

This is to certify that the thesis work entitled "Privacy-Preserving Iris Based Authentication System" has been carried out by Radha Agrawal for the degree of Master of Technology in Information and Communication Technology at *Dhirubhai Ambani Institute of Information and Communication Technology* under my/our supervision.



Prof. Priyanka Singh
Thesis Supervisor



Prof. Manjunath Joshi
Thesis Co-Supervisor

Acknowledgments

I want to express my sincere gratitude to all those who have contributed to the completion of this project. I learned how to deal with challenging problems that come up in the field of computer science. I want to thank the Dhirubhai Ambani Institute of Information and Communication Technology for giving me this excellent opportunity.

Firstly, I would like to thank my supervisor, Prof. Priyanka Singh and Prof. Manjunath Joshi, for their valuable guidance, constructive feedback, and unwavering support throughout the project. Their expertise and insights have been instrumental in shaping the direction of my research.

I would also like to thank the participants who participated in the study and generously shared their time and experiences. Without their cooperation, this research would not have been possible.

Furthermore, I am grateful to my colleagues and friends for their encouragement, helpful suggestions, and assistance in various aspects of the project. Their support has been indispensable in making this project a success. Finally, I would like to acknowledge the support of my family. Their love and encouragement have constantly motivated and inspired me throughout this project.

Contents

Abstract	iv
List of Tables	v
List of Figures	vi
1 Introduction	1
1.1 Introduction	1
1.2 Organization	2
2 Preliminaries	3
2.1 Paillier encryption:	3
2.2 Different Types of Attacks:	4
3 Related Work	6
4 The Proposed Framework	9
4.1 Threat Model	9
4.2 Performing morphing on the proposed framework	15
4.3 Proposed Algorithms	18
5 Implementation and Results	21
5.1 Dataset	21
5.2 Experimental details of the proposed framework	21
5.3 Experimental details of Morph Attack	23
5.4 Comparative Analysis	26
6 Security Analysis	28
7 Conclusion	30
References	32

Abstract

Biometric authentication systems have gained immense popularity due to their ability to provide secure and convenient authentication. However, the leakage of sensitive biometric data can compromise an individual's privacy and security. To address this issue, a privacy-preserving biometric authentication system based on iris data is proposed in this paper. The framework exploits the homomorphic properties to process encrypted data, thereby ensuring the privacy of sensitive data, even while using the services of third-party cloud service providers (CSPs). In the initial stage of the experiment, we encrypt the data, and comparison was done by using hamming distance, but after completion of the first experiment, we realized that data can be morphed through an insecure channel by using multiple attacks to overcome this we have proposed framework were morphing is performed on the iris data by using a man-in-the-middle attack. Two iris identification Algorithms are proposed, with a success rate of over 60% and a false match rate of 5%, and are vulnerable to morph attacks. We also examine how comparable the original and morphed iris images must be. Using original images, we present our findings for morphing iris detection. The proposed privacy-preserving biometric authentication system offers a robust framework that minimizes time complexity compared to other state-of-the-art approaches. This framework ensures the privacy of sensitive data and provides a secure biometric authentication system.

List of Tables

4.1	Description of variables and functions used in the proposed method	11
5.1	Comparison of time required for each step (in seconds) of other state-of-the-art approaches with the proposed framework	23
5.2	DHVMR algorithms to iris morph attacks in terms of MAMR%.	24
5.3	SMR algorithm to iris morph attacks in terms of MAMR%.	24
5.4	Comparison of the state-of-the-art approaches with the Proposed scheme	26

List of Figures

4.1	Detailed steps of proposed framework	10
4.2	After performing Localization on scanned iris image	12
4.3	After performing pre-processing	13
4.4	Computing Hamming distance in plain-text domain	14
4.5	Computing Hamming distance in encrypted domain	14
4.6	Detailed steps of proposed framework	16
4.7	Morph iris images of left and right eyes using random substitution technique.	17
5.1	Analysis of steps for encryption of client’s iris code	22
5.2	Computing hamming distance between plain-text domain and en- rypted domain	22
5.3	Comparison of total computation time of other state-of-the-art ap- proaches with the proposed framework	23
5.4	Examples of original and morphed images.	24
5.5	Comparison of DHVMR algorithm with the original image on right iris images.	25
5.6	Comparison of SMR technique with the original image on right iris images.	25

CHAPTER 1

Introduction

1.1 Introduction

Biometric authentication systems have become increasingly popular in recent years, with many highly populated countries employing biometric systems for personal identification. The integration of crucial documents, such as voter IDs, passports, and driver's licenses, into apps like Digilocker, has minimized the need to carry physical identification documents and remember numerous passwords. However, the storage of these databases by third-party cloud service providers (CSPs) poses a threat to people's security and privacy, as a compromised database may lead to leaked information, including biometric traits. This information may be used to gain unauthorized access, such as by a criminal forging a morphed biometric trait to fool the authentication system.

To address this concern, privacy-preserving biometric systems are urgently needed so that even if the database is compromised, it does not lead to irreparable losses. The Turkey minister took steps to satisfy the needs of privacy-aware citizens by printing biometric passports, claiming to have 27 security features. Hernandez et al. proposed a biometric system for teachers and postgraduates with privacy-preserving features, while Guo et al. proposed a biometric system without touch to minimize the spread of Covid-19.

Most biometric encryption systems (BES) only provide security against specific attacks, which can limit the system's effectiveness. However, privacy-preserving BES are liable for key generation and key-binding, which does not include encryption, and they do not store biometric data. One such trait that is unique and involves no touch is the iris. Privacy-preserving systems based on iris were proposed, such as Devi et al. proposal to use N -th degree truncated polynomial ring (NTRU) homomorphic encryption (HE) to secure the iris template database. HE permits operations such as addition and multiplication to be performed directly on top of the encrypted data.

However, combining two individual's biometric images to create a morphed image can lead to a security concern for biometric systems. A single passport with a morphed face image can be associated with two individuals, as the resulting biometric template may match both identities. Landmark-based approaches are commonly used for image-level morphing techniques, but feature-level morphing employing minutiae points has also been suggested. To address the susceptibility of biometric systems to morphing, Rathgeb and Busch developed a feature-level morphing method for iris recognition in which stability-based bit substitution is used to change iris codes. Erdogan proposed a different method for iris morphing using normalized iris images, while we proposed a landmark-based scheme for iris morphing on unnormalized iris images.

The major contributions of the work are as follows:

- We have proposed a privacy-preserving biometric authentication system based on iris data that leverages the services of the CSP's for storage and authentication without compromising the user's privacy.
- The computational cost is also minimized compared to the other state-of-the-art approaches [2, 3, 7].
- Introducing a method for iris morphing at the image level using a random substitution technique.
- Assessing the susceptibility of two iris recognition algorithms to morph assaults using the IITD dataset, which is available publicly.
- Evaluating the similarity required between images for a successful morphed iris image.
- Presenting true match rate and false match rate.

1.2 Organization

The work is organized as follows: Chapters 2 and 3 present preliminaries and related work. Chapter 4,5 presents the proposed frameworks and their result. We present a security analysis in Chapter 6. Chapter 7 compares the proposed framework with other state-of-the-art, And finally, the conclusion.

CHAPTER 2

Preliminaries

This chapter briefly overviews the Paillier homomorphic encryption and potential attack scenarios: poison attack, man-in-the-middle attack, SQL injection attacks, insider threat attack, and frequency analysis attack.

