



Developing a Biometric Based Authentication to Secure Transferring Data via Cloud Environment

Thesis for the Degree of Doctor of Philosophy (PhD)

Kaabi Jasim Hussein Lafta

Supervisor: Dr. Tornai Róbert

UNIVERSITY OF DEBRECEN
Doctoral Council of Natural Sciences and Engineering
Doctoral School of Informatics

Debrecen, 2025

Hereby I declare that I prepared this thesis within the Doctoral Council of Natural Sciences and Engineering, Doctoral School of Informatics, University of Debrecen in order to obtain a PhD Degree in Engineering at Debrecen University.

The results published in the thesis are not reported in any other PhD theses.

Debrecen, 2025.04.04.

.....
signature of the candidate

Hereby I confirm that Kaabi Jasim Hussein Lafta candidate conducted his studies with my supervision within the Data Science and Visualization Doctoral Program of the Doctoral School of Informatics between 2018 and 2024. The independent studies and research work of the candidate significantly contributed to the results published in the thesis.

I also declare that the results published in the thesis are not reported in any other theses.

I support the acceptance of the thesis.

Debrecen, 2025.04.04.

.....
signature of the supervisor

Developing a Biometric Based Authentication to Secure Transferring Data via Cloud Environment

Dissertation submitted in partial fulfilment of the requirements
for the doctoral (PhD) degree in Informatics

Written by Kaabi Jasim Hussein Lafta certified Computer Science, MSc

Prepared in the framework of the
Informatics doctoral school of the University of Debrecen
(Data Science and Visualization programme)

Dissertation supervisor: Dr. Tornai Róbert

The official opponents of the dissertation:

Dr.
Dr.

The evaluation board:

chairperson: Dr.
members: Dr.
Dr.
Dr.
Dr.

The date and venue of the dissertation defence: Debrecen, ... 2025.

Contents

1	General Introduction	1
1.1	Introduction	1
1.2	Security of Cloud Computing	2
1.3	Encryption	4
1.4	Biometric	4
1.5	Related Works	5
1.6	Research Questions	10
1.7	Research Goals	11
1.8	Outline of the Dissertation	11
2	Cloud Computing and Bio-Cryptography	13
2.1	Introduction	13
2.2	Information Security	13
2.2.1	Biometrics	14
2.2.2	Authentication	16
2.2.3	Cryptography	16
2.3	Data Security Issues in the Cloud Environment	17
2.3.1	Key Management in Cloud Computing	18
2.3.2	Encrypt Data in Cloud Computing	19
2.4	Bio-Cryptography	19
2.5	Cloud Computing	20
2.5.1	Characteristics of Cloud Computing	21
2.5.2	Cloud Architecture	21
2.5.3	Services Model	22
2.5.4	Cloud Computing Types	22
2.5.5	The Layer Model of Cloud Computing	23
2.6	Iris	24
2.7	Neural Networks	25
2.8	Mathematical Background	26
2.8.1	Polynomial	26
2.8.2	Primality Testing Problem (PTP)	26
2.8.3	Modular Arithmetic	27
2.9	AES Algorithm Description	29
2.10	RSA Algorithm Description	31
2.10.1	RSA Public Key Generation	31
2.10.2	Main Features of RSA System	32
2.10.3	Problem Associated with RSA Algorithm:	33
2.11	Randomness Statistical Tests	33

2.11.1	Frequency Test (mono bit)	34
2.11.2	Frequency Test within a Block	34
2.11.3	Serial Test (Two-Bit Test)	35
2.11.4	The Run Test	35
2.11.5	Auto Correlation Test	36
2.11.6	Test for Longest Run of Ones in a Block	36
2.12	Mersenne Prime Number	37
2.13	B-spline	37
2.14	Zernike Moment	38
2.15	Convolution	40
2.16	Security Analysis	40
3	Design of the Proposed System	43
3.1	Introduction	43
3.2	The Proposed System	43
3.2.1	Read the Eye Images	44
3.2.2	Iris Image Preprocessing	45
3.2.3	Extract Features from the Iris	51
3.2.4	Authentication Stage	57
3.2.5	Encryption Stage	61
4	Performance and Evaluation of the Proposed System	67
4.1	Introduction	67
4.2	Description of Dataset	67
4.3	Implementation of the Proposed System	68
4.4	Evaluate the Result of the Proposed System	76
4.4.1	Case Study 1	76
4.4.2	Case Study 2	78
5	Theses	81
5.1	Authentication and Biometric Contributions	81
5.2	Cryptographic Contributions	82
5.2.1	Biometric-Based AES Key Generation	82
5.2.2	RSA Public Key Generation and Modified RSA (MRSA)	82
5.3	Performance Overview	82
6	Summary	83
	List of publications	85
	Bibliography	96

List of Figures

1.1	Cloud Computing	2
1.2	Cloud Computing Deployment Model & Service Model.	3
2.1	C.I.A. Triangle. Source: Singh et al. (2014)	14
2.2	Biometric Classifications. Source: Internet	15
2.3	Verification System	16
2.4	Identification System	16
2.5	Proposed Approaches To Ensure Data Security.	19
2.6	Cloud Computing Model. Source: Marinescu (2013)	20
2.7	Cloud Architecture. Source: Tanwer et al. (2010)	21
2.8	Cloud Services Model. Source: Ali et al. (2022)	22
2.9	Cloud Computing Types. Source: A et al. (2022)	23
2.10	Cloud Computing Architecture. Source: LI et al. (2013)	24
2.11	Block Diagram Of AES Algorithm	30
2.12	Block Diagram of RSA Algorithm	32
3.1	Block Diagram Of The Authentication Scheme	44
3.2	Block Diagram Of The Confidentiality Scheme	45
3.3	Iris Images, From CASIA Database. Source: CBSR Website.	45
3.4	The Outlines Show Iris And Pupil Localization Results.	46
3.5	False Iris Localization	47
3.6	Non-Ideal Iris Images Example.	47
3.7	Fixed Masking Used For Excluding Areas With Noise	48
3.8	Variable Shape Masking Used for Excluding Areas with Noise	48
3.9	The Proposed Partial Iris Pattern	49
3.10	Stack of Iris Images Of The Same Size	49
3.11	The Proposed Partial Iris Pattern $IR(2,1)$ Region	50
3.12	The Proposed Partial Iris Pattern $IR(2,3)$ Region	51
3.13	Flowchart To Get The Set Of Polynomials	53
3.14	Two Vectors Normalization Plan For Radial And Zernike Moments.	54
3.15	Flowchart To Compute The Radial Polynomials	55
3.16	Flowchart To Compute Zernike Basis Functions	56
3.17	CNN Layer-By-Layer Sampling Overview	59
3.18	One Sample Of ORL Dataset With Different Expressions.	60
3.19	Improvement Process Of The Neural Network Training	60
3.20	Flowchart To Compute Zernike Basis Functions	65
4.1	Preprocessing Of Iris: Case Study 1.	68

4.2 B-Spline Curve 71

4.3 Period When Implemented the AES Algorithm 72

4.4 Relation Between the Length Of Private Key And Time 74

4.5 Preprocessing Of Iris: Case Study 2. 74

List of Tables

1.1	Comparison: Traditional vs. Cloud Computing	4
1.2	Comparison: Proposed System With The Related Works	7
1.3	Summary of The Related Works	8
2.1	48 Mersenne Prime Numbers	27
2.2	Block Length M Based On Minimum Tested Sequence Length n . . .	36
2.3	The Frequencies v_i Of Longest Runs Of Ones In Each Block	37
3.1	List of The 8 First Orders Zernike Moments	53
3.2	Different Parameters In Our CNN	59
3.3	Initial Values For The Parameters	59
3.4	The Results Of Eye Recognition For The Images From Database . .	60
4.1	List Of The First 8 Orders Of Zernike Moments	69
4.2	Eight Features And Control Points	69
4.3	Congruence Control Points With Database	70
4.4	Select The Last Eight Digits To Obtain 128 Bits	71
4.5	128-Bit Hexadecimal Encrypted Key	72
4.6	Implementation Of AES Algorithm	72
4.7	Expand The Value Of Control Points	73
4.8	Eight Features And Control Points For Case Study 2	75
4.9	Congruence Of Control Points With Database	75
4.10	Select The Last Eight Digits To Obtain 128 Bits For Case Study 2 .	75
4.11	128-Bit Hexadecimal Encrypted Key	76
4.12	Summary Of Randomness Tests	80

List of Algorithms

1	Pre-Processing For Zernike Moments	52
2	Specifying The Order (p) And Repetition (q) Of Zernike	52
3	Compute The Radial Polynomials	54
4	Compute The Zernike Basis Functions	57
5	Compute The Zernike Moments	57
6	Biometric Authentication Process	61
7	Generate The AES Encryption Key	62
8	Bézier Curve	63
9	Determine The Largest Primes As p And q For RSA Algorithm . . .	64

Chapter 1

General Introduction

Cloud computing (CC) has emerged as the next phase framework in the evolution of Internet service. This promising technique is an improved and efficient approach to storing and/or processing information in the cloud to meet a set of applications and services. Therefore, cloud computing has revolutionized the computing industry since it has been adopted already by many large institutions. These institutions provide online resources, from online applications to data storage and server access. Consequently, individuals and organizations have benefited from it due to the flexibility of users' access to applications from anywhere without the headache of downloading and installing them. Moreover, reducing hardware and software is very cost-effective (Lourida et al., 2013).

1.1 Introduction

The potential of cloud computing could change many aspects of our daily life, from simple services such as email to more complicated ones such as e-governmental and health services (Greengard, 2010). The cloud typically consists of a service provider that stores the data and may perform other processing. These service providers should have security techniques to protect their client data, such as from disclosure of the data by unauthorized persons (N.Kumbhar et al., 2012).

Cloud computing can be defined as the ability to access applications on the internet by storing and protecting application data while providing a service. Cloud storage could include holding applications, business information, and personal data. In other words, cloud computing is a platform for easy and instant access to reconfigurable computational and communicational resources, such as servers, networks, storage, databases, applications, and other services. As a result, these resources can be swiftly provided and/or granted with minimal service provider interaction or administration overhead (Mell and Grance, 2009). Figure 1.1 illustrates cloud computing.

Cloud computing has three delivery models, namely, SaaS, IaaS, and PaaS, and four deployment models, namely, public cloud, private cloud, community cloud, and hybrid cloud (Khatoon and Ikram, 2014), as illustrated in Figure 1.2.

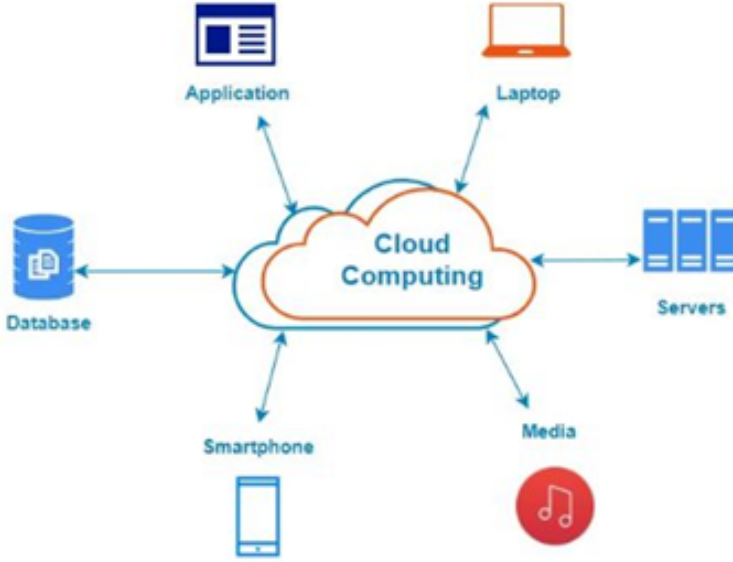


Figure 1.1: Cloud Computing

1.2 Security of Cloud Computing

Security could be a significant issue in cloud computing when handling sensitive data. Service providers must address this issue by maintaining data privacy and protection while allowing easy user on-demand access.

Therefore, security issues become an exceptionally challenging problem in cloud computing (Jouini et al., 2012). Security is the primary concern of user data in the cloud and can be classified into three categories: availability, traditional security, and third-party data control. However, current data controls are not sufficient to provide data storage security. Therefore, further research is required to improve and address these issues. Moreover, since cloud computing exploits variant types of service models (IaaS, SaaS, PaaS), security seems to be an even more challenging issue (Jouini et al., 2012). Therefore, some studies suggested that cloud security should be included and examined in each service model where different security levels exist for this purpose (Subashini and Kavitha, 2011).

The most dominating challenge in cloud computing is a security issue. To upload data and run software in other service providers' storage and servers, the service users need to trust and ensure the safety of these services. In addition, the pooled computing resources and the multi-tenancy model have introduced extra security challenges, necessitating original solutions to handle them. For instance, cloud computing usually offers far reliable framework services with relatively inexpensive cost for hackers (Chen et al., 2010).

To get into the cloud computing environment, it is necessary to care about the following things: privacy, security, reliability, performance, and portability. Security is the primary threat to cloud computing because of the multi-tenancy architecture. Therefore, the service providers also give the comfort of "trust us" to the users who are willing to move into the cloud environment (Subbiah et al., 2013).

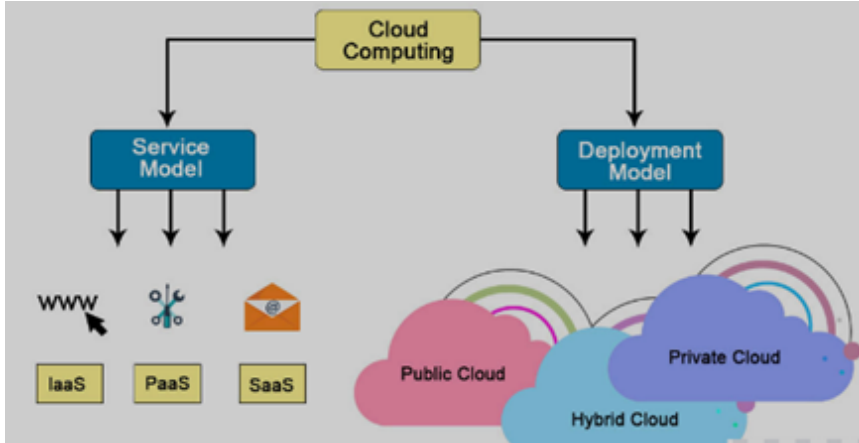


Figure 1.2: Cloud Computing Deployment Model & Service Model. Source: Mishra and Priyadarshini (2023)

Due to the unique nature of cloud computing and its services, some security vulnerabilities can rise as a real threat, even though they have not been considered an issue with other traditional computing. For instance, in typical system security cases where all resources are known, controls and policies could be quickly adopted since threats could be recognized and, therefore, can be efficiently mitigated.