2.1 Paillier encryption:

The Paillier cryptosystem is an additive asymmetric partial homomorphic encryption scheme. It consists of the following three main phases:

1. Generation of public-private key pair.
2. Encryption of a message.
3. Decryption of a message.

Prior to evaluating the public and private keys, the first two large prime numbers, p and q are chosen. Public key n and g are calculated as follows:

$$n = p1 \times q1 \tag{2.1}$$

$$g = n + 1 \tag{2.2}$$

The private keys λ and μ are computed as follows:

$$\lambda = lcm(p - 1, q - 1) \tag{2.3}$$

$$\mu = mod(\alpha, n) \tag{2.4}$$

Here, *lcm* implies the least common multiple and *mod* denotes the modulo operator.

With the public (n, g) and private (λ, μ) keys generated, now we describe how to encrypt and decrypt a message.

Encryption:

A plain-text message m can be encrypted as follows:

1. Let m be a message to be encrypted where $0 \leq m \leq n$.
2. Select random r where $0 < r < n$
3. Compute cipher-text as $C = g^m r^n \text{ mod } n^2$.

Decryption:

A ciphertext message C is decrypted as follows:

1. Compute plain-text message as:

$$I = L(c^\lambda \text{ mod } n^2) \cdot \mu \text{ mod } n \tag{2.5}$$

Homomorphic property:

Let cipher-texts values $c1$ and $c2$ are the encrypted value of messages $msg1$ and $msg2$. The homomorphic property is:

1. The product of two cipher-texts $c1$ and $c2$, decrypt to the sum of their corresponding plain-texts, $msg1$ and $msg2$.

$$D((c1 * c2 \text{ mod } n^2); \lambda, \mu, n) = (msg1 + msg2) \text{ mod } n. \tag{2.6}$$

2. A ciphertext $c1$ raised to the scalar value s of plaintext and decrypt to the product of plain-texts.

$$D((c1^s \text{ mod } n^2); \lambda, \mu, n) = (msg1 * s) \text{ mod } n. \tag{2.7}$$

2.2 Different Types of Attacks:

In this section, we have discussed the different types of attacks.

Poison Attacks:

An assault that uses poison combines an erasing attack with a copycat attack fraud.

- Erasure attack: The adversary deletes the original data from the server to obstruct user communication.
- Duplicate faking attack In a duplicate faking attack, bogus data is substituted for the original data on the server.

In both scenarios, the real data is lost by the legitimate user, breaching data integrity.

Man-In-The-Middle Attacks:

A cyber attack is a "man-in-the-middle" attack where an unauthorized user secretly intercepts a conversation between two authenticated users without either party's knowledge. When two users communicate over an un-encrypted channel, this attack is feasible.

SQL injection Attacks:

SQL injection, also called SQLI, is a popular attack method that uses malicious SQL code to manipulate backend databases and access data that was not meant to be displayed. Any number of things, such as private information, user lists, or sensitive corporate data, may be included.

Insider Threat Attacks:

A cyber security danger that comes from within an organisation is referred to as an insider threat. It often happens when a current or former employee, contractor, vendor, or business partner who has access to the organization's networks, systems, and data abuses their access. Insider threats might be carried out purposefully or accidentally. Whatever the motivation, compromised enterprise systems and data integrity, confidentiality, and/or availability are the ultimate results. Most data breaches are the result of insider threats.

Frequency analysis attack:

The perpetrator keeps track of the frequency of the messages exchanged between users. He can determine that repeated transmission of the same ciphertext refers to the same initial message. Consider the case where the ciphertext C that corresponds to the original message M is sent. An attacker can determine that ciphertext C corresponds to the same message M if the attacker notices that ciphertext C is sent more than once.

CHAPTER 3

Related Work

Kaur et al. collected iris data and ensured to maintain the good quality of the data [8]. However, ensuring that the technique could handle the challenges regularly encountered in the acquisition process was imperative. It generally involves challenges, such as hazy images, camera diffusion, noise, light reflection, and other elements that may impact the segmentation process [9]. Li et al. proposed to secure fingerprint photographs by applying a unique chaotic fingerprint image encryption strategy, integrating shuttle operation and a nonlinear dynamic [6].

Jan et al. developed biometric personal identification techniques, including ear and finger knuckle measurements [15]. Zhao et al. proposed that finger knuckle bending's image pattern was incredibly distinctive and full of different texture patterns [14]. It demonstrated the effectiveness of various biometric systems that change depending on the application. Patsakis et al. proposed a privacy-preserving biometric authentication based on iris images and used NTRU homomorphic encryption for encrypting iris images [2]. The protocol successfully held a lot more information without compromising its security. Though the overall capacity was lowered, the post-quantum era ensured that the protocol was secure.

Khan et al. provided Dougman's rubber sheet model for iris images, and it has been widely used in industries to normalize data [17]. A newer technique called image registration has been proven to be more effective. Although normalization addresses pupil dilation issues, it may not always be correct since surface patterns could vary from person to person [10]. The subsequent step was feature extraction, in which distinct aspects of the normalized image were recovered and stored as a biometric template. Only important features should be encoded to compare two templates more confidently and properly. Nithyanandam et al. proposed two main comparison techniques for comparing the templates [10]. Comparing two templates made from several irises fell under inter-class comparisons. The so-called intra-class comparison yield a varied range of values for the same com-

parison [5].

The tactic worked incredibly well for preserving privacy in financial situations when the transactions were primarily linked to addition or subtraction operations on the balance amount [7]. Shankar et al. proposed using the hamming distance based on the XOR operator to distinguish between any corresponding bits [12]. The bits must not be affected by outside factors, such as eyelids, eyelashes, inconsistent lighting, or different types of noises. Then, the appropriate hamming distances were calculated and normalized.

Sharma et al. proposed an adversary can trick an iris recognition system into accepting a morphing image as a real identity [16]. By combining two distinct iris images using computer vision algorithms, the assault is carried out. Using two publicly accessible iris recognition datasets, the authors assess the effectiveness of the attack and show that the modified photos may frequently outsmart the detection algorithms. The study emphasizes the need to strengthen iris recognition systems to prevent such assaults.

Scherhag et al. proposed examining the vulnerability of face recognition systems to morphed face attacks. In these attacks, a hacker fabricates a digital composite of two separate faces to trick the system [4]. The scientists discovered that the success rate of these attacks could reach 100% and are very likely to affect current facial recognition systems. In addition to a framework for assessing face recognition systems' susceptibility to morphing face attacks, the study makes a case for the need for more secure and reliable face recognition systems. Rathgeb et al. have developed the feasibility of constructing morphed iris-codes and looked into creating a morphed iris-code, a composite of two separate iris-codes [11]. The scientists used openly accessible iris recognition systems in their tests and discovered that morphing iris codes could be successfully constructed and used to trick these systems. They recommend that countermeasures such as random challenges and the usage of several iris recognition algorithms can be used to strengthen the security of iris recognition systems. Gomez-Barrero et al. proposed altered biometric data to forecast how susceptible biometric systems are to attacks. A morphing dataset is created, features are extracted from the morphed photos, and a machine learning classifier is trained to distinguish between real and morphed biometric data. The findings show that the suggested method can accurately and, with a low percentage of false positives, estimate the susceptibility of biometric systems to morphing assaults [13].

Wang et al. discuss research into the advancement of techniques for rating the quality of digital photos. The author introduces a novel method called the Struc-

tural Similarity Index (SSIM), which compares the structural data of two photos to determine their similarity. The SSIM measure, according to the authors, is superior to earlier ones because it considers how sensitive the human visual system is to changes in structural information [18]. It also reviews how image quality assessment has evolved, emphasizing the importance of understanding human perception to create efficient image evaluation techniques. Basit et al. describe a novel iris recognition strategy integrating several image processing methods to boost iris detection's precision and effectiveness. The suggested approach combines a brand-new iris segmentation algorithm and a Gabor filter-based feature extraction technique. The proposed method has demonstrated great accuracy and speed compared to other iris recognition techniques, making it a promising option for human identification in various applications [1].