Because traditional computational systems' components can be broken down to a specific perimeter in the organization, they are relatively easy to control and manage, and the required security is easy to determine. However, the computational power of typical systems is minimal, except for big organizations that can devote many resources to IT and its security measurements.

The service provider transparently manages most components (depending on the adopted cloud computing model) of the cloud environments, such as data storage devices, networking infrastructure, and servers. This approach seems to have advantages and disadvantages. Regarding the benefits of this approach, small organizations can spend less on security controls and, therefore, efficiently manage security. The disadvantage is the uncertainty of whether or not the cloud provider employs the proper security measurements because of the absence of visibility of security measurements. Even though the complex cloud has been developed for general use, every organization has a set of particular security requirements. However, it is very costly for these organizations to hire IT professionals with exceptional technical skills to implement and manage the cloud computing architecture that meets their security requirements.

Both traditional computing and cloud computing have pros and cons regarding security. However, security issues of cloud computing require further research and study to be fully explored, identified, and addressed since cloud computing has recently emerged and is still evolving. However, the merits of cloud computing justify most possible security issues that could appear (Lourida et al., 2013).

1.3 Encryption

The function of information security is to protect the integrity and privacy of the data and not allow unauthorized persons to access the system or data. To get a high level of security, we use the cryptosystem to encrypt the data and then send it to the network (Sattarova Feruza and Kim, 2007).

The encryption algorithm and encryption key are the base for any encrypted data, and the management of keys is the weak point for any encryption system. Therefore, we must find a suitable method to save these keys (Barker et al., 2012).

Cryptography is the art and science of developing a technique that enables information to be transmitted, stored, and/or processed securely so that this information is accessible by only authorized individual(s) (Shankar and Akshaya, 2014).

Cryptography plays a significant role in data protection and authenticity in applications running in a system connected to a network. It allows people to communicate or transfer data electronically without worries of deceit and deception (confidentially), in addition to ensuring the integrity of the message and the authenticity of the sender. There is a need for cryptographic algorithms because of the exponential increase in electronic transfer of data in several fields (Goshwe, 2013).

Table 1.1: Comparison: Traditional vs. Cloud Computing

Security Risk	Traditional Internet	Cloud Computing
Managed by policies to protect computing environment.	✓	✓
Locally managed.	✓	✗
Infrastructure on-site.	✓	✗
Encryption.	✓	Limited
Depend on the CIA security model.	✓	✓
Use the digital certificates (Authentication & non-repudiation).	✓	✓
An anti-virus and anti-malware are used.	✓	✓
Use the firewalls.	✓	✓
Auditing.	✓	✓
Data recovery cost.	High	Low
Disaster recovery.	✗	✓
The time of the service recovery.	Moderate to high	Low to moderate
Downtime risk.	Moderate	Low

1.4 Biometric

Biometric recognition is "automated recognition of persons by their behavioral and biological characteristics" (Bidgoli, 2006). Security and privacy of information are

significant challenges in the present world. As a result, essential techniques such as biometric recognition have been developed, which is considered a vital security technique, such as the authentication process. This is because biometric features such as fingerprints, iris, and faces are unique. As a result, these techniques are exploited with many other techniques to improve information security (Feng et al., 2008).

The biometric authentication process validates the unique persons depending on the physiological data such as fingerprint, iris and face, or behavioral qualities such as signature (Jain et al., 2004). To apply the authentication process, extract the features from biometrics and compare them with the accumulated biometric profile to allow authorized users access to the system or data.

Recently, a biometric authentication system has received increasing attention as a reliable replacement for password-based authentication. This is because it avoids the need to remember passwords. In addition, biometric features are complicated to be forged, guessed, or copied. Moreover, this authentication system presents a non-repudiation feature, where authenticated users' actions cannot be denied.

A cryptosystem based on biometrics represents a new promising approach since it integrates the merits of biometrics with cryptography, also known as crypto-biometric security systems (Chafia et al., 2010). The combination of biometrics and cryptography is used in key generation depending on biometrics. When the input biometric is congruent with the biometrics templates, the system will exploit to release the secret key. The biometric is immeasurably rounded to the keys in the state of biometrics key generation. Biometrics and cryptography have played an essential role in providing security for user information. To get a high level of security, combine biometrics and cryptography; this will produce a bio-cryptosystem that encrypts and decrypts the data (Saraswathi and Balasubramaniam, 2010).

Biometric security systems can identify individuals using specific (personal) biophysical characteristics of humans like voice, iris, ear, gait, and fingerprint (Zhang et al., 2011).

There are two ways to use a biometric key:

1. Biometric-based key generation for cryptography.
2. Biometric identification.

1.5 Related Works

Researchers in the security field's main goal is to protect important data when transmitting it in an insecure channel. This section shows the recent literature related to this study, specifically in terms of enhancing the security of cloud computing by using biometric features.

Ilankumaran and Deisy (2019) proposed a novel C2 code derived using orientation and magnitude information extracted from finger vein and iris images to improve the authenticating system. The C2 code eliminates the feature selection operator, reducing the process complexity by combining the orientation and magnitude information from finger vein and iris image inputs. Due to its reduced data handling complexity, this methodology can be implemented in a cloud computing environment-based biometric authentication system. This reduced data makes the cloud database more secured and authentication possible anytime, anywhere using the cloud environment (Ilankumaran and Deisy, 2019).

Yogesh Awasthi, Nikhi Kumar, and R. P. Agarwal explore the techniques/methods by which the user can authenticate a document using a combination of unique keys, which will be the user's iris and fingerprint. The iris and fingerprint codes will always be matched with the owner of the cloud account (Kumar et al., 2016).

In the work of Bineet Kaur, Sukhwinder Singh, and Jagdish Kumar, a quantum invariant feature set for rotation, noise, and illumination consisting of Zernike-Moment and polar harmonic shifts was used because it is difficult to extract the iris in an uncontrolled acquired environment (Kaur et al., 2018).

Balakumar P. and Venkatesan R. proposed a technique to secure the system using fingerprint and iris biometrics since they claim it is unlikely for an intruder to acquire both biometrics at the same time. A fusion approach joins these two biometrics-extracted features. After that, these extracted fused features created a relatively strong cryptography key. As a result, the suggested security technique is superior to state-of-the-art existing methods because it depends on multi biometrics to implement the security (Balakumar and Venkatesan, 2011).

Mingwu Zhang, Bo Yang, Wenzheng Zhang, and Tsuyoshi Takagi propose a novel multiple biometric-based encryption and authentication scheme that provides confidentiality, undeniability, unforgeability, and verifiability. It employs various biometrics to encrypt the message. With the help of public value produced by fuzzy extractor, it can reproduce the secure key from distinction biometric to decrypt the ciphertext (Zhang et al., 2011).

Parsi Kalpana and Sudha Singaraju stated that the security of the cloud relies on trusted computing and cryptography. They use the RSA encryption algorithm to protect the data so that only the authorized user(s) can access the data. Since users encrypt the data and then store it in the cloud storage, when a citrine user issues a request from the cloud provider to access the data, the cloud provider authenticates the user and grants his request by enabling him to access the data. The data is secured by implementing the RSA algorithm. Therefore, only the authorized user can access this work's data (Kalpana and Singaraju, 2012).

Jagadeesan, Thillaikkarasi, and Duraiswamy incorporate the extracted iris and fingerprint features to produce a 32-byte cryptographic key. This approach improves security by combining the complexity of factoring in large numbers. Initially, texture properties and minutiae points are extracted from the scanned images of the iris and fingerprint, respectively. After that, these features are fused at the feature level to generate multi-biometric templates, which are used to create a 256-bit secure cryptographic key. The experimental researcher's evaluations have illustrated the proposed approach security efficiency by producing unique and strong cryptographic key for every user (Jagadeesan et al., 2010).

Veerraju Gampala et al. implemented digital signature and elliptic curve data encryption for cloud computing security. This study suggests that elliptic curve cryptography can be used to achieve secure data transmission between cloud computing environments. The researchers presented a model consisting of: generate of key, generate of signature, encrypt and decrypt algorithm and signature verification (Gampala et al., 2012).

Amandeep Kaur and Sarpreet Singh presented a novel security method based on RSA and Twofish algorithms. The proposed approach generates a secure key using both these techniques, which is sent to the user by mail. Moreover, to improve security, a signature is being used. Therefore, this unique signature is needed when

users store their data. This obviously should improve cloud security (Kaur and Singh, 2013).

Shilpa M Pund and Chitra G Desai implemented the RSA algorithm using Mersenne primes. RSA's real challenge is selecting the public key and generating a private key when it depends on prime numbers. In this paper, the public key is generated randomly. Mersenne primes have been used to create large primes to enhance security. This algorithm can generate a private key in the 15 to 5000 digits (Pund and Desai, 2014).

Tirthani et al. have proposed a secure cloud architecture design. This architecture ensures the secure data transaction between clients and servers. This has been achieved using Elliptic-curve (EC) cryptography to encrypt data and the Diffie-Hellman key exchange (DHKE) technique for key distribution and channel establishment. The proposed encryption algorithm uses elliptical and linear cryptography techniques. In addition, the proposed approach has three main security components: key generation, authentication, and data encryption (Tirthani and R, 2014).

Rameshwari Malik and Pramod Kumar presented a model to protect the data using the AES-128 bit algorithm to encrypt the data. They also used the Diffie-Hellman algorithm to authenticate it before sending it to the cloud, producing data confidentiality and security (Malik and Kumar, 2015).

Table 1.2 compares our proposed systems with the related works.

Table 1.2: Comparison: Proposed System With The Related Works

Extract Features from Iris using Zernike Moments or CNN		
Reorder These Features to Produce Control Points of B-spline and Bézier		
Generate Encryption Key		Authentication
AES	RSA	
Using the control points of the B-spline to obtain 128 bits. This approach increases the randomness of the bit sequence for AES encryption keys. This key is then used to encrypt data before sending it to cloud computing.	Using the Bézier curve points to obtain the first prime number p , then applying Mersenne prime numbers to get the second prime number M_q . This approach produces a substantial prime number and generates a highly secure private key.	- Reordering eight features into the control points of B-spline and Bézier curves produces highly congruent iris accuracy. - CNN is applied for individual eye recognition, which is used in security systems for person identification. The method has been tested on different images and provides the highest accuracy for image recognition used in security systems.

Table 1.3: Summary of The Related Works

Authors	Methodology	Briefly Work Results
Kaur et al. (2018)	Extracting the features from fingerprint by Zernike moments to produce high congruent accuracy of iris proves to be effective for near-infrared and visible images collected from different iris sensors. Each of these is extracted from localized iris regions till the 15th.	The Zernike moments can match the iris images with high accuracy.
Qader et al. (2007)	Extracting the features from fingerprint by Zernike moments to produce high congruent accuracy of fingerprint.	The Zernike moments can match the fingerprint images.
Jagadeesan et al. (2010)	Extracted the texture properties and Minutiae points from the fingerprint and iris images and used them to generate a 256-bit cryptographic key.	The generated 256-bit cryptographic key can provide better user authentication and better security.
Balakumar and Venkatesan (2011)	Generate an encryption key from the features of fingerprint and iris, which are used as an authentication key for the person in the system.	The proposed techniques result in better security than the existing techniques.
Yang et al. (2011)	Using a set of invariant moment (Zernike & Geometric Moment) to extract the features using a powerful classification tool. This new approach can input various conditions that are used in cloud computing.	The proposed method has a higher matching accuracy than traditional or individual feature-based methods on public databases.
Zhang et al. (2011)	Using a multiple biometric-based encryption and authentication. It benefits from multiple biometrics to encrypt the data and generate the public key produced by a fuzzy extractor. It also decrypts the ciphertext by reproducing the secure key from distinct biometrics.	Can reproduce the secure key from distinct biometrics to decrypt the ciphertext. It provides security analysis, including semantic security and forgery resistance in the random oracle model.
Continued on next page		

Table 1.3 – continued from previous page

Authors	Methodology	Briefly Work Results
Li et al. (2012)	Using weighted Zernike moments on multi-discs as the features of fingerprint images. This method does not require rotational alignment and has robust performance since multiple centralized discs (multi-discs) are employed.	The proposed method has higher verification accuracy than most state-of-the-art texture-based methods.
Salman (2012)	Using fuzzy filter and minutiae extractions to generate the key using locations in the fingerprint image.	The key is generated dynamically using fingerprints and based on a fast and secure cubic spline.
Kalpana and Singaraju (2012)	Using RSA algorithm in cloud computing to enhance data security.	Only the authorized user can access the data because security is ensured by implementing the RSA algorithm.
Ilankumaran and Deisy (2019)	Extracting features from Minutiae to generate a (64-bit) encryption and decryption key from fingerprints, used in the DES algorithm.	Generates a (64-bit) key from human fingerprints for DES encryption.
Gampala et al. (2012)	Exploring data security in cloud computing by implementing digital signatures and encrypting them using elliptic curve techniques to enhance security.	Provides confidentiality and authentication of data between clouds using Elliptic Curve Cryptography.
Kaur and Singh (2013)	Using RSA & Twofish algorithm, this new scheme generates a key sent to the user via email.	Two techniques help generate a secure key, adding additional security features. The signature increases security in the cloud.
Chandra et al. (2013)	Using Minutiae points to generate RSA keys and encrypt and decrypt speech after converting it to text.	Using a unique set of keys enhances security and protects data from unauthorized access.
Pund and Desai (2014)	Using Mersenne prime numbers to obtain large p and q values for encryption with the RSA algorithm.	Small primes generate large Mersenne primes in the RSA algorithm (15–5000 digits).
Continued on next page		

Table 1.3 – continued from previous page

Authors	Methodology	Briefly Work Results
Tirthani and R (2014)	Using Diffie-Hellman Key Exchange (DHKE) for key distribution and channel establishment, combined with Elliptic Curve (EC) cryptography for data encryption in cloud architecture.	Three security checkpoints: authentication, key generation, and data encryption.
Malik and Kumar (2015)	Protecting data using the AES algorithm and authenticating via the Diffie-Hellman algorithm before launching in the cloud.	Ensures data confidentiality and security in cloud computing.

1.6 Research Questions

The ultimate goal of this dissertation is to find a secure method to enhance the security in cloud computing. Therefore, the research questions that the proposed research has answered are:

- I What are the necessary measurements and precautions that can be taken to improve privacy?
- II How can the keys be made unique to ensure they are associated with one and only one person?
- III How can the keys be made highly secure, difficult to steal, and periodically updated?
- IV The big question: "How can we enhance security in cloud computing using biometrics, B-splines, and Bézier curves?"