CHAPTER 4

The Proposed Framework

This chapter gives a detailed description of the proposed frameworks. It also discusses the involved entities and their specific roles.

4.1 Threat Model

This section presents the details of the involved entities. This model has four entities: Staff, Bio-metric expert, CSP, and Verifier. Communication between the staff and the bio-metric expert is secure, and communication between CSP and the verifier is insecure.

Staff:

- The staff entity scans the client's iris. Staff provides a scanned iris image to a biometric expert for pre-processing the iris image. The staff is considered an honest entity.

Bio-metric Expert:

- On receiving the scanned iris image from the staff, the biometric expert performs pre-processing of the image by localization, normalization, and feature extraction, as shown in Figure 5.6. After performing pre-processing, obtained iris code is transferred to the encryption block for encrypting the client's iris code. Later encryption of the encrypted iris code of one block moves to the cloud services provider for computing hamming distance. The bio-metric expert is considered an honest entity.

CSP-Cloud Service Provider:

- The client's encrypted iris code of one block is provided to CSP. CSP computes hamming distance(HD) of the stored database of encrypted iris code and the client's encrypted code. HD is compared to the set threshold (threshold=5.0), and a decision is transferred to the verifier. The CSP is considered a semi-honest entity.

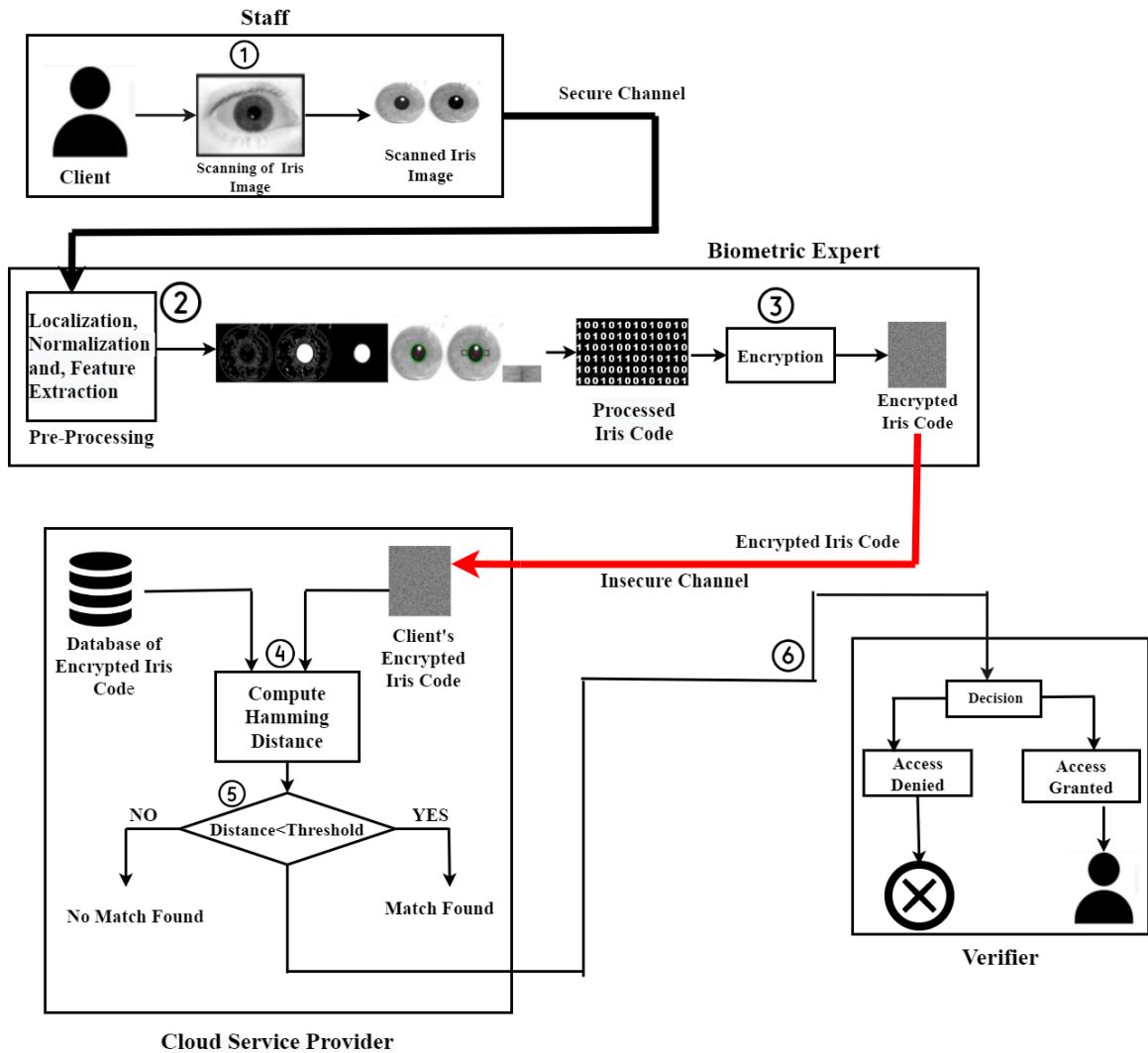


Figure 4.1: Detailed steps of proposed framework

Verifier:

- CSP provides a decision to the verifier and then transmits the decision to the client, whether he is accepted or not. If the decision is accepted, the access is granted to the client, else the access is denied, and he might be a third party. The verifier is considered a semi-honest entity.

Step 1: The client comes to the staff entity to log in to the system; the iris is scanned by staff. After scanning the iris, the image is provided to a Biometric expert.

Step 2: The Bio-metric expert received a scanned iris image from staff and performs pre-processing on the image as follows:

Table 4.1: Description of variables and functions used in the proposed method

Abbreviation	Definition
Enc()	Encryption of iris code
Dec()	decryption of iris code
Count	Count ("1") for comparison
\ominus	Compressed the image
X	original image
P	pixel
Z	Binary image 2 dimensions z*Z
r	the interval [0,1]
x and y	size of image
λ and θ	$\frac{1*\pi}{4}$
γ	0.4
ϕ	0
DHVMR	Difference Hash Value Match Rate
SMR	Similarity match Rate
MAMR	Modify Attack Match Rate
TMR	True Match Rate
FMR	False Match Rate

1. Localization.
2. Normalization.
3. Feature Extraction

The above three steps are described in the below section

Localization:

Finding the iris's precise location and contour in an image as:

1. Edge Detection Process - Canny Method.
2. Gaussian Filter Kernel Example (5 x 5)
3. Gradient intensity
4. Filling all the connected region
5. Pixel are removed whose pixel size is less than 80 pixel

Normalization:

The annular iris area is transferred to a dimensionless pseudo-polar coordinate system:

1. The morphological processing (Erosion and then Dilation)

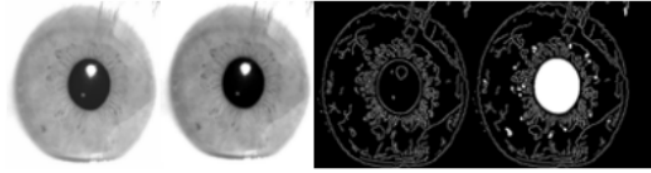


Figure 4.2: After performing Localization on scanned iris image

2. Dilation: Expand the image by the following formula:-

$$X \oplus B = \{P \in Z * Z | P = x + b, x \in X, b \in B\} \quad (4.1)$$

3. Erosion: Shrink the image

$$A \ominus B = \{z \in E | Bz \subseteq A\} \quad (4.2)$$

4. Daugman's Rubber Sheet Model

$$I(x(r, \theta), y(r, \theta)) \rightarrow I(r, \theta) \quad (4.3)$$

where, r is on the interval $[0,1]$

$$x = r \cos(\theta) \quad (4.4)$$

$$y = r \sin(\theta) \quad (4.5)$$

Feature Extraction:

We used the Log Gabor for the feature extraction mechanism to get the iris code. This algorithm requires 10 1-D signals, with the first eye's pupil and the last eye's sclera area, as input.