I did not use *composite Bézier curves* or *Bézier splines* in my thesis work. The difference between a B-spline (consisting of interconnected Bézier curves) and a Bézier curve is that the latter is globally modified while the first is locally adjustable by manipulating the control points. This helps create more complex curves that leads to more unique random numbers.

To get the ultimate answer, there are some sub-questions to be addressed:

- 1. What types of biometrics are suitable for use in cloud computing?
- 2. What algorithms are best suited for extracting features from biometric data?
- 3. Which encryption algorithms are most effective for enhancing security in cloud computing?
- 4. What method will generate a key from biometric data?

1.7 Research Goals

The main objective of this study is to enhance the security of cloud computing by.

1. Using biometrics to generate a symmetric key for the AES algorithm by using the Zernike moment to extract the features from biometrics. The AES algorithm is one of the algorithms used in cloud computing.
2. Using biometric and Bézier curves to generate a large prime number through applying Mersenne prime numbers to use in the RSA algorithm, which is used in cloud computing.
3. Using control points of B-spline to authenticate the user in cloud environment through extracting the eight features from biometric.

Bézier curves are not directly used to obtain large prime numbers, but there is some mathematical relationships worth exploring. Bézier curves are primarily applied in computer graphics for smooth curve modeling. Their points define the shape of a curve using polynomial functions, particularly Bernstein polynomials.

1.8 Outline of the Dissertation

Following this introductory chapter, the remainder of this study is organized as follows.

- Chapter 2 lists the theoretical background to bring about the concepts used in this study.
- Chapter 3 presents the details of the proposed system, which comprises the architecture and all implementation algorithms.
- Chapter 4 shows the results and discussion of the proposed system.
- Chapter 5 lists the theses points and gives a performance overview.
- Chapter 6 draws conclusions and suggests future works.

Chapter 2

Cloud Computing and Bio-Cryptography

This chapter is divided into two parts. The first part describes the architectures, types, services, and security of cloud computing. The second part describes bio-cryptography principles, which are divided into Biometrics and cryptography.

2.1 Introduction

The iris is selected as a biometric and describes the characteristics and types of irises. Also, it tells the Zernike moments that will be used as a method to extract the features from the iris to make authentication, then use them as a control point to draw the curve of the B-spline, then generate an encrypted key to encrypt the data by using AES-128. Also, it describes the point of the Bézier curve to get the first prime number p , and the second prime number will be selected from the Mersenne prime number M_q , which will be used in the RSA algorithm.

2.2 Information Security

Defined as the degree of protecting data against unauthorized access, use, disruption, or destruction. However, this protection, whether in storage or processing, is against denial of service to the authorized users. Information security involves detecting, documenting, and countering such risks (Singh et al., 2014). Therefore, information security is considered one of the most important subjects, especially in the computer and communication fields, because it transmits information through insecure channels. This subject has many challenges because of the increasing number of security breaches worldwide (Balakumar and Venkatesan, 2011).

In 2002, Don Parker proposed an alternative model to the classic CIA triad called the Six Atomic Elements of Information. These elements are as follows: Confidentiality, possession, integrity, authenticity, availability, and usefulness. Figure 2.1 illustrates the CIA triangle.

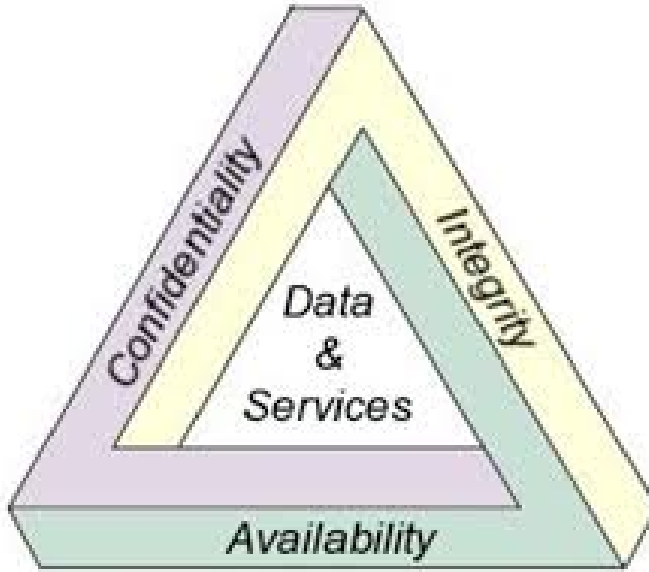


Figure 2.1: C.I.A. Triangle. Source: Singh et al. (2014)

Confidentiality: In general, Confidentiality is inactivating the information and resources for unauthorized persons, which means the Confidentiality makes the information seen only for the authorized people. Saving the secrecy of information from unauthorized access is an essential aspect of information security. Saving information secrecy arises from using computers in sensitive fields. Organizations need to protect their information and resources from malicious actions.

Integrity: Integrity represents how trustful the data and resources are. It usually means prohibiting unauthorized and inappropriate access to data. Integrity can be for data, i.e., the information content, or integrity of the source, i.e., data source. Integrity means that the information guaranteed remains without any change. That means it must be advertent for any possible changes from malicious actions or risks. The integrity of data consists of accuracy and reliability.

Availability: It is the ability to use the resources and access the information when the clients need it. The organization's information must be available to the applications and authorized users. The resources are not beneficial when they are not available. Sometimes, information must be constantly changed, which means the authorized user can access it. Availability and security may need to be considered together since someone may arrange to limit access to a service to improve security (Singh et al., 2014).

2.2.1 Biometrics

Biometrics provides automated techniques for verifying or identifying depending on features (physiological or behaviour), such as voice, iris, fingerprint, and gait. These features are measurable and unique.

The biometric systems are computation methods used to validate the persons based on the physical or behavioural characteristics. The "computation methods"

are illustrated in the biometrics systems executed by digital computers. The physical or behavioural characteristics refer to the biometrics that can recognize the person depending on these features. Therefore, the Biometrics answer the question "who you are", whereas the other techniques explore "what you have", such as card or phone (Wayman et al., 2005; Ross and Flynn, 2008).

Biometric Characteristics

Biological and behavioural characteristics must have the following aspects for an accepted biometric application.

- **Stability:** The features of Biometrics for the person must be constant with the time for the specific algorithm.
- **Effectiveness:** Recognition accuracy and the resources needed to achieve that the accuracy must accept the application's limitation.
- **Universality:** Everyone using the applications must have the feature.
- **Acceptability:** Application users must provide biometric system features.
- **Evasion:** The personal feature could be duplicated through biological features and simulation of behavior properties.
- **Individuality:** The service properties must have good offer for the users.
- **Measurable:** Must be converting the biometric features to the digital format to use in the system that do not cause the annoyance to the users. The data obtained must be manipulated to extract sets of features (Jain et al., 2006).

Many biometric features exist and are employed by various applications. Figure 2.2 illustrates the classifications of Biometrics. Each biometric method has its advantages and disadvantages, making it suitable for certain applications but not for others. In fact, there is no perfect biometric that can meet the demands of all applications.

The suitability of Biometrics for the performance of an application depends on the mode of application and the benefit of the biometric feature (Jain et al., 2004).

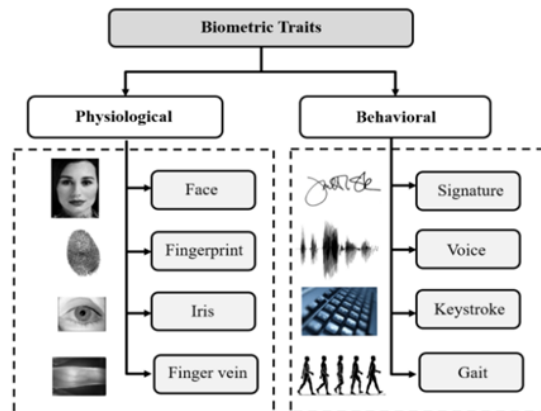


Figure 2.2: Biometric Classifications. Source: Internet

2.2.2 Authentication

There are two types of authentication searches in the database. The first type is a verification process when the process can only match the iris image with a known image. It performs a comparison to authenticate the person, as shown in Figure 2.3.

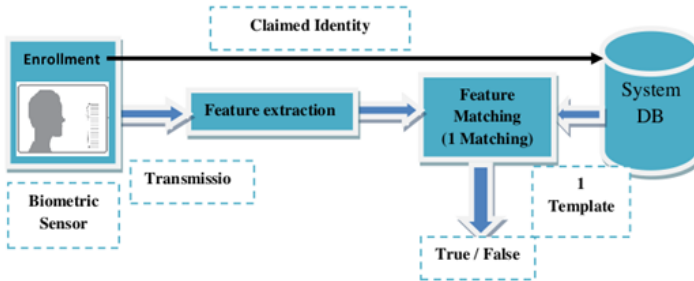


Figure 2.3: Verification System

The second type of search is the identification process in which the search is one-to-many, as shown in Figure 2.4. The system continues to search the entire template database for matches. The result may succeed if the matching is true or not if there is no record for the person in the database.

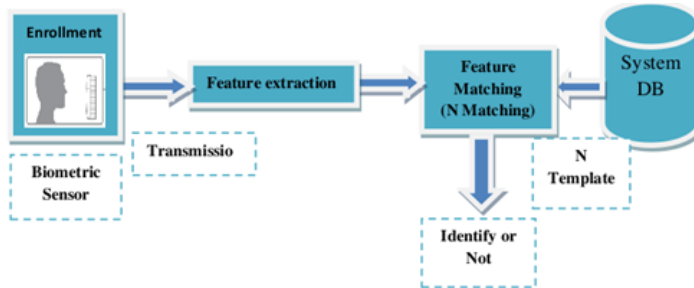


Figure 2.4: Identification System

Before the system identifies the user, he must register with his biometric data. These data are recorded, processed, and then stored. This process is usually called enrollment.

2.2.3 Cryptography

Cryptography is a crucial topic in the security of communication systems. Its importance is increasing rapidly as a basic cornerstone in computer security. Computers with communications systems are increasingly used, which has resulted in increased information risk. Although these threats may require a variety of countermeasures, encryption is a major way of ensuring secure data communication and storage.

The purposes of cryptography are:

- Integrity: Guarantees the contents of the message sent from the sender must remain unchanged when it gets to the intended recipient.
- Access Control: Specifies and controls who can access.
- Confidentiality: The contents of the message sent by the sender can only the intended recipient read it.
- Non-Repudiation: Not allowing the sender to say "the message is not sent from me".
- Authentication: Authentication techniques help in establishing a proof of identity. This ensures that the sender of the message is known.
- Availability: The data and resources are accessible by the authorized persons when they need (Mitali et al., 2014).

Cryptographic systems are categorized according to three main factors:

1. The methods used for ciphering plaintext. The encryption algorithms use two general operations: substitution and transposition. In substitution, plaintext elements (such as bits, letters, groups, etc.) are transformed into another element. Meanwhile, the transposition shuffles the elements of the plaintext. The standard requirement is when the information must not be messed up (all operations can be reversible). Most cryptographic systems consist of many stages of substitution and transposition.
2. Number of keys. When the sender and receiver use one key, the encryption is called "symmetric" and is sometimes called "single-key", "secret-key", or conventional. When different keys are used for the sender and receiver. The encryption is called asymmetric, two-key, or public-key.
3. The method of processing plaintext. A block cipher process is element-block by element-block, and the outputs are produced similarly. On the other hand, a stream cipher continuously processes elements, producing one element per transaction on the output (Stallings, 2006).

Examples of basic applications in cryptography are encryption and signature schemes. Encryption schemes guarantee the transferred information's Confidentiality (or privacy), while signature schemes ensure information reliability (Goldreich et al., 2005).

2.3 Data Security Issues in the Cloud Environment

Cloud computing security concerns the primary inhibitor for adopting cloud computing services.

1. **Data Availability:** CC data is usually kept in chunks and distributed across several servers located at various locations or clouds. In such a system, making data available becomes a major issue. This is because providing seamless provision is difficult.

2. **Data Integrity:** The providers of cloud services must apply some techniques to save the data integrity.
3. **Storage, Backup and Recovery:** When the data passes to the cloud, the provider must be guaranteed enough flexibility in storage systems for the data. Minimally, the provider must be able to provide the RAID (Redundant Array of Independent Disks) storage systems; also, the provider must save these data in multiple copies in several servers.
4. **Privacy and Confidentiality:** The client's data in the cloud must have assurance to access these data and control the authorized access. Another risk that can occur on the important client data through an inappropriate access by cloud personnel this state can be as a potential to cloud data. Guarantees must be provided by the cloud provider to the clients, and they should feel safe when storing their data or using the cloud services.
5. **Data Location:** In CC, it is difficult for the user to locate his data because it is mobile. As for the organization, which has very important data in the cloud, its location can be known.

Also, the provider must provide an important aspect of cloud computing, a backup service, which is very important to the businesses that the cloud applications depend on (Kalpana and Singaraju, 2012).

2.3.1 Key Management in Cloud Computing

Key management is an important and complex issue, and systems that handle data encryption are similarly complex. The organization must determine the accuracy of the organization's encryption security and manage the main life cycle. Access to these private encryption keys is only allowed to authorized users. It can be deleted or revoked when users no longer need access. That's the purpose of encryption keys: to encrypt important data in a secure way. Several options must be provided to manage the storage and transfer of encryption keys within a secure element. For example, encryption keys are stored on the same server or a different server from the data. The key management service must be compatible with current standards.

Ideally, for the greatest protection, key management should be completely separated from the cloud service provider that contains or hosts the data (Alliance, 2012), one design to consider is a remote key management service where the customer maintains the KMS or Enterprise Key Management (EKM) solution on-premises. Companies maintain and support their own KMS.

KMS frequently uses the HSM module to create and protect keys. It can include a cloud provider of encryption/decryption services if it already provides a full information management service (KMS) as a service for these solutions, A key management model through which the customer values. Full control over encryption/decryption keys. Thus, the cloud provider does not keep keys and cannot decrypt customer data, thus facilitating the process of storing encrypted data. This type of solution can be used by cloud storage providers, SaaS providers, and IaaS providers (Alliance, 2012).

2.3.2 Encrypt Data in Cloud Computing

Encryption and key management are used to protect data in the cloud, and it is one of the basic methods that private cloud computing systems must use to protect against breaches by unauthorized persons. The protection of resources is provided by encryption, through which key management allows access to protected resources. Cloud users encrypt their data as cipher-text, through which the key decrypts their data and then sends it as cipher-text to the cloud data storage vendor.

Data-at-rest encryption is one of the most common issues in IaaS environments. In PaaS environments, data encryption at rest is complex and requires tools specific to the provider's offerings. In SaaS environments, data encryption at rest is one of the features that users cannot implement but can be provided by service providers (Amazon Web Services, 2009).

It has also been confirmed that encryption is one of the most secure, confidential, and used ways to maintain the security of user data when sending it in the cloud. It was discussed as one of the solutions in the cloud environment, as it represents one of the influential tools that can be used efficiently. Cloud service providers generally support the encryption of data storage. Thus, encryption increases the trustworthiness of users to benefit from cloud services with confidence by knowing that their confidential data is fully protected by encryption (Dillon et al., 2010).

Encrypting the data on the backup media protects it from lost, stolen, or lost media, which the cloud service provider implements transparently. It is the responsibility of the customer and the data provider to verify that this encryption has occurred. As for the encryption infrastructure, it deals with the longevity of the existing data. Dynamic data encryption attempts to exploit attacks on cloud service providers and data in memory (Lei et al., 2010).

In Soofi et al. (2014), the review result was identified in Figure 2.5, which shows the proposed cryptography-based methods for data security in cloud computing.

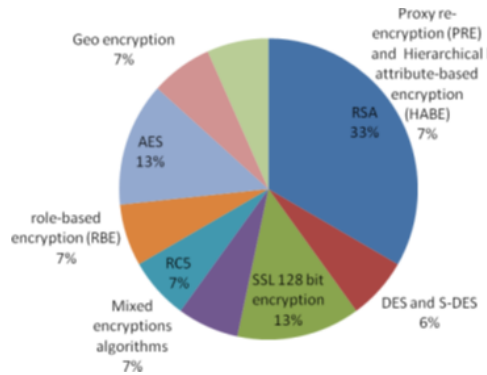


Figure 2.5: Proposed Approaches To Ensure Data Security Using Encryption Techniques. Source: Soofi et al. (2014)

2.4 Bio-Cryptography

Cryptography ensures the security of data transmitted over non-secure means of transmission. It is the key to authenticate the message, not personal information. Large keys are needed for encryption and decryption when sending and receiving

messages. These keys may be predicted and thus broken. Furthermore, keeping and exchanging random keys during encryption and decryption are problematic for cryptography systems. One solution for this problem is using Biometrics. Biometric cryptosystems use cryptography with Biometrics to get more secure. Security is achieved by cryptography, biometrics supply access, and the need to remember pass codes or carry pass cards, tokens, etc. is removed. Biometric cryptosystems work in such a way that the biometric template generates cryptographic keys for the end users and stores them in the database. The key cannot be exposed without biometric authentication (Sarkar and Chatterjee, 2014).

2.5 Cloud Computing

Defined as a model created to provide convenient, broad, and cheaper user access to shared resources on demand and with minimal effort (such as servers, networks, storage devices, and services) that can be easily managed with minimal effort (Mell and Grance, 2009; Raov et al., 2009).

Cloud computing is a paradigm by which the access of shared resources and common infrastructure is possible. It also provides services on demand over the network to comply with the changing business needs. The locations of the hardware resources are not known to the end user. Cloud computing provides facilities to develop, deploy, and manage the users' applications, which need the virtualization resources that manage and maintain themselves. Using browsers to access the applications of the CC and deploying them on computers via the internet. Figure 2.6 illustrate the CC model (Rimal et al., 2009).

In the CC, everything is a service. Examples are (Kaur, 2012):

- OaaS, SaaS, PaaS, Haas, DaaS, IaaS, BaaS and FaaS



Figure 2.6: Cloud Computing Model. Source: Marinescu (2013)

The security in the CC depends on protecting the data security and access control. The basis of access control is user authentication, which keeps unauthorized access and allows authorized persons in with a minimum of worry. Authentication and access control are very important in this environment; the cloud devices

and data are accessible to everyone over the internet (Rajasekar and Manigandan, 2015).

2.5.1 Characteristics of Cloud Computing

Many characteristics of CC that provide different traditional services are:

- Business model: CC provides a business model since there is usually a fee for using services and resources.
- Global network access: CC provides services anywhere using a standard terminal like smart phones, PCs, and PDAs.
- On-demand service: This is possible since CC is a large service pool that can provide a service or resource at any required place.
- Resource pooling independent of location: Suppliers of computing resources are grouped to allow the serving of many customers by what is known as using a “multitenant model”. This is possible because different physical and virtual resources are not fixed but assigned and reassigned dynamically following the requests.
- Easy use: Most cloud providers offer internet GUIs that simplify application program interfaces and allow clients to use CC services easily (Gampala et al., 2012).

2.5.2 Cloud Architecture

It is a group of distributed, interconnected, and virtual computers that can be dynamically acquired and provided as one or more computing resources. Cloud computing is also a type of parallel system.

Figure 2.7 generally describes the layered architecture of cloud computing, distinct layers from physical hardware down to end-user applications and data centers.

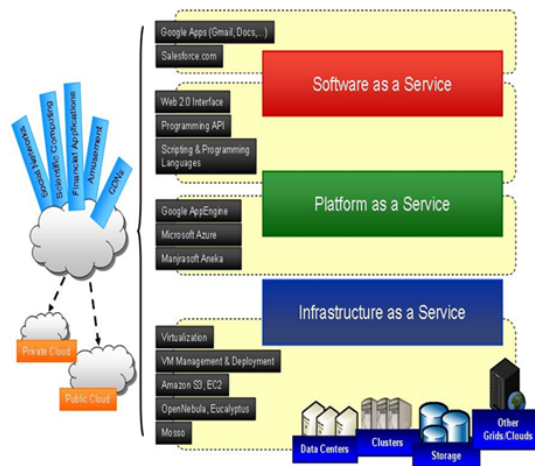


Figure 2.7: Cloud Architecture. Source: Tanwer et al. (2010)

Infrastructure is the availability of data centers because it hosts hundreds or thousands of devices and also helps and supports all cloud deployments on a very large scale. Home clouds are a more homogeneous scenario where we can utilize idle CPU cycles.

2.5.3 Services Model

Figure 2.8 demonstrates models for cloud services, and they consist of:

1. IaaS: Consumers use IT infrastructure directly. In this type, virtualization is used to integrate and/or analyze physical resources in a way that meets increasing or decreasing demands on resources. Typical infrastructure components included in this type are virtual machines. The main strategy for virtualization is to set up independent virtual machines (VMs) that are separate from both the machines underneath them and from other VMs (Dillon et al., 2010; Asha and Rao, 2013; Alnajdi et al., 2016; Ashalatha and Vaidehi, 2012).
2. SaaS: In this type of service model, consumers release their products into a hosting environment. Consumers can access their products through networks that include multiple clients. Consumers cannot control the cloud infrastructure, which often uses a multi-tenancy system architecture. On a SaaS cloud, applications are organized into a single logical environment that enables scaling and optimization in various aspects such as speed, availability, security, disaster recovery, and maintenance.
3. PaaS: It is a development platform that supports the software to support agreed-upon services and its entire life cycle. Consumers can build cloud applications (such as SaaS) directly on the PaaS cloud, operating systems, virtual memories, and applications.



Figure 2.8: Cloud Services Model. Source: Ali et al. (2022)

2.5.4 Cloud Computing Types

The various types of CC models depicted in Figure 2.9, which are public, private, hybrid and community cloud.

1. **Private Cloud:** The single organization exclusively operates it. The internal or third-party vendor can also manage it.
2. **Community Cloud:** In this type of cloud, the infrastructure can be shared, managed, and hosted by many organizations or through third-party providers.
3. **Public Cloud:** The resources (applications, storage devices) are publicly available by www. The services of this type of cloud may be on a pay basis or free. Therefore, this cloud model is a standard model of computing.
4. **Hybrid Cloud:** It is a mixture for more than one cloud that are connected together to provide different models, or we can say, it is many types of clouds which are grouped in a method to allow the applications/data to be moved seamlessly among deployment systems (Zhang et al., 2010).

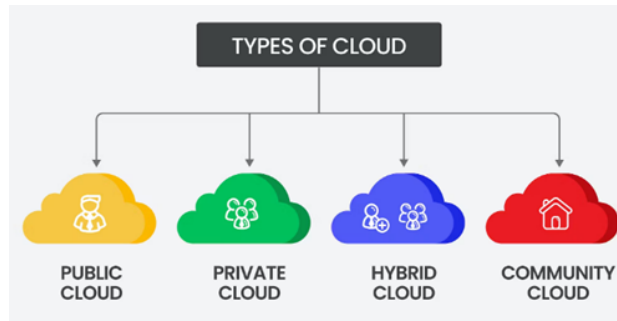


Figure 2.9: Cloud Computing Types. Source: A et al. (2022)

2.5.5 The Layer Model of Cloud Computing

There are four layers in the architecture of a CC environment: Hardware layer (HL), infrastructure layer (IL), platform layer (PL), and application layer (AL), as shown in Figure 2.10.

1. **Hardware Layer (HL):** In the cloud, this layer will be managing the physical resources. These resources include power, routers, cooling systems, and switches.
2. **Infrastructure or virtualization layer (IL):** Also called virtualization layer. It is a group of storage and other computing resources that virtually split the physical resources. The infrastructure layer is a major component in CC. This is mainly because many key features are available via virtualization technologies.
3. **Platform layer (PL):** It consists of application frameworks and OS. Its main function is to reduce the burden of deploying live applications in VM (virtual machine) containers as much as possible.
4. **Application layer (AL):** The application layer is the actual cloud computing applications. The main difference between traditional applications and

cloud applications is that the latter can leverage the automatic-scaling feature to improve performance, availability, and results in lower operating cost (Li and Xu, 2009).

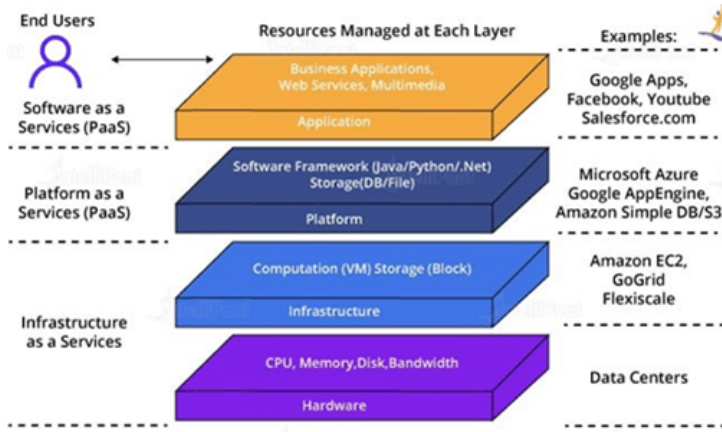


Figure 2.10: Cloud Computing Architecture. Source: LI et al. (2013)

2.6 Iris

Person identification and authentication are daily procedures, such as accessing gates to restricted areas, using ATMs, police and immigration processes, using equipment, and more. In traditional systems, these procedures rely on something a person knows (such as usernames and passwords) or something they carry (such as magnetic cards, keys, or chip cards). As technologies and services become available everywhere, these methods are no longer secure. The passwords can be attacked. Cards may be stolen. To achieve secure human identification as well as reduce the fraudulent claims of individuals, we should use something that actually discriminates each person. Biometrics technology offers an efficient way for human identification based on individuals' measurable physiological and/or behavioral characteristics. These characteristics may include fingerprints, the human face, hand geometry, iris, signature dynamics, voice, etc. It has been proven that these characteristics are unique for each individual and can be used efficiently for human identity verification (Bowyer et al., 2008).

Iris is one of the famous biometric techniques for identification and verification because it is unique for any person, and it is not changed through life, except for the distortion that may happen on the iris by injury and not natural physiological changes (Omelina et al., 2021).

Iris is one of the most well-known biometric technologies used as evidence in courts of law worldwide.

Recently, there have been many tremendous achievements in the field of iris biometric systems and their development on a large scale, such as in security issues, surveillance systems, banking services, education issues, etc.

These systems have developed significantly and decisively in the past two decades (Yang and Zhang, 2012).

The iris is inherently unique. Because it contains wide blood vessels with great chaos, the pattern is unique and stable over time, and a picture of the iris is taken directly from the person. It means the iris has no forgery or theft, etc. (Daugman, 1998; Tisse et al., 2002).

The iris has a fluctuating pattern, a unique ring shape, and a constant size, which is a major mathematical advantage.

Acting and not being sensitive to corner illumination. It also affects capturing the iris at different angles; convergent transitions only easily cause pupil dilation. Pattern distortion is non-reversible and better for personal identification systems (Ma et al., 2003; Aggarwal, 2018a).

2.7 Neural Networks

It is considered one of the most important parts of artificial intelligence. In an attempt to mimic the human nervous system, neurons are connected to areas called synapses, which have to be classified into 1000 classes. Since 2012, it has performed best in these challenges and outperforms human ability to classify images (Russakovsky et al., 2015). In statistics, it is used as a model for solving classification or regression problems, which is known as the state-of-the-art model. Technical techniques, especially image recognition and classification (O'Shea, 2015), contain three different layers: input layer, hidden layers, and output layer (Lecun et al., 1998).

Convolutional Neural Networks

The first successful application of CNN (Ma et al., 2003) stems from the 1990s, while the first application of deep CNN was in 2012 (He et al., 2016). It performed amazingly in many areas, especially in the ImageNet challenge.

And many modifications to the network architecture have led to further improvements in recent years (Simonyan, 2014; Szegedy et al., 2017, 2015; Zeiler and Fergus, 2014; Heaton, 2017).

Detailed descriptions and efficient training of the CNN can be found (University, 2024; Aggarwal, 2018b).

Researchers are developing algorithms and advancing techniques for image classification in machine learning and related problems. The key to building algorithms in CNNs is designing diverse architectures.

Use it to learn higher order features in data using convolution. It also works exceptionally well to recognize objects in images, for text analysis with character recognition, and in several applications, including self-driving cars, robotics, and drones (Simonyan, 2014). CNN have excellent recognition accuracy and the cost of computational complexity is very high. CNNs can extract the features that are most relevant to the task to be solved as well. It is considered one of the most advanced technologies in areas such as image recognition, object discovery, image segmentation, and other applications (Smart, 2016). Major improvements happened in image recognition tasks, especially between 2011 and 2015, as the top-5 error rate decreased from over 25% to around 4% (Russakovsky et al., 2015).