1. Log Gabor:

$$g = \exp\left[\frac{x^2 + \gamma * y^2}{2 * \sigma^2} \exp \frac{2\pi * x + \phi}{\lambda}\right] \quad (4.6)$$

Step 3: The processed iris code is sent to the encryption block to encrypt the client's iris code using paillier homomorphic encryption. After encryption, the

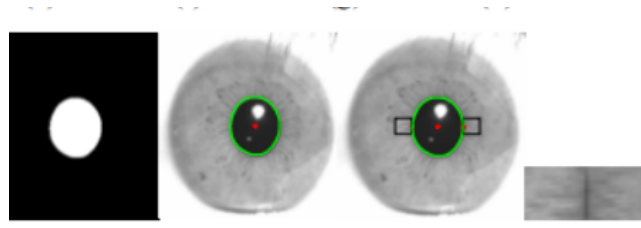


Figure 4.3: After performing pre-processing

encrypted iris code is transmitted to CSP.

Plaintext Domain

1. Let's client scanned iris code be $Iris_c$
2. Let's store database of iris code be $Iris_d$
3. Compute the difference between $Iris_c$ and $Iris_d$ using homomorphic property and stored in variable $Diff$.

$$Diff = (Iris_c + ((-1)Iris_d)) \bmod n \quad (4.7)$$

$$Diff = (Iris_c - Iris_d) \bmod n \quad (4.8)$$

4. Calculate hamming distance of $Diff$ and counting the "1" in $Diff$ variable by using count function and storing in result in h_d .

$$h_d = Diff.count("1") \quad (4.9)$$

As shown in Figure 4.4. the computing hamming distance in the plain text where $Iris_c$ and $Iris_d$ by using the paillier homomorphic property as shown in equation (8) and storing it in the $Diff$ variable [0011]. Compute hamming distance of $Diff$ by counting ("1"). Stored it in h_d as shown in equation (9). Compared with the set threshold and thus, the client gets access.

Encrypted Domain

1. To encrypted $Iris_c$ and $Iris_d$ two prime numbers $p1 = 17, q1 = 19$ are set with, g as generator, r as random numbers

$$\widetilde{Iris}_c = g^m r^n \bmod n^2 \quad (4.10)$$

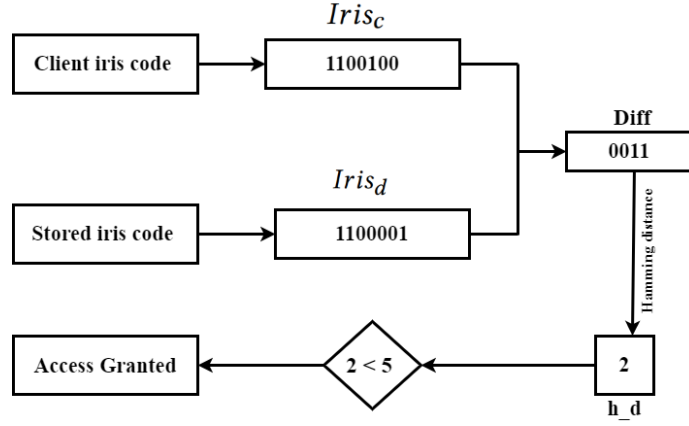


Figure 4.4: Computing Hamming distance in plain-text domain

$$\widetilde{Iris}_c = Enc(Iris_c) \quad (4.11)$$

$$\widetilde{Iris}_d = Enc(Iris_d) \quad (4.12)$$

$$\widetilde{Diff} = \widetilde{Iris}_c * \widetilde{Iris}_d^{(-1)} \bmod n^2 \quad (4.13)$$

2. Dec() function is used for decryption.

$$Decrypt = Dec(\widetilde{Diff}) \quad (4.14)$$

3. The CSP computes hamming distance (h_d) by counting "1" in Decrypt variable.

$$h_d = Decrypt.count("1") \quad (4.15)$$

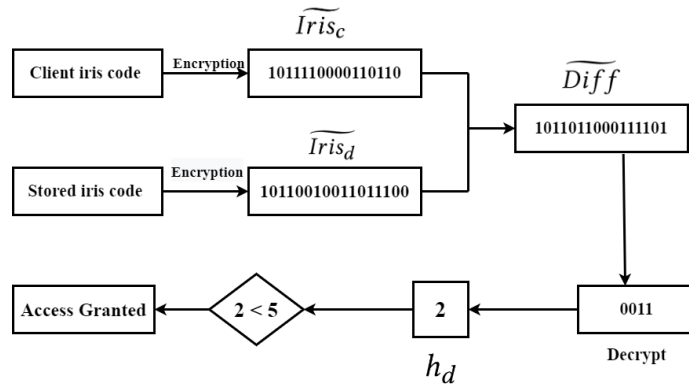


Figure 4.5: Computing Hamming distance in encrypted domain

As shown in Figure 4.5, the computing hamming distance in the encrypted domain where \widetilde{Iris}_c and \widetilde{Iris}_d are encrypted by using the encryption function. Com-

pute \widetilde{Diff} using the homomorphic property as shown in equation (10). Now, decrypt the \widetilde{Diff} using the decryption function and store it in the decrypt variable. Compute hamming distance of the decrypt by counting ("1"). Store in h_d and compare the result with the set threshold, and the client gets access.

Step 4: Then Hamming distance is compared to threshold δ and decision is transferred to verifier.

```
if  $h_d < \delta$  then Access granted  
else Access denied  
end if
```

Step 5: The verifier receives the decision and forwards it to the client; they can log in if the client is authenticated. Else access denied.

4.2 Performing morphing on the proposed framework

This model has five entities: Employer, Interceptor, Manager, CSP, and Validator.

The Employer, Manager, CSP, and Validator perform the same task as performed in 4.1. But the interceptor is an adversary and performs the role of a man-in-the-middle attack. He tries to fetch the original iris image through an insecure channel and perform morphing on the image. Morphing is performed by random substitution technique; then, the morphed iris image is created. This morphed iris image is provided to the manager through an insecure channel to extract the hash value. The modifier expert is considered a malicious entity.

The CSP receives the iris image hash values from the manager. The CSP compares hash values obtained from the manager and stored in the database by using two algorithms, DHVMR and SMR. The results of the algorithm are transferred to the accuracy detection block. The accuracy detection block detects accuracy in percentage by True Match Rate (TMR - 60%) and False Match Rate (FMR - 5%). The CSP is considered an honest entity.

Step 1: The client goes to the employer entity to log into the system; the employer scans the client's iris. After scanning the iris, the image is provided to the manager.