2.8 Mathematical Background

2.8.1 Polynomial

A polynomial function is a function such as a quadratic, a cubic, and so on, involving only non-negative integer powers of x . A polynomial of degree n is a function of the form

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_2 x^2 + a_1 x + a_0 \quad (2.1)$$

Where the a 's are real numbers (sometimes called the coefficients of the polynomial). Although this general formula might look complicated, particular examples are much simpler. For example,

$$f(x) = 4x^3 - 3x^2 + 2 \quad (2.2)$$

Polynomial of degree 3, as 3 is the formula's highest power of x . This is called a polynomial

$$f(x) = x^7 - 4x^5 + 1 \quad (2.3)$$

Polynomial of degree 7, as 7 is the highest power of x . Notice here that we don't need every power of x up to 7; we need to know only the highest power of x to determine the degree. An example of a kind you may be familiar with is

$$f(x) = 4x^2 - 2x - 4 \quad (2.4)$$

Polynomial of degree 2, as 2 is the highest power of x . This is called a quadratic. Functions containing other operations, such as square roots, are not polynomials. For example,

$$f(x) = 4x^3 + \sqrt{x} - 1 \quad (2.5)$$

is not a polynomial as it contains a square root. And

$$f(x) = 5x^4 - 2x^2 + \frac{3}{x} \quad (2.6)$$

is not a polynomial as it contains a 'divide by x '. The degree of a polynomial is the highest power of x in its expression. Constant (non-zero) polynomials, linear polynomials, quadratics, cubics, and quarters are polynomials of degree 0, 1, 2, 3, and 4, respectively. The function $f(x) = 0$ is also a polynomial, but we say that its degree is 'undefined' (Yan, 2012).

2.8.2 Primality Testing Problem (PTP)

PTP can be formally defined as follows:

$$PTP^{DEF} = \begin{cases} \text{Input: } n > 1 \\ \text{Output: } \begin{cases} \text{Yes: } n \in \text{Primes} \\ \text{No: Otherwise} \end{cases} \end{cases} \quad (2.7)$$

Theoretically, PTP can be solved in polynomial time; that is, PTP can be solved efficiently on a computer. However, it may still be difficult to determine whether a number is a large prime. Call a number a Mersenne prime if it is of the form $M_p = 2^p - 1$, where p is prime and $2^p - 1$ is also prime.

Table 2.1 illustrates that only 48 Mersenne prime numbers have been found. Note that $257885161 - 1$ is not only the largest known Mersenne prime but also the largest known prime in the world to date. The search for the largest Mersenne prime and/or the largest prime has always been a hot topic in computational number theory. EFF (Electronic Frontier Foundation) has offered in total 550,000 US dollars to the first individual or organization who can find the following large primes (Lewinter and Meyer, 2015).

2.8.3 Modular Arithmetic

Modular arithmetic sheds light on the relation of integers to their remainders when they are divided by a given positive integer.

Modular Equation

$ax \equiv b \pmod{n}$, where a , b , and n are given, is a modular equation. An example such as $3x \equiv 7 \pmod{11}$ should make you recall the ordinary equation $3x = 7$. We are looking for an integer solution, as $\frac{7}{3}$ is unacceptable in modular arithmetic. You might guess that $x = 6$ is a solution, since $36 = 18 \equiv 7 \pmod{11}$. We shall develop a method for solving modular equations only after determining when they are solvable. We shall also see that there may be several distinct solutions, such

Table 2.1: 48 Mersenne Prime Numbers

No	p	Digits(M_p)	Time	No	p	Digits(M_p)	Time
1	2	1	-	2	3	1	-
3	5	2	-	4	7	3	-
5	13	4	1461	6	17	6	1588
7	19	6	1588	8	31	10	1750
9	61	19	1883	10	89	27	1911
11	107	33	1913	12	127	39	1876
13	521	157	1952	14	607	183	1952
15	1279	386	1952	16	2203	664	1952
17	2281	687	1952	18	3217	969	1957
19	4253	1281	1961	20	4423	1332	1961
21	9689	2917	1963	22	9941	2993	1963
23	11213	3376	1963	24	19937	6002	1971
25	21701	6533	1978	26	23209	6987	1979
27	44497	13395	1979	28	86243	25962	1982
29	110503	33265	1988	30	132049	39751	1983
31	216091	65050	1985	32	756839	227832	1992
33	859433	258716	1994	34	1257787	378632	1996
35	1398269	420921	1996	36	2976221	895932	1997
37	3021377	909526	1998	38	6972593	2098960	1999
39	13466917	4053946	2001	40	2.1E+07	6320430	2003
41	24036583	7235733	2004	42	2.6E+07	7816230	2005
43	30402457	9152052	2005	44	3.3E+07	9808358	2006
45	37156667	1.1E+07	2008	46	4.3E+07	1.3E+07	2009
47	43112609	1.3E+07	2008	48	5.8E+07	1.7E+07	2014

as unequal mod n . Consider $2x \equiv 8 \pmod{16}$. This modular equation has two distinct, or different, solutions $x = 4$ and $x = 12$ since $4 \not\equiv 12 \pmod{16}$. We wouldn't consider $x = 20$ an additional solution since $20 \equiv 4 \pmod{16}$; that is, it isn't distinct from the solution $x = 4$. The same applies to $x = 28, 36, 44, \dots$

Here is a theorem on linear modular equations: modular equations of first degree (as opposed to $x^2 \equiv 1 \pmod{7}$, which is quadratic).

Theorem 1. *The congruence equation $ax \equiv b \pmod{n}$ is solvable if and only if $d \mid b$, where $d = \gcd(a, n)$.*

Proof. The equation $ax \equiv b \pmod{n}$ is equivalent to the statement $n \mid (ax - b)$, which can be rewritten as the Diophantine equation:

$$ax - b = kn$$

for some integer k , or equivalently:

$$ax - kn = b.$$

As seen in the previous chapter, this Diophantine equation has a solution if and only if $d \mid b$, where $d = \gcd(a, n)$, completing the proof. Furthermore, there are exactly d solutions modulo n .

Notice that if $\gcd(a, n) = 1$, then the congruence $ax \equiv b \pmod{n}$ is solvable for any b , and the solution is unique modulo n .

Laws of Modular Arithmetic

The following five statements are different ways of expressing the same concept:

1. $a \equiv b \pmod{n}$.
2. $a - b$ is a multiple of n , that is, $n \mid (a - b)$, or n divides $a - b$.
3. a and b belong to the same column in a modular chart for n , meaning they are in the same congruence class modulo n .
4. a and b have the same remainder when divided by n .
5. a and b have the same form, i.e., one is $kn + r$ and the other is $jn + r$, where r satisfies $0 \leq r \leq n - 1$.

Congruence equations exhibit many properties similar to ordinary equations. Here are some fundamental properties:

1. If $a \equiv b \pmod{n}$, then $b \equiv a \pmod{n}$. (Symmetry)
2. If $a \equiv b \pmod{n}$ and $b \equiv c \pmod{n}$, then $a \equiv c \pmod{n}$. (Transitivity)
3. If $a \equiv b \pmod{n}$, then $a \pm c \equiv b \pm c \pmod{n}$. (Addition and Subtraction)
4. If $a \equiv b \pmod{n}$ and $c \equiv d \pmod{n}$, then $a \pm c \equiv b \pm d \pmod{n}$. (Pairwise Addition and Subtraction)
5. If $a \equiv b \pmod{n}$, then $ac \equiv bc \pmod{n}$. (Multiplication)

6. If $a \equiv b \pmod{n}$ and $c \equiv d \pmod{n}$, then $ac \equiv bd \pmod{n}$. (Pairwise Multiplication)
7. If $a \equiv b \pmod{n}$, then $ac \equiv bc \pmod{n}$ for any integer c . (Scaling)

Note that $ac \equiv bc \pmod{n}$ does not permit us to conclude that $a \equiv b \pmod{n}$. As an illustration, canceling a 6 in the congruence equation $12 \equiv 18 \pmod{6}$ yields the false equation $2 \equiv 3 \pmod{6}$ when the factor and the modulus are relatively prime. Consider the equation $20 \equiv 50 \pmod{6}$. Upon canceling the factor 5, we obtain the correct equation $4 \equiv 10 \pmod{6}$. Of course, 5 and 6 are relatively prime, so the cancellation was assured to be legitimate.

Another situation that permits cancelation is $ac \equiv bc \pmod{nc}$. This means that $nc|ac - bc$, or $nc|c(a - b)$, implying that $n|a - b$, or $a \equiv b \pmod{n}$. The price we pay in this case, however, is a change in the modulus from cn to n (Stallings, 2013).

Euler's theorem

Theorem 2. *Let a and n be positive integers with $\gcd(a, n) = 1$. Then,*

$$a^{\phi(n)} \equiv 1 \pmod{n}. \quad (2.8)$$

Proof. Let $r_1, r_2, \dots, r_{\phi(n)}$ be a reduced residue system modulo n . Then the set

$$ar_1, ar_2, \dots, ar_{\phi(n)}$$

is also a reduced residue system modulo n . Thus, we have:

$$(ar_1)(ar_2) \cdots (ar_{\phi(n)}) \equiv r_1 r_2 \cdots r_{\phi(n)} \pmod{n}. \quad (2.9)$$

Since $ar_1, ar_2, \dots, ar_{\phi(n)}$ must be congruent in some order to $r_1, r_2, \dots, r_{\phi(n)}$, it follows that:

$$a^{\phi(n)} r_1 r_2 \cdots r_{\phi(n)} \equiv r_1 r_2 \cdots r_{\phi(n)} \pmod{n}. \quad (2.10)$$

Since $r_1 r_2 \cdots r_{\phi(n)}$ is invertible modulo n , we can divide both sides by this product, yielding:

$$a^{\phi(n)} \equiv 1 \pmod{n}. \quad (2.11)$$

(Lewinter and Meyer, 2015) This completes the proof.

2.9 AES Algorithm Description

Advanced Encryption Standard (AES) is used widely in cryptography as a symmetric-key standard. Each cipher in AES uses a 128-bit block size, and the key can be 128, 192, or 256 bits in size, respectively. The code in the AES algorithm is encrypted in a very secure manner. AES is a block cipher intended to replace DES for commercial applications. Each full round consists of four functions: byte substitution, permutation, arithmetic operations over a finite field, and XOR with a key (Vijayakumar, 2022).

The block diagram in Figure 2.11 illustrates the AES algorithm. The cipher consists of rounds, where the number of rounds depends on the key length: 10 rounds for a 16-byte key, 12 rounds for a 24-byte key, and 14 rounds for a 32-byte key. All rounds consist of four distinct transformation functions: SubBytes, ShiftRows, MixColumns, and AddRoundKey, except the final round, which consists of three transformation functions: SubBytes, ShiftRows, and AddRoundKey.

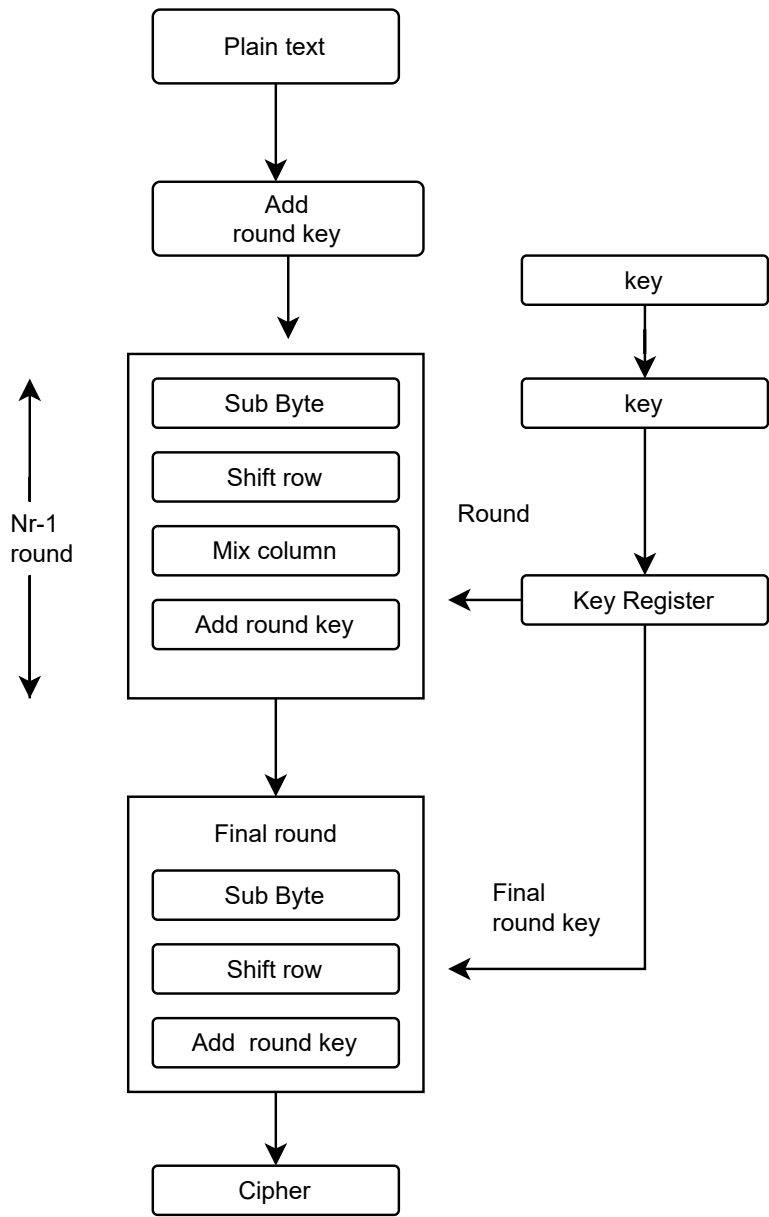


Figure 2.11: Block Diagram Of AES Algorithm

The steps of the AES algorithm are as follows:

1. **Key Expansion**
2. **Initial Round**
3. **Add Round Key**
4. **Rounds**
 - (a) **Sub Bytes** – A nonlinear replacement step in which another replaces each byte from a lookup table.
 - (b) **Shift Rows** – A transposition step where a cyclic shift is performed on each state row by a certain number of positions.
 - (c) **Mix Columns** – Operates on the columns of the state, combining the four bytes of each column to produce new values.
 - (d) **Add Round Key** – Each state byte is XORed with a round key derived from the cipher key using a key schedule.
5. **Final Round (without Mix Columns)**
 - (a) **Sub Bytes**
 - (b) **Shift Rows**
 - (c) **Add Round Key** (L., 1987)

2.10 RSA Algorithm Description

The RSA algorithm is considered one of the strongest algorithms used for encryption, as it relies entirely on the public key and works with the block cipher system. The RSA algorithm uses a pair of keys: a private key and a public key that is known to everyone and used to encrypt plaintext, which have to decrypt the cipher text and this key remain in secure (Sharma et al., 2012).

The power of the RSA scheme is based on a hard computational problem: the computational intractability of factoring a large integer. Large integer factorization is still an unresolved and challenging problem in modern number theory, even after the advent of efficient high-speed computers. If we know that a message is encrypted by using the public key encryption scenario, then we may not be able to decrypt the ciphered text even if we know the public key; therefore, the security of the system is still high, the Figure 2.12 illustrates the block diagram of RSA algorithm (Paar and Pelzl, 2014).

2.10.1 RSA Public Key Generation

The public key is obtained in RSA by multiplying two prime integers: $n = p * q$, where p and q are different prime integers. Note that n is composite, not prime.

We need to know two things to encrypt a message: the composite integer n and e , which is a public key. The public key e should be coprime with $\Phi(n)$, the Euler totient function. Now, to decrypt the message, we need two things: n (same as that used for encryption) and a private key d . This private key is the inverse of the public key e modulo $\Phi(n)$. Hence, we need to find the factors of n , i.e., the two

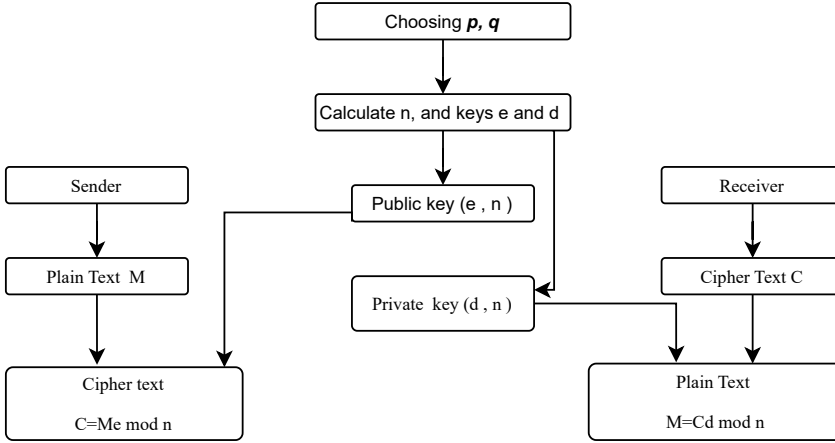


Figure 2.12: Block Diagram of RSA Algorithm

primes p and q . However, this is a hard mathematical and computational problem. The difficulty increases significantly with the size of n . Hence, the size of n will decide the level of RSA security (Paar and Pelzl, 2014).

2.10.2 Main Features of RSA System

1. The public key is known to all, including the attacker. The system's security relies on the difficulty of solving the following problem: finding the private key d given the public keys e and n .
2. If the number of bits allowed in the message (plaintext) is m , then we must have:

$$m < l$$

where l is the bit length of n . This constraint exists because encryption and decryption operate in modulo n .

3. In encryption, we compute:

$$c \equiv m^e \pmod{n}$$

and in decryption:

$$m \equiv c^d \pmod{n}$$

Therefore, an efficient exponentiation algorithm is required for practical implementation.

4. To prevent a brute-force attack, given a specific large integer n , there should be many possible combinations for private-key/public-key pairs (Kumar et al., 2011).

2.10.3 Problem Associated with RSA Algorithm:

The RSA algorithm has several disadvantages, which can be summarized as follows:

1. Multiplicative Property

The multiplicative property of RSA signatures, also known as the homomorphic property, can be expressed as follows:

If

$$S_1 \equiv m_1^d \pmod{n}$$

and

$$S_2 \equiv m_2^d \pmod{n}$$

then for message signatures m_1 and m_2 , we have:

$$S_1 S_2 \equiv (m_1 m_2)^d \pmod{n}$$

This property makes it computationally convenient to derive the private key d when two distinct signed messages are available from the same person. Since the values S_1, S_2, n, m_1 , and m_2 are already known, an attacker could exploit this property to extract d .

2. Integer Factorization

To deduce the private key d , an attacker needs to factorize the public modulus n to compute the Euler totient function $\varphi(n)$. Once $\varphi(n)$ is determined, the extended Euclidean algorithm can be used to compute d by solving:

$$ed \equiv 1 \pmod{\varphi(n)}$$

If an attacker successfully factors n , they can break the encryption system. To prevent this and enhance security, the prime factors p and q must be sufficiently large so that factorizing n becomes computationally infeasible (Marton et al., 2012).

2.11 Randomness Statistical Tests

Cryptographic keys must be unpredictable for the attacker, meaning high information content and high uncertainty; usually, entropy is the measure of these properties. Thus, the sequences chosen for cryptographic key must also exhibit independence of values, uniform distribution, and irreproducibility (Vijayakumar, 2022).

In most cryptographic functions, the key length is an important security parameter. Encryption strength is often described in terms of the size of the keys. In general, longer keys increase the possible space of keys, which provides stronger encryption when the key length is measured in bits (Bassham et al., 2010).

Random sequences are often used in cryptography to diversify the behavior of algorithms, generate secret keys, or provide random seeds, for instance (Vijayakumar, 2022).

There are two important properties of the randomness sequences (independence and equally likely); these properties can be represented mathematically as below:

S_n : sequence of random variables, where $S_n = X_1 + X_2 + \dots + X_n$

The possible values of the sequence X_n are 0 and 1.

$$P(X = 1 \mid \text{all others}) = 0.5$$

Any sequence that satisfies the above conditions can be considered random.

The properties of binary random sequences are:

1. **Independence:** The terms (bits) of a binary random sequence are independent, such that for any $i = j$,

$$P(x_i = x_j) = 0.5.$$

2. **Unpredictability:** If t bits of the sequence are known, it is impossible to predict another bit.
3. **Uniformity:** Zeros and ones are uniformly distributed across the sequence.

Randomness testing is based on statistical hypothesis tests, and each statistical test depends on some data function. There are many statistical tests listed below (Schoenberg, 1969):

2.11.1 Frequency Test (mono bit)

This test checks the randomness of a sequence based on the proportion of zeroes and ones in the entire sequence. It determines whether the number of ones and zeroes in the sequence is approximately the same as expected for a truly random sequence.

Let n be the length of the bit string, and let $\mathcal{Z}$ be the sequence of binary bits generated.

Convert the zeroes to -1 and the ones to $+1$ for all bits in the sequence, and sum them together to produce:

$$S_n = x_1 + x_2 + \cdots + x_n$$

where

$$x_i = 2 \times \mathcal{Z}_i - 1.$$

The test statistic is given by:

$$S_{\text{obs}} = \frac{|S_n|}{\sqrt{n}} \quad (2.12)$$

$$p\text{-value} = \text{erfc}\left(\frac{S_{\text{obs}}}{\sqrt{2}}\right). \quad (2.13)$$

2.11.2 Frequency Test within a Block

This test starts with dividing the sequence of length n into N blocks, each of length M ($N = \frac{n}{M}$). The purpose of this test is to check the proportion of ones within each M -bit block. The frequency of ones in every M -bit block should be $M/2$, as expected for a random bit sequence.

Consider a bit sequence $\hat{\zeta} = \hat{\zeta}_1, \hat{\zeta}_2, \dots, \hat{\zeta}_n$. The test determines the proportion of ones π_i in each M -bit block using the equation:

$$\pi_i = \frac{\sum_{j=1}^M (\hat{\zeta}_i - 1)(M + j))}{M} \quad (2.14)$$

Using the computed proportions π_i , the chi-square statistic is calculated as:

$$\chi_{\text{obs}}^2 = 4M \sum_{i=1}^N \left(\pi_i - \frac{1}{2} \right)^2 \quad (2.15)$$

A chi-square statistic determines if the distribution of observed frequencies differs from the theoretically expected frequencies. The P -value is computed using the incomplete gamma function:

$$P\text{-value} = \text{igamc} \left(\frac{N}{2}, \frac{\chi_{\text{obs}}^2}{2} \right) \quad (2.16)$$

If $P\text{-value} > 0.01$, we conclude that the tested sequence is random. For this test to yield accurate results, it is recommended that each sequence consists of at least 100 bits.

2.11.3 Serial Test (Two-Bit Test)

The aim of this test is to determine whether the number of occurrences of the sequences (00, 01, 10, 11) in the bit sequence is approximately the same as that in a random sequence.

The chi-square statistic is given by:

$$\chi^2 = \frac{4}{n-1} \sum_i (n_{ij})^2 - \frac{2}{n} \sum_i (n_i)^2 + 1 \quad (2.17)$$

A good sequence must satisfy the condition $\chi^2 \leq 5.99$.

2.11.4 The Run Test

This test aims to determine whether the number of zeroes and ones in a sequence are approximately the same as expected for a truly random sequence.

Let n be the length of the bit string and $\mathcal{Z}$ be the sequence of binary bits generated.

Let $v_n(\text{obs})$ represent the total number of runs.

Test Steps:

1. Compute the pre-test proportion π of the ones in the input sequence:

$$\pi = \frac{\sum \mathcal{Z}_j}{n}$$

2. Compute if $\left| \pi - \frac{1}{2} \right| \geq \tau$, where $\tau = \frac{2}{\sqrt{n}}$.

3. Compute the test statistic:

$$v_n(\text{obs}) = \sum_{k=1}^{n-1} r(k) + 1 \quad (2.18)$$

where $r(k) = 0$ if $Z_k = Z_{k+1}$, and $r(k) = 1$ otherwise.

4. Compute the p -value:

$$p\text{-value} = \text{erfc} \left(\frac{|v_n(\text{obs}) - 2n\pi(1 - \pi)|}{2\sqrt{2n\pi(1 - \pi)}} \right) \quad (2.19)$$

2.11.5 Auto Correlation Test

This test counts the "misses" (different bit values in the same position) between the test sequence and a shifted version of the same sequence. The test tries all shifts with $1 \leq d \leq N$. It calculates the following statistics:

For all bits of the binary sequence, calculate the number of ones n_1 and the number of zeros n_0 , then calculate the value μ when:

$$\mu = \frac{n_1^2(n-1)}{n^2} \quad (2.20)$$

After each shift d , which represents the shift value for the sequence, the new sequence (after shifting) is compared with the original sequence to calculate the number of ones with the same positions in both sequences. Then, calculate the value of x^2 when:

$$x^2 = \frac{(A(d) - \mu)^2}{\mu} \quad (2.21)$$

A good sequence is one where all values of $x^2 < 3.84$.

2.11.6 Test for Longest Run of Ones in a Block

As the name of the test indicates, it focuses on the longest run of ones within M -bit blocks. This test determines whether the length of the longest run of ones is in the same proportion as expected for a random sequence.

The test starts by dividing the sequence into bit blocks of length M . The length M of the blocks, together with the number of degrees of freedom K and the number of blocks N , is determined by the length of the sequence according to Table 2.2.

Table 2.2: The Block Length M Of Each Block According To The Minimum Length n Of The Tested Sequence

Minimum length n	M	K	N
128	8	3	16
6272	128	5	49
750000	10^4	6	75

The frequencies v_i of the longest runs of ones in each block are tabulated in categories in Table 2.3, where each cell contains the number of ones of a given length.

Table 2.3: The Frequencies v_i Of Longest Runs Of Ones In Each Block

v_i	M = 8	M = 128	M = 104
v_0	≤ 1	≤ 4	≤ 10
v_1	2	5	11
v_2	3	6	12
v_3	≥ 4	7	13
v_4		8	14
v_5		≥ 9	15
v_6			≥ 16

Afterwards, the χ^2 statistic is computed:

$$\chi_{\text{obs}}^2 = \sum_{i=0}^k \frac{(v_i - N\pi_i)^2}{N\pi_i} \quad (2 - 41) \quad (2.22)$$

The values K and N are determined by the value of M following Table 2.3. Finally, the P-value is computed:

$$\text{P-value} = \text{igam}\left(\frac{k}{2}, \frac{\chi_{\text{obs}}^2}{2}\right) \quad (2.23)$$

2.12 Mersenne Prime Number

Prime number: A prime number is defined as an integer greater than 1 that has no positive divisor other than 1 and the number itself.

A Mersenne number, named after Marin Mersenne, a French monk who studied these numbers in the early 17th century, is a positive integer that is one less than a power of two. Mersenne prime numbers are of the form $2^p - 1$, where p is a prime number. Therefore, the equation for a Mersenne prime number is:

$$M_p = 2^p - 1 \quad (2.24)$$

Since 1997, the "Great Internet Mersenne Prime Search" (GIMPS) has discovered all the new Mersenne primes through a distributed computing project on the Internet (Pund and Desai, 2014).

2.13 B-spline

As with the Bézier representation of polynomial curves, it is desirable to write a spline $s(u)$ as an affine combination of some control points C_i , namely

$$s(u) = \sum_i C_i N_i^n(u) \quad (2.25)$$

where the $N_i^n(u)$ are basis spline functions with minimal support and certain continuity properties. Schoenberg introduced the name B-splines for these functions (Schoenberg, 1988; Prautzsch et al., 2002; Teh and Chin, 1988).

B-spline Properties

We summarize the basic properties of B-splines:

- The B-splines of degree n with a given knot sequence that do not vanish over some knot intervals are linearly independent over this interval.
- A dimension count shows that the B-splines $N_0^n, \dots, N_m^n$ with the knots $a_0, \dots, a_{m+n+1}$ form a basis for all splines of degree n with support $(a_0, a_{m+n+1}]$ and the same knots.
- Similarly, the B-splines $N_0^n, \dots, N_m^n$ over the knots $a_0, \dots, a_{m+n+1}$, restricted to the interval (a_n, a_{m+1}) , form a basis for all splines of degree n restricted to the same interval.
- The B-splines of degree n form a partition of unity, i.e.,

$$\sum_{i=0}^m N_i^n(u) = 1, \quad \text{for } u \in (a_n, a_{m+1}).$$

- A spline $s[a_n, a_{m+1}]$ of degree n with n -fold end knots, ($a_0 = a_1 = \dots = a_n$) and $a_{m+1} = \dots = a_{m+n} = a_{m+n+1}$, has the same end points and end tangents as its control polygon.
- The end knots a_0 and a_{m+n+1} have no influence on N_0^n , and N_m^n over the interval $[a_n, a_{m+1}]$.
- The B-splines are positive over the interior of their support,

$$N_i^n(u) > 0, \quad \text{for } u \in (a_i, a_{i+n+1}).$$

- The B-splines have compact support,

$$\text{supp}(N_i^n) = [a_i, a_{i+n+1}].$$

- The derivative of a single B-spline is given by

$$\frac{d}{du} N_i^n(u) = \frac{n}{u_{i+n} - u_i} N_i^{n-1}(u) - \frac{n}{u_{i+n+1} - u_{i+1}} N_{i+1}^{n-1}(u).$$

- The B-spline representation of a spline curve is invariant under affine maps.
- Any segment $s_i[a_j, a_{j+1}]$ of an n -th degree spline lies in the convex hull of its $n + 1$ control points $c_{j-n}, \dots, c_j$.
- A degree elevation formula (Teague, 1980).