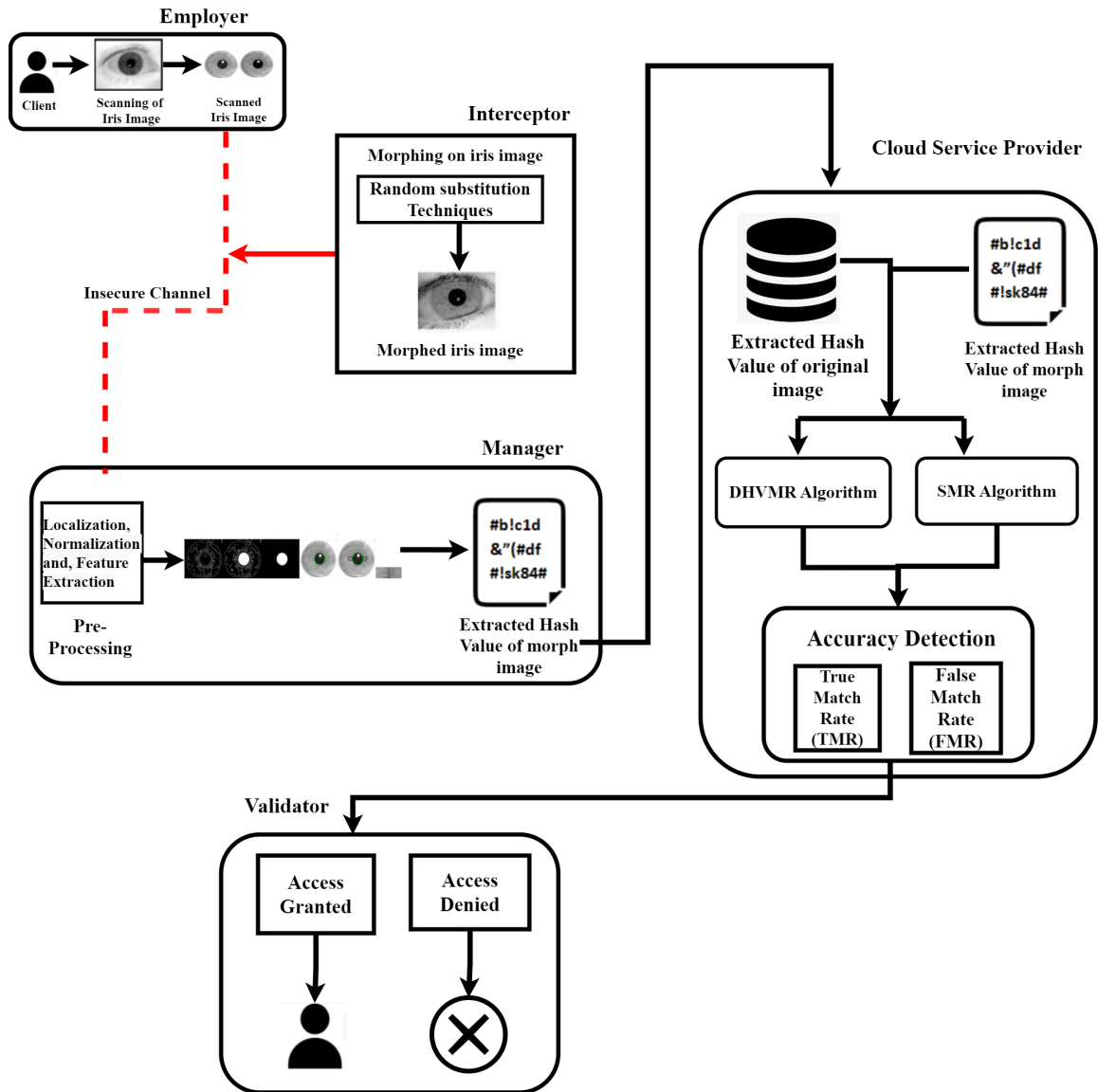


Figure 4.6: Detailed steps of proposed framework

Step 2: The interceptor performs a man-in-middle attack and tries to collect the iris image from the employer to perform morphing on the image using a random substitution technique. The total number of left and right iris images is 2,240. We successfully morphed the image and collected 48,916 morph iris images, as shown in Figure 4.7.

Step 3: The interceptor provides the morph iris image to the manager for extracting the hash values.

Step 4: The manager performs pre-processing using localization, normalization, and feature extraction, and then the hash value is extracted.

1. Localization: Finding the iris's precise location and contour in an image.

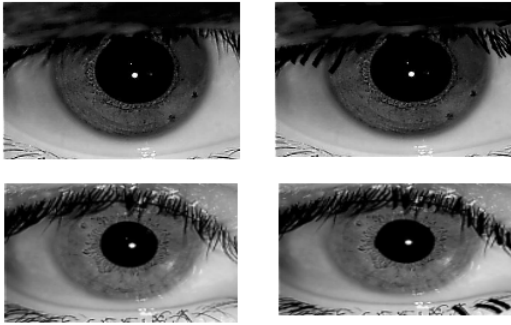


Figure 4.7: Morph iris images of left and right eyes using random substitution technique.

2. Normalization: The annular iris area is transferred to a dimensionless pseudo-polar coordinate system.
3. Feature Extraction: We used the Log Gabor for the feature extraction mechanism to get the iris code. This algorithm requires 10 1-D signals, with the first eye's pupil and the last eye's sclera area, as input.

Step 5: The extracted hash value is provided to the CSP for comparison.

Step 6: The CSP fetches the original iris image hash value from the stored database. Then performs a comparison of original and received hash values using the two algorithms, DHVMR and SMR. DHVMR is used for the comparison of hash values. And SMR algorithm is used to find the similarity between the original and morph iris image.

Step 7: After the comparison, using algorithms, the result is sent to the accuracy detection block for detecting true and false match rates.

Step 8: The MAMR technique sets the threshold for TMR and FMR for accuracy detection. According to the MAMR threshold, we have set the threshold for the True match rate (60%) and the False match rate (5%). According to the threshold, results are generated.

Step 9: If the True match rate (60%) is less than the set threshold, then the image is morphed into a very less percentage, and if the False match rate (5%) is greater than the set threshold then the image is morphed in high percentage.

Step 10: The validator receives the decision and forwards it to the client; they can log in if the client is authenticated. Else access denied.

4.3 Proposed Algorithms

This subsection represents the algorithms for the proposed frameworks.

Algorithm 1 Random Substitution Technique *RST*

INPUT: Original Iris Image(O_{II}) of size $N_1 \times N_2$

OUTPUT: Morphed Image(M_I) of size $N_1 \times N_2$

- 1: Read original image in *img*
- 2: Take input h_i, w_i, c_l
- 3: $h_f = w_i \div 2$
- 4: $left_{part} = img[:, : h_f]$
- 5: $right_{part} = img[:, h_f :]$
- 6: print the $left_{part}$ and $right_{part}$
- 7: $h_{f2} = h \div 2$
- 8: $t_p = img[h_{f2}, :]$
- 9: $b_m = img[h_{f2} :, :]$
- 10: saving all the images $left_{part}, right_{part}, t_p, b_m$
- 11: $block = random.img(left_{part}, right_{part}, t_p, b_m)$
- 12: $block_{top} = block_{location}[X]$
- 13: $block_{left} = block_{location}[X]$
- 14: $block_{bottom} = block_{top} + block_{size}[0]$
- 15: $block_{right} = block_{left} + block_{size}[1]$
- 16: **for** i in range($block_{top}, block_{bottom}$): **do**
- 17: **for** j in range($block_{left}, block_{right}$): **do**
- 18: $image[i][j] = color$
- 19: **end for**
- 20: **end for**
- 21: *save_image*

Algorithm 2 Difference Hash Value Match Rate *DHVMR*

INPUT: Original Iris Image (O_I) and Morphed iris image (M_I)