2.14 Zernike Moment

A Zernike moment (ZM) is a feature extraction method that is widely used for computer vision applications. When rotational invariance is required, moments defined over polar coordinates are used; examples are radial moments and Zernike moments (Hse and Newton, 2004).

Teague was the first to introduce Zernike moments, based on orthogonal Zernike polynomials, to handle information redundancy in the popular geometric moments (Lajevardi and Hussain, 2009).

Zernike moment is an efficient image feature extraction method that does not vary with image rotation. Arbitrary orders of these moments can be found.

Higher order components of ZM can detect fine details of objects in the image, but they have higher noise sensitivity as well (Khotanzad and Hong, 1990).

Zernike moments are shown to be relatively immune to Gaussian noise (Chong et al., 2003). In terms of redundancy, ZM has very low redundancy (near-zero), and the orthogonal moments, including Zernike moments, are better than other types of moments in terms of information redundancy and image representation (due to the orthogonality of basic functions) (Ambedkar and Bedi, 2011).

This is attributed to the fact that they are defined in terms of polar coordinates (Ambedkar and Bedi, 2011).

Description of ZM

ZM's definition is based on Zernike polynomials. These are a complex complete orthogonal set formed inside the unit circle $x^2 + y^2 = 1$. If we denote the polynomials by $\{V_{nm}(x, y)\}$, then their general form is given by:

$$V_{nm}(x, y) = V_{nm}(p, \theta) = R_{nm}(p)e^{jm\theta} \quad (2.26)$$

where:

- $j = \sqrt{-1}$
- n is a non-negative integer
- m is a positive or negative integer, depending on the condition $n - |m|$ even, $|m| \leq n$

The vector length p from $(0, 0)$ to (x, y) is given by:

$$X = \frac{2}{N-1}I - 1, \quad Y = \frac{2}{N-1}J - 1 \quad (2.27)$$

and:

$$p = \sqrt{x^2 + y^2} \quad (2.28)$$

The angle θ between vector p and the x-axis in the counterclockwise direction is given by:

$$\theta = \tan^{-1} \left(\frac{-y}{x} \right) \quad (2.29)$$

The radial polynomial $R_{nm}(p)$ is given by:

$$R_{nm}(p) = \sum_{s=0}^{\frac{n-|m|}{2}} (-1)^s \frac{(n-s)!}{s! \left(\frac{n+|m|}{2} - s \right)! \left(\frac{n-|m|}{2} - s \right)!} p^{n-2s} \quad (2.30)$$

The Zernike moment (ZM) for an image $f(x, y)$ of order n and repetition m is:

$$A_{nm} = \frac{n+1}{\pi} \sum_{(x,y)} f(x, y) V_{nm}(x, y), \quad x^2 + y^2 \leq 1 \quad (2.31)$$

The difference in ZM between a rotated image and the original is a phase shift only, and the magnitude stays the same, which makes $|A_{nm}|$ a rotation-invariant feature for $f(x, y)$ (Lauter, 2004; Ambedkar and Bedi, 2011).

2.15 Convolution

The convolution operation describes the convolution of two continuous functions $s(t) = \int_{-\infty}^{\infty} f(\alpha)\omega(t-\alpha)d\alpha$. However, representations that can be stored in memory consist of measurements from discrete points in time and are of limited size. In addition, the indices are oftentimes flipped for easier computation on arrays. Therefore, the operation actually realized by convolutional neural networks is a cross-correlation rather than a convolution (University, 2024).

The formulation for the default use case of image processing, i.e., two-dimensional data, is:

$$s(i, j) = (I * K)(i, j) = \sum_{m=1}^M \sum_{n=1}^N I(i + m, j + n)K(m, n)$$

This equation describes the convolution of the input I with the kernel K . Intuitively, the filter with a small spatial extension compared to the original input is moved across the input. For each position, a dot product of the values covered by the filter and the filter weights is computed.

2.16 Security Analysis

Any cryptosystem depends on the key and the algorithm. Therefore, the security of the key depends not only on the key's length but also on the encryption algorithm as a fundamental base in the cryptosystem. To secure data transmission in cloud computing, a secure encryption algorithm and a secure key length must be determined (Kleijnung et al., 2010).

Classical asymmetric RSA with 1024 bits (RSA-1024) provides sufficient security for current applications. RSA security fully depends on the difficulty of integer factorization and the absence of alternative attack methods. The time required for factorization depends on the size of the number.

In 2009, Kleijnung et al. could factor a 768-bit number with 232 decimal digits (RSA-768) using the Number Field Sieve (NFS). However, they required 24 months of collaborative work and hundreds of powerful computing machines to achieve this. This demonstrates that factoring large prime numbers remains computationally difficult (Khotanzad and Hong, 1990).

The Kleijnung group estimated that factoring a 1024-bit RSA modulus is approximately 1,000 times harder than their effort with the 768-bit modulus. Using the same hardware and conditions, they estimated it would take about 1,500 years. Their assumption was based on a 2.2 GHz AMD Opteron processor with 2 GB RAM. This remains the current record for factoring integers. However, a 1024-bit RSA modulus is expected to be factored within the next few years (Moghaddam et al., 2013).

In 2012, Childers successfully factored a 1061-bit number using the Special Number Field Sieve (SNFS). He used SNFS to determine the full factorization of a Mersenne number $2^{1061} - 1$ using available software.

The current factoring records are:

- 768-bit, 232-digit number from RSA's challenge list in 2009, which took 3300 CPU years.
- 200-digit number in 2005, which took 170 CPU years.
- Mersenne number $2^{1061} - 1$ factored by SNFS in 2012, which took 400 CPU years.
- 1024-bit, currently in progress. With existing techniques, this may take more than five years, but algorithm advancements could accelerate the process (Masek, 2003).

The security of RSA has been analyzed against three types of attacks: brute force, mathematical, and timing attacks.

Mathematical Attack

This type of attack exploits the underlying structure of RSA. The most straightforward approach involves factoring the modulus N . Several factorization methods exist, all relying on decomposing N into its two prime factors. The security of RSA can be improved by using a 2048-bit key length, making such attacks computationally infeasible (Bakstein, 2014).

Timing Attack

A timing attack is a side-channel attack that determines the private exponent by analyzing execution time variations in modular exponentiation. This attack can be mitigated by introducing random delays in the exponentiation algorithm or by multiplying the ciphertext with a random number before decryption.

Brute Force Attack

A brute force attack (BFA) involves testing all possible public and private key combinations to guess the private key. Short secret keys are particularly vulnerable to this attack. However, increasing the key length results in higher computational costs for encryption, decryption, and key generation. The security of RSA against brute force attacks improves significantly when an exponent of at least 2048 bits is used.

Chapter 3

Design of the Proposed System

The contents of this chapter are published in articles P2 and P4.

Generally, the iris is a biometric measurement used to identify and verify individuals. The automated process determining whether two irises belong to the same individual is called iris authentication.

3.1 Introduction

Therefore, the proposed system utilizes iris features to enhance the security of two encryption algorithms (AES and RSA) used in cloud computing. When extracting features from the iris using the Zernike moments method, these features authenticate users attempting to log in to the cloud computing system. This step is the initial measure to enhance cloud computing security, as illustrated in Figure 3.1.

Furthermore, these features are treated as control points, which are used in a B-spline to draw a curve, generating the AES-128 encryption key to encrypt the data stored in the cloud. Additionally, the control points are applied in the Bézier curve to draw a curve, from which the points are utilized to derive the prime numbers p and q . The nearest Mersenne prime number to p is selected as the second prime number, q . Using a predefined step, I generate the points of the curve one by one in the $[0, 1]$ closed interval starting at parameter 0. If none of the calculated points satisfy the requirements until the end of the interval, a different step is applied and the generation started again from 0 until I find a good starting point.

These prime numbers are then used in the RSA algorithm to encrypt the AES encryption key. The confidentiality process is illustrated in Figure 3.2.

3.2 The Proposed System

The proposed approach selects the iris as the biometric to extract authentication features and generate cryptographic keys in cloud computing. The following subsections illustrate the main steps of the proposed system.

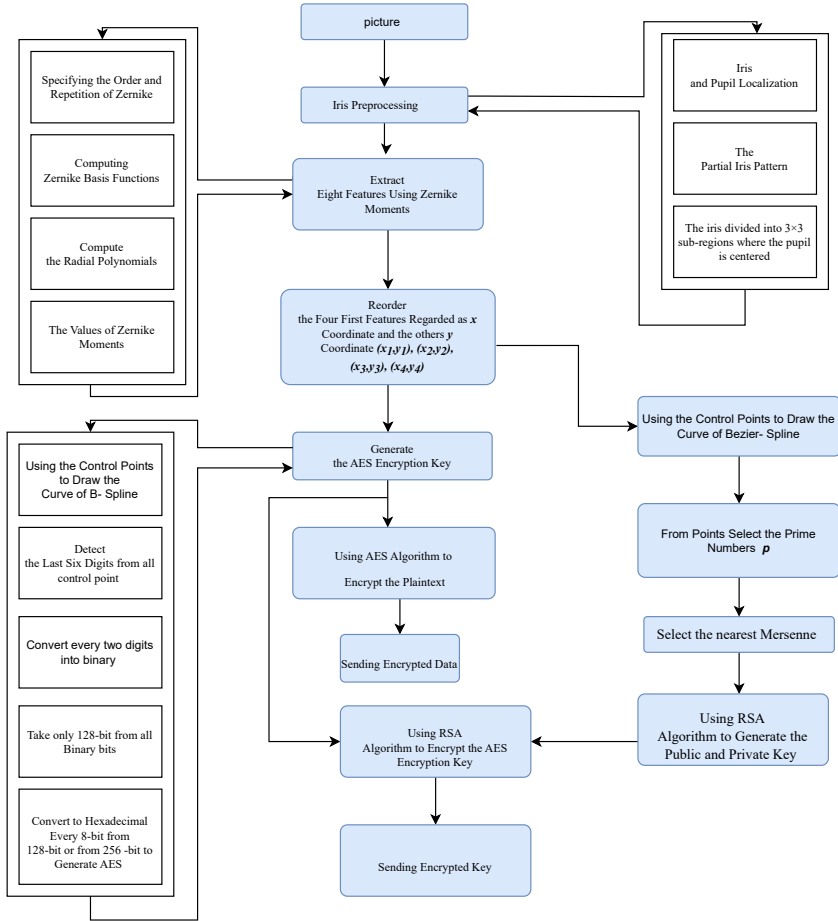


Figure 3.1: Block Diagram Of The Authentication Scheme

3.2.1 Read the Eye Images

The proposed system deals with the iris as a biometric tool. This biometric data is obtained from WVU, CASIA, and MMUD datasets. In most of these databases, the iris image contains not only an iris but the whole eye and part of the surrounding eye region.

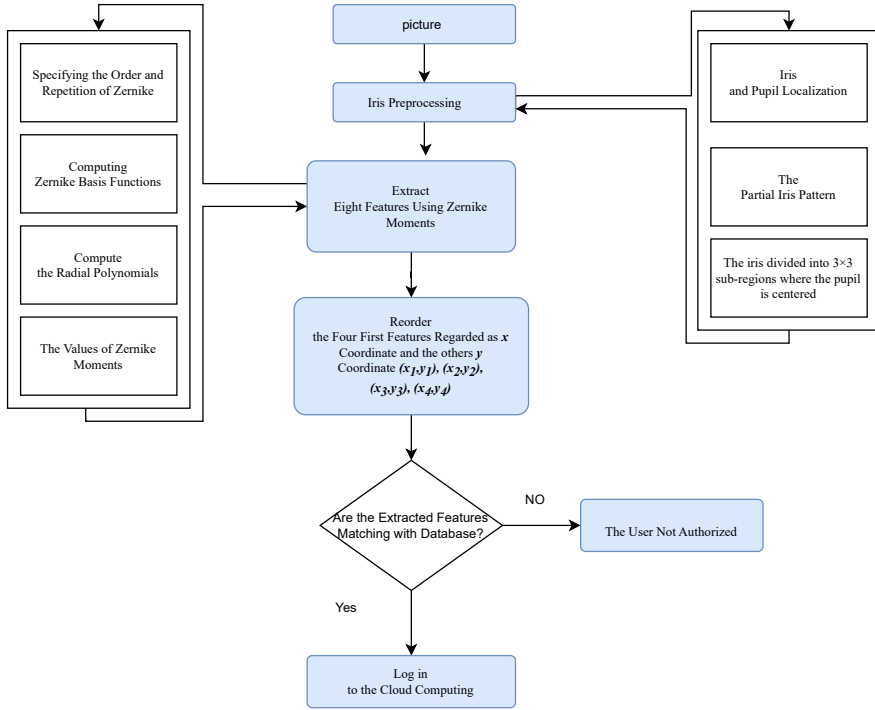


Figure 3.2: Block Diagram Of The Confidentiality Scheme

Therefore, a preprocessing process such as pupil and iris localization is required before extracting features. Figure 3.3 shows the examples used for the iris image.

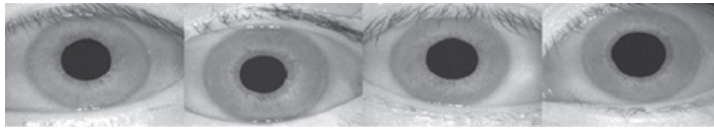


Figure 3.3: Iris Images, From CASIA Database. Source: CBSR Website.

3.2.2 Iris Image Preprocessing

Iris images come from different image scanning systems. They probably do not have the same adequate quality for the next stages. Hence, preprocessing is needed. Image preprocessing is an essential method of preprocessing before feature extraction. It improves image quality.

Therefore, the following enhancement processes explain the preprocessing of iris enhancement before applying the proposed system.

Iris and Pupil Localization

The iris recognition system must identify the region of interest (ROI) by locating the pupil and iris in the eye image. In an uncontrolled environment, image capturing cannot be expected to produce an image that encloses only the iris. We are dealing with an eye image containing the entire eye and the surrounding regions.