OUTPUT: Hash value differences of (O_I) and (M_I)

```
1: Read original image ( $O_I$ )
2: Read morphed image ( $M_I$ )
3:  $Hash_{O_I} = cv2.(O_I) - hash.BlockMeanHash - create()$ 
4:  $HValue_{O_I} = hsh.compute(O_I)$   $\triangleright$  Hash value for original iris image
5:  $Hash_{M_I} = cv2.(M_I) - hash.BlockMeanHash - create()$ 
6:  $HValue_{M_I} = hsh.compute(M_I)$   $\triangleright$  Hash value for morphed iris image
7: Create two empty list  $temp_{arr}, n_w$ 
8: for  $i$  in range(len( $HValue_{M_I}$ )) do
9:   for  $j$  in range(len( $HValue_{M_I}[i]$ )) do
10:    if  $HValue_{M_I}[i][j] \neq HValue_{O_I}[i][j]$  : then
11:       $temp_{arr}.append(HValue_{M_I}[i][j])$ 
12:    end if
13:     $n_w.append(temp_{arr})$ 
14:  end for
15: end for
16: count = 0
17: for element in  $n_w$  do
18:   count += len(element)
19: end for
20: return count
```

Algorithm 3 Similarity match Rate *SMR*

INPUT: Original Iris Image (O_I) and Morphed iris image (M_I)**OUTPUT:** Similarity between images block and pixels. (S_{block}) and (p_{ixel})

```
1: Read original image ( $O_I$ )
2: Read morphed image ( $M_I$ )
3:  $Orimg_{block_{top}} = block_{location}[X]$ 
4:  $Orimg_{block_{bottom}} = block_{location}[X]$ 
5:  $Orimg_{block_{left}} = block_{top} + block_{size}[0]$ 
6:  $Orimg_{block_{right}} = block_{left} + block_{size}[1]$ 
7:  $mopimg_{block_{top}} = block_{location}[X]$ 
8:  $mopimg_{block_{bottom}} = block_{location}[X]$ 
9:  $mopimg_{block_{left}} = block_{top} + block_{size}[0]$ 
10:  $mopimg_{block_{right}} = block_{left} + block_{size}[1]$ 
11: for  $i$  in range(len( $mopimg_{block}$ )) do
12:   for  $j$  in range( $mopimg_{block}[i]$ ) do
13:     if  $mopimg_{block}[i][j] == Orimg_{block}[i][j]$  : then
14:        $temp_{arr}.append(mopimg_{block}[i][j])$ 
15:     end if
16:      $S_{block}.append(temp_{arr})$ 
17:   end for
18: end for
19: count = 0
20: for pixel in  $S_{block}$  do
21:   count += len(element)
22: end for
23: return  $P_{ixel}$  and  $S_{block}$ 
```

CHAPTER 5

Implementation and Results

This chapter presents the details of our experiments to validate the proposed frameworks.

5.1 Dataset

We run experiments with publicly accessible iris datasets, an open-source toolkit from the University of Schalzburg, Germany, that has been utilized for extracting iris codes from client iris images. The 2,240 iris images from 224 participants comprise the IITD iris dataset. There are 10 iris images per subject (5 left and 5 right). The images are captured using JIRIS, JPC1000, and digital CMOS sensors. Between 14 and 55 years old are the individuals in the dataset. There are 48 women and 176 men in the dataset. These images have a 320×240 -pixel resolution.

5.2 Experimental details of the proposed framework

Pre-processing steps are performed on scanned iris images, localization, normalization, and feature extraction. Processed iris code is transferred for encryption using paillier homomorphic encryption, a partially asymmetric homomorphic encryption scheme. Encryption is performed by computing LCM, inverse modulo, and cipher value compared in Figure 5.1. This comparison shows that the time required in each step for encrypting the iris code is less than NTRU encryption. After the encryption of iris codes, validation is carried out of two iris codes, i.e., of stored encrypted iris code and the client's iris code, and hamming distance is computed. For verification, we have set a threshold ($\delta = 5.0$), so if the h_d is smaller than the threshold, then the client receives access granted; otherwise, access is denied.

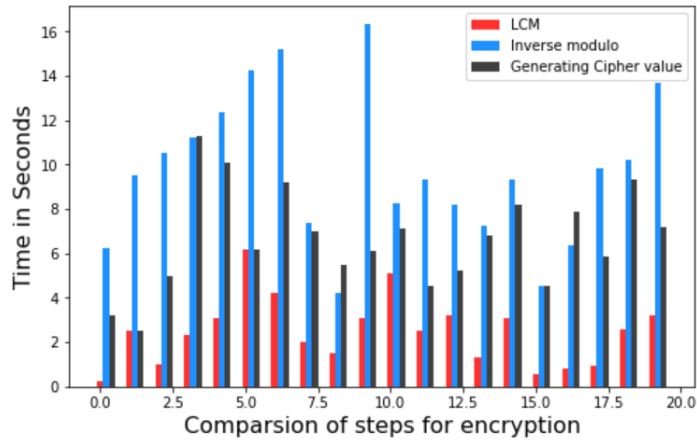


Figure 5.1: Analysis of steps for encryption of client’s iris code

As shown in Figure 5.1. Comparison of steps such as LCM, inverse modulo, and generating cipher value for the client’s iris code of 20 iris images. LCM, Inverse modulo, and cipher value are required for encryption of iris code, so here we are comparing three steps to show that the time required to encrypt iris code with paillier encryption is less than the NTRU encryption.



Figure 5.2: Computing hamming distance between plain-text domain and encrypted domain

As shown in Figure 5.2. Comparison of computing the hamming distance between scanned iris image(plain) and encrypted iris code for 15 iris images shows that encrypted iris code required more time than plain text.

Figure 5.3, and Table 5.1 show the time required in seconds for each step to encrypt the client’s iris code, comparing the state-of-the-art approaches with the proposed framework for 10 different iris images. The proposed framework re-

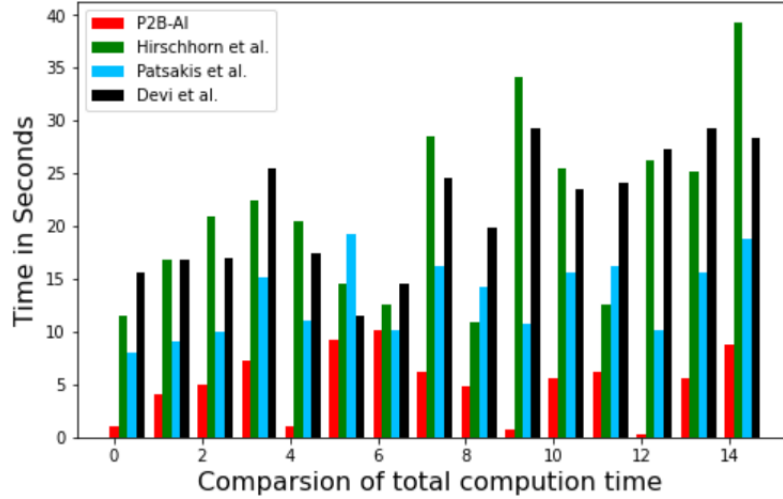


Figure 5.3: Comparison of total computation time of other state-of-the-art approaches with the proposed framework

Table 5.1: Comparison of time required for each step (in seconds) of other state-of-the-art approaches with the proposed framework

Parameters	Hirschhorn et al. [7]	Patsakis et al. [2]	Devi et al. [3]	Proposed framework
LCM	2.52 mins	3.2 mins	2.63 mins	1.2 mins
Inverse modulo	7.8 mins	6.63 mins	5.89 mins	5.2 mins
Encryption	16.2 mins	15.5 mins	13.5 mins	10.2 mins
Hamming distance	20.56 mins	15.23 mins	15.98 mins	12.36 mins

quires less time for each step compared to other approaches.

5.3 Experimental details of Morph Attack

We evaluate the robustness of two iris identification algorithms against morphed iris images. The first is the best-performing difference hash value match rate (DHVMR) algorithm. DHVMR algorithm compares extracted hash values from the CSP and morphed iris images. The similarity match rate (SMR) measures the similarity between iris codes.

Several examples of the original images and morphed images of left and right iris images created using IITD datasets are illustrated in Figure 5.4. Using IITD datasets, the recognition performance of these two algorithms is evaluated. Using the datasets from the IITD, we produce morphed iris images. The dataset includes 224 left-eye classes and 224 right-eye classes. We randomly select one image from each class to generate the morphs, resulting in 48,916 ($224C2 + 224C2$) altered im-

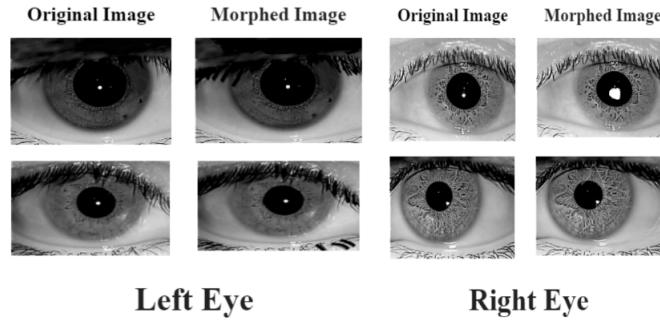


Figure 5.4: Examples of original and morphed images.

ages. Because landmarks in certain images with partial could not be seen, a total of 48916 morphs were created. The two iris recognition methods (morph attack) to the morphed iris images and evaluate their vulnerability using the Modify Attack Match Rate (MAMR). The MAMR determines the ratio of successful morph attacks to all morph attacks—the True Match Rate (TMR) for the DHVMR and SMR at 60%. False Match Rate (FMR) is 5% to measure the accuracy of the morph images.

The results of this evaluation, in terms of the MAMR technique at various thresholds (5% FMR), are presented in Table 5.2 and (60% TMR), are presented in Table 5.3.

Table 5.2: DHVMR algorithms to iris morph attacks in terms of MAMR%.

Algorithm	MAMR % - False Match Rate					
	Left Eye			Right Eye		
No. of image	1	2	3	1	2	3
DHVMR Algorithm	4.4	0.32	3.8	2.5	9.6	2.8

The DHVMR algorithm is used to find the difference between the original and morph iris images, and the false match rate for three left and right iris images is shown in Table 5.2. If the FMR is ($> 5\%$), then the morphing on a particular image is significantly high. We can see that the right eye 2nd image FMR is greater than the set threshold, so the morphing on the particular iris image is in high percentage.

Table 5.3: SMR algorithm to iris morph attacks in terms of MAMR%.

Algorithm	MAMR % - True Match Rate					
	Left Eye			Right Eye		
No. of image	1	2	3	1	2	3
SMR Technique	57.6	99.2	64.0	76.8	64.0	73.6

The SMR algorithm is used to find the similarities between the original and morph iris images, and the true match rate for three left and right iris images is shown in Table 5.3. If the TMR is ($< 60\%$), then the morphing on a particular image is significantly less. We can see that the left eye 1st image TMR is smaller than the set threshold, so the morphing on the particular iris image is in significantly less in percentage.

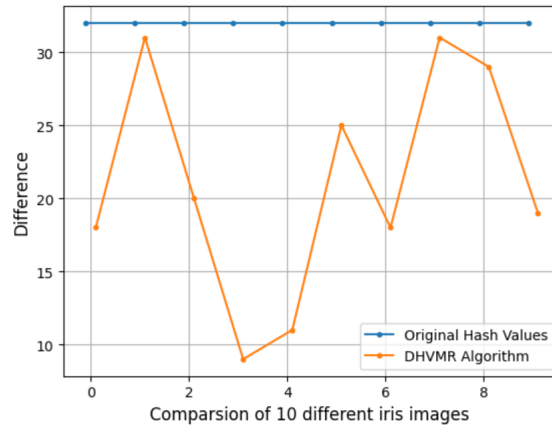


Figure 5.5: Comparison of DHVMR algorithm with the original image on right iris images.

The analysis of comparing the original iris image hash value with the morphed iris image hash value using the Dispute Value Match rate technique of 10 different left iris images is shown in Figure 5.5. We can see a high difference between the original and morphed images.

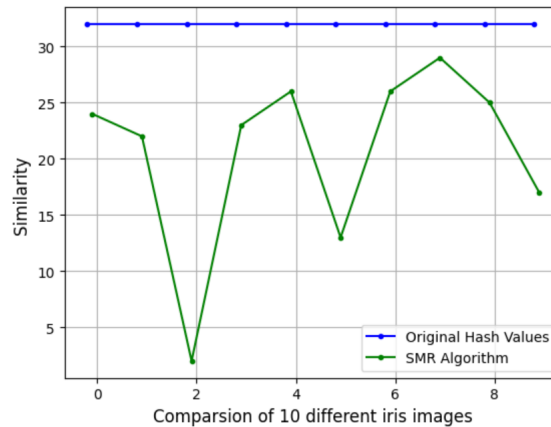


Figure 5.6: Comparison of SMR technique with the original image on right iris images.

We use similarity analysis to compare the original and morphed iris images using the similarity match rate of 10 different right iris images, as shown in Figure 5.6. We can see a significant difference between the original and morphed images.

5.4 Comparative Analysis

In this section, we have compared our proposed framework with other state-of-the-art as shown in Table 5.4.

Table 5.4: Comparison of the state-of-the-art approaches with the Proposed scheme

Factors	Hirschhorn et al. [7]	Patsakis et al. [2]	Devi et al. [3]	Proposed
Cryptosystem based on	N-th degree truncated polynomial ring (NTRU)	N-th degree truncated polynomial ring (NTRU)	N-th degree truncated polynomial ring (NTRU)	Paillier Homomorphic encryption
Accuracy for generating result	70%	78%	85%	96%
Storage	Occupy at most 20 GB	Occupy at most 15 GB	Occupy at most 17 GB	Occupy at most 10 GB
Total Computation time required for experiment	$O(n^2)$	$O(n^2)$	$O(n^2)$	$O(n)$
Robust against	secure against frequency analysis	secure against poison attacks	secure against SQL-injection attacks	secure against poison attacks, man-in-the-middle, frequency analysis, insider attacks, SQL-injection attacks

A comparative study of the proposed scheme with the other state-of-the-art approaches is based on various criteria and has been tabulated in the Table. 5.4. A few of the criteria are described as follows:

1. Cryptosystem-based: It refers to homomorphic encryption techniques like NTRU, paillier, and RSA. The proposed methodology is based on paillier homomorphic encryption.
2. We have performed our proposed framework biometric system and other state-of-the-art on 8GB DDR4 RAM, 512GB SSD, Processor: AMD Ryzen 5, and OS: Windows 11 configuration to compare accuracy and storage.

3. Total computation time required for experiment: The total time required for the biometric system to encrypt iris code, compute hamming distance, and compare it with threshold is less than other state-of-the-art because other approaches use NTRU homomorphic encryption, which is lattice-based. In contrast, paillier homomorphic encryption uses random key generation. The time complexity of the proposed framework is $O(n)$.
4. Security analysis: We have compared our methodology with different attacks like man-in-the-middle, poison attack, frequency analysis, SQL injection attacks, and insider attacks. The proposed framework holds the client's information without compromising its security.

CHAPTER 6

Security Analysis

In this chapter, we have analyzed the security of the proposed frameworks.

1. **Lemma 5.1.:** The proposed frameworks are secure against poison attacks.

Proof. A duplicate fake attack and an erasure attack are combined to create a poison attack. The adversary erases the original data on the server in an erasure attack. The actual information on the server is replaced with false data in a duplicate faking attack. Data integrity is violated in both scenarios because the legitimate user loses access to the actual data. In P2B-AI, if an adversary tries to add or erase our original data can't do so because we have honest and trusted entity staff and Biometric experts; through this data, integrity is preserved.

2. **Lemma 5.2.:** The proposed frameworks are secure against man-in-the-middle attacks.

Proof. An attack where a third party acts as a man-in-the-middle between two parties and secretly passes messages between them that they believe are being sent directly to each other. In P2B-AI, client scanned iris image is transferred by staff to a biometric expert for encryption of iris code where staff and biometric expert are an honest entity, and then encrypted iris code is transferred to CSP, so there is no chance of third party involvement. Therefore, P2B-AI is secure against man-in-the-middle attacks.

3. **Lemma 5.3.:** The proposed frameworks are secure against SQL injection attacks.

Proof. Attackers can access a web application database without authorization by inserting a string of malicious code into a database query, a technique known as SQL injection. In the P2B-AI database, details are shared with honest entity staff and Biometric experts so an unauthorized person can't add a malicious string to our database. Therefore, P2B-AI is secure against SQL injection.

4. **Lemma 5.4.:** The proposed frameworks are secure against frequency analysis.

Proof. To defend against the frequency analysis attack, we implement the probabilistic Paillier homomorphic encryption method. We obtain various ciphertexts even for the same plain text. There is no direct mapping between the pairs of plain text and ciphertext. Therefore, P2B-AI is secure against frequency analysis.

5. **Lemma 5.5.:** The proposed frameworks are secure against insider threat attacks.

Proof. Insider threat attacks are malicious threats to the system from people within the system or the organization. In P2B-AI, we have two trusted entity staff and a biometric expert where insider attacks can't work. Therefore, P2B-AI is secure against insider threat attacks.

CHAPTER 7

Conclusion

Biometric authentication systems based on iris data are becoming more prevalent in various fields. However, these systems are not entirely foolproof, as they are susceptible to morphing, spoofing, and replay attacks. We propose a privacy-preserving biometric authentication system based on iris data resistant to these attacks. Our system utilizes an end-to-end encrypted solution based on exploiting the homomorphic properties of Paillier encryption. This ensures that the user's privacy is maintained during the entire authentication process, even while availing of third-party CSP services. Furthermore, the overall time complexity of the authentication process is minimized to $O(n)$. In contrast, a recent study discovered that morphing iris images could pose significant security risks to biometric authentication systems. The researchers successfully produced iris scans that embodied two different identities, indicating a high success rate for morph attacks on two iris recognition algorithms, DHVMR and SMR, when assessed on IITD datasets. These results highlight the need for enhanced security measures in biometric authentication systems. To address these security concerns, we propose extending our work to provide more security against man-in-the-middle attacks, insider attacks, poison attacks, and frequency analysis attacks. By enhancing our system's security, we can ensure that biometric authentication systems based on iris data are reliable and secure. As a result, this can have significant implications for various fields, including finance, healthcare, and government services. Our privacy-preserving biometric authentication system based on iris data provides an efficient and secure authentication process while maintaining user privacy. Moreover, by addressing security concerns such as morph attacks, man-in-the-middle attacks, insider attacks, poison attacks, and frequency analysis attacks, we can further enhance the security of biometric authentication systems based on iris data.

Future work aims to provide a comprehensive and robust privacy-preserving biometric authentication system based on iris data. This system will minimize the

risks associated with morphing, spoofing, and replay attacks and address a wider range of security concerns. The implications of this work are significant, as it can foster the adoption of biometric authentication in critical sectors such as finance, healthcare, and government services, where the reliability and security of user authentication are important.

References

- [1] M. A. A. Abdul Basit, Muhammad Younus Javed. Efficient iris recognition method for human identification. 2005.
- [2] M. C. Constantinos Patsakis, Jeroen van Rest and M. Bouroche. Privacy-preserving biometric authentication and matching via lattice-based encryption. pages 169–182, 2015.
- [3] P. D. E. Devi. Iris-based privacy-preserving biometric authentication using ntru homomorphic encryption. 2020.
- [4] M. GomezBarrero, C. Rathgeb, U. Scherhag, and C. Busch. Predicting the vulnerability of biometric systems to attacks based on morphed biometric information. *IET Biometrics*, 7(4):333–341, mar 12 2018.
- [5] S. H. and S. Malisuwan. A study of image enhancement for iris. *Journal of Industrial and Intelligent Information*, vol. 3, 70(2):4169–4184, 2018.
- [6] Z. S. Haiqing Li. Iris recognition on mobile devices using near-infrared images. 11(4):16–24, 2017.
- [7] M. S. Hiroyuki Suzuki. Secure biometric image sensor and authentication scheme based on compressed sensing. pages 5453–5458. IEEE, 2013.
- [8] R. Kaur, I. Chana, and J. Bhattacharya. Data deduplication techniques for efficient cloud storage management: a systematic review. *The Journal of Supercomputing*, 74(5):2035–2085, 2018.
- [9] J. Z. Muhammad Khurram Khan. Improving iris recognition performance using segmentation. *Quality Enhancement, Match Score Fusion, and Indexing - IEEE Journals and Magazine*, 2(4):1396–1400, 2006.
- [10] S. Nithyanandam. *A new Iris normalization process for recognition system with cryptographic Techniques*, 2011.

- [11] U. Scherhag, C. Rathgeb, J. Merkle, R. Breithaupt, and C. Busch. Face Recognition Systems Under Morphing Attacks: A Survey. *IEEE Access*, 7:23012–23026, 2019.
- [12] K. Shankar and P. Eswaran. Rgb based multiple share creation in visual cryptography with aid of elliptic curve cryptography. *China Communications* 14.2, 80(2):118–130, 2017.
- [13] R. Sharma and A. Ross. Image-Level Iris Morph Attack. *2021 IEEE International Conference on Image Processing (ICIP)*, sep 19 2021.
- [14] X. Y. Song Zhao. A secure and efficient fingerprint images encryption scheme. pages 2803–2808. IEEE, 2008.
- [15] J. Stanek, A. Sorniotti, E. Androulaki, and L. Kencl. A secure data deduplication scheme for cloud storage. In *International conference on financial cryptography and data security*, pages 99–118. Springer, 2014.
- [16] L. H. Trung. <https://euroasia-science.ru/pdf-arxiv/the-controllability-function-of-polynomial-for-descriptor-systems-23-31/>. *EurasianUnionScientists*, 4(65), 2019.
- [17] A. Ullah, K. Hamza, M. Azeem, and F. Farha. Secure healthcare data aggregation and deduplication scheme for fog-oriented iot. In *2019 IEEE International Conference on Smart Internet of Things (SmartIoT)*, pages 314–319. IEEE, 2019.
- [18] H. R. S. Zhou Wang, A. C. Bovik and E. P. Simoncelli. *Image quality assessment: from error visibility to structural similarity*. 2004.