Thus, iris localization is a critical step that should be capable of dealing with variable occlusion, low intensities, edges, etc. Suppose certain parts of the iris are occluded (e.g., by eyelids or hair). In that case, the system must correctly locate the iris regions no matter the type of occlusion and return the corresponding parameters modeled by the center coordinates x_c and y_c and the radius r . These parameters define the circle equation:

$$x_c^2 + y_c^2 = r^2$$

The pupil should also be located accurately, as the pupil is typically darker than the iris. However, recent advances in iris recognition have proposed different methods to deal with images captured in unconstrained environments. In this work, the implemented method was adopted from the work of Masek (Masek, 2003), which is based on using the Circle Hough Transform (CHT) along with the Canny edge detector.

This method is robust for detecting circular objects in an image and is insensitive to noise. Thus, the system returns six coordinates: x_c , y_c , and the radius r of the pupil, along with x_c , y_c , and the radius r of the iris. Theoretically, it is assumed that the center coordinates of the pupil and the iris are the same. However, in practice, the algorithm returns different center coordinates. In this work, the average of the center coordinates is calculated and used for both. Figure 3.4 shows an example of iris and pupil localization results.

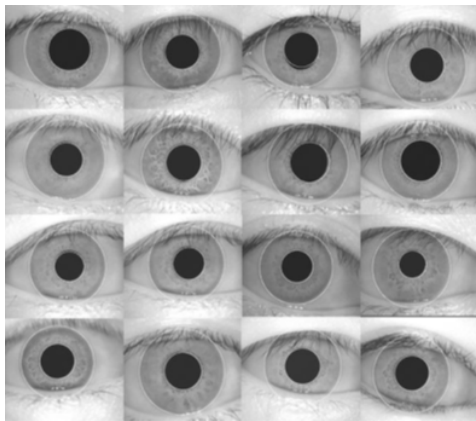


Figure 3.4: The Outline Boundaries Show Results Of The Iris And Pupil Localization

Only that portion of the iris between the inner and outer boundaries must be analyzed for the subsequent steps. However, false iris localization can hardly be avoided since we deal with unlimited variations that may lead to unpredictable results. Figure 3.5 shows an example of false iris localization.

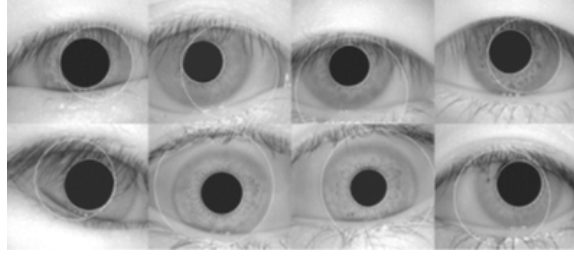


Figure 3.5: False Iris Localization

The Partial Iris Pattern

In practical experience, acquiring an iris image is a significant issue for automated iris recognition. The iris is a small object. Capturing iris images should be performed with high enough resolution, at a distance, under unconstrained conditions, and finally, without disturbing people (as individuals are sensitive about their eyes).

The challenges associated with iris image acquisition can be attributed to the following factors: scaling, rotations, translation, pose, illumination, and occlusions (i.e., the iris may be partially occluded by objects such as eyelashes, eyelids, hair, or eyeglasses). Furthermore, various factors may cause degradation during image capturing, such as incorrect focus, motion, camera characteristics, and random sensor errors. Therefore, the quality of the training dataset is possibly one of the biggest obstacles to a successful application.

A challenging issue affecting dataset quality is the high variability in iris images, which can be attributed to the factors above. To the best of our knowledge, the literature lacks adequate evidence to address the limitations of iris images. Figure 3.6 shows examples of non-ideal iris images where 30–55% of the iris texture has been lost.

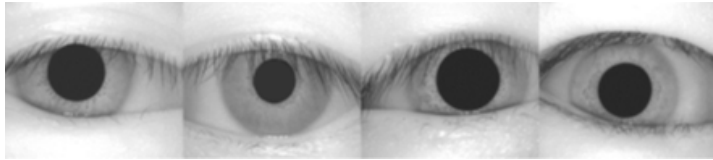


Figure 3.6: Non-Ideal Iris Images Example (i.e., 30–55% Of Iris Texture Has Been Lost)

The first step for improving the quality of iris images is to reduce the amount of iris variations between images. This would provide the most compact space for iris images. To our knowledge, most previous works have used whole iris patterns for feature extraction and encoding procedures. Therefore, the non-ideal iris images, such as those shown in Figure 3.4, may degrade the system's accuracy.

As a result of the general fact that the eye is oval-shaped while authentic iris images are 2D rectangular images, one way to decrease iris image variability is by utilizing a binary mask similar to the concept of stencil buffer of OpenGL. Generally, masking excludes areas with noise (i.e., eyelids, eyelashes, etc.). Practically, I use 1 as a masking bit to mark useful ROI areas (i.e., the region of interest) and I use 0 as a masking bit for area not part of ROIs (i.e., non-iris). Excluding (or

ignoring) non-iris areas from the training images ensures that the system does not wrongly introduce any unwanted structures into the iris representation, thereby reducing iris variability.

The shape of the mask can be variable or fixed. Figure 3.7 shows a sample of a fixed-shape mask proposed by Masek (2003).

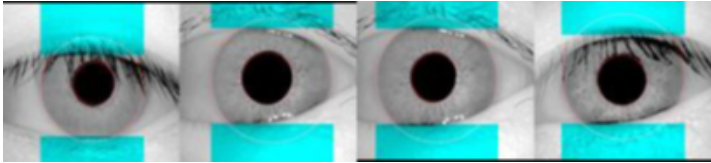


Figure 3.7: Fixed Masking Used For Excluding Areas With Noise

Figure 3.8 shows a sample of variable masking proposed by Bakštein (2014).

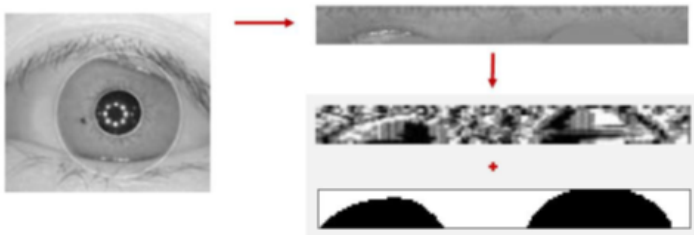


Figure 3.8: Variable Shape Masking Used for Excluding Areas with Noise

Although the masking technique improves the quality of iris samples, this solution still implies a drawback to pattern variability. Based on this conclusion, a novel iris pattern, i.e., partial iris pattern, has been suggested instead of the whole iris pattern. To our knowledge, this is the first attempt to employ a partial iris pattern for human identification. The basic idea behind that is that the iris constitutes a circular shape that can be easily bounded in a square iris region IR (Figure 3.9a). The IR region is divided into nine sub-regions (i.e., 3×3), where the pupil would be centered at IR(2, 2), as shown in Figure 3.9b.

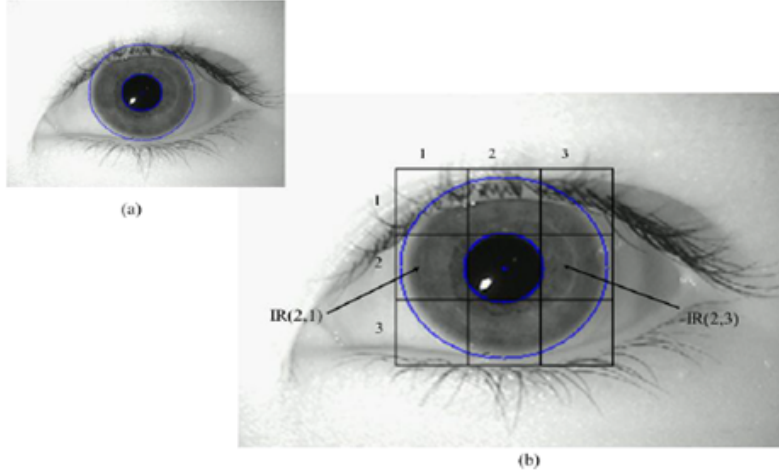


Figure 3.9: The Proposed Partial Iris Pattern (a) The Iris Pattern; (b) The Iris Region IR Is Divided into 3×3 Sub-regions where the Pupil is Centered At IR(2,2) Entry

This step aims to compute the standard deviation (σ) of each squared sub-region to identify the ones with the lowest variation in both cases: the same individual and different individuals. Consider that we have a set of iris images stacked on each other, as shown in Figure 3.10. There are m pixels for any given coordinates (i, j) , with one pixel at that location for each image.

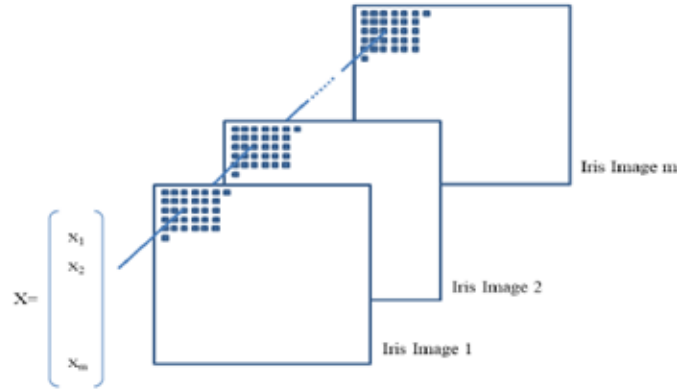


Figure 3.10: Stack of Iris Images Of The Same Size

The average and standard deviation of each sub-region are calculated as follows:

Step 1: Obtain iris images $I_1, I_2, \dots, I_m$ (training dataset), where m is the number of iris images. The iris images should have the same size, with the pupil centered at IR(2,2).

Step 2: Represent every image I_i as a vector Γ_i .

Step 3: Compute the average iris vector Ψ :

$$\Psi = \frac{1}{m} \sum_{i=1}^m \Gamma_i$$

Step 4: The standard deviation vector (σ) is presented as an iris image.

Experimentally, the calculated standard deviations revealed that the regions $IR(2,1)$ and $IR(2,3)$ show the lowest variations (Figure 3.9b). In pattern recognition problems, the regions with higher variations degrade the quality of the training data. Furthermore, it was found that about 50% of the corner sub-regions are non-iris.

Thus, we conclude that using a partial iris pattern instead of the whole iris pattern will improve the quality of the input data. The partial iris pattern proposed in this study comprises the $IR(2,1)$ and $IR(2,3)$ sub-regions. Figure 3.11 shows examples of the proposed partial iris pattern in the $IR(2,1)$ region.

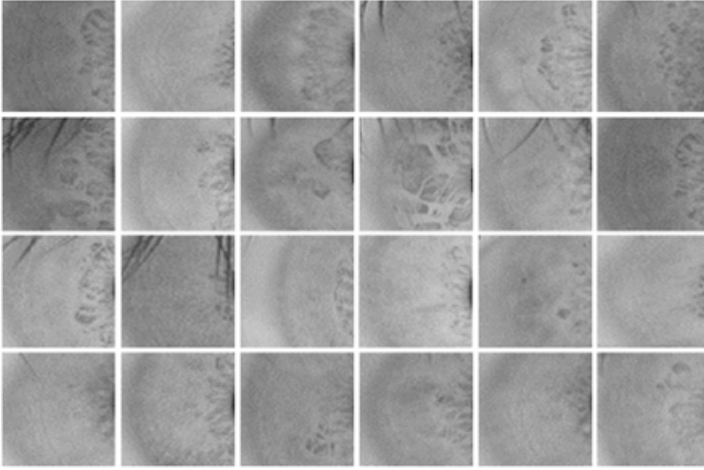


Figure 3.11: The Proposed Partial Iris Pattern $IR(2,1)$ Region

Figure 3.12 shows examples of the proposed partial iris pattern $IR(2,3)$. Compared to the binary masking technique, the partial iris pattern shows a minimal noise area or is completely free of noise.

In practice, iris encoding must resist changes in the iris size, which depends on the distance to the eye and zoom. Thus, we must create an iris pattern invariant to its size in the eye image. After locating and extracting the ROI, the size of the partial iris pattern is experimentally fixed at 60×60 pixels to preserve sufficient resolution for extracting discriminative features while keeping the dimensionality of the iris space manageably small.

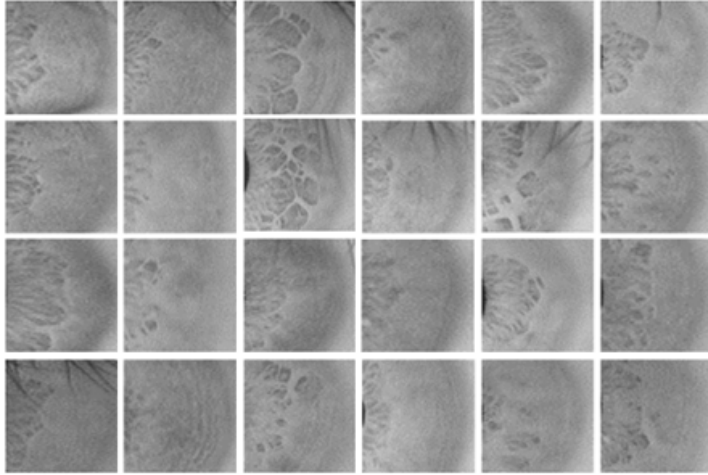


Figure 3.12: The Proposed Partial Iris Pattern $IR(2, 3)$ Region

3.2.3 Extract Features from the Iris

After enhancing the iris, we need to extract eight features that will be used for the authentication and generate encryption keys by applying B-spline and Bézier curve. Zernike moments are applied to feature extraction from an iris image. Zernike moments are constructed using set orthogonal Zernike polynomials with a property state that each moment is unique and independent of the information in the image. This property eliminates the redundancy and overlap of information between moments in different orders and repetitions. Section 2.14 in chapter two discussed the Zernike moments.

After detecting the values of Order and Repetition, several steps exist to compute the Zernike moments from the image. These steps are listed as follows:

- **Step 1:** Before computing the Zernike moments, pre-processing is required. This pre-processing is illustrated in Algorithm 1.
- **Step 2:** Detect the values of p and q to obtain the set of polynomials. This step is illustrated in Algorithm 2.
- **Step 3:** Compute the radial polynomials. This step is illustrated in Algorithm 3.
- **Step 4:** Compute the Zernike basis function. This step is illustrated in Algorithm 4.
- **Step 5:** Compute the Zernike moments by projecting the image onto the basis function. This step is illustrated in Algorithm 5.

Algorithm 1 Pre-Processing For Zernike Moments

```

1: procedure PREPROCESSZERNIKE( $f_i$ ) ▷ Iris Image
2:   Convert image  $f_i$  into gray level.
3:   Find size of image  $f_i$  ( $N$  and  $M$ ).
4:   Find the smaller value of  $N$  and  $M$ .
5:   Make  $N$  and  $M$  even using the following condition:
6:   if  $\text{mod}(N, 2) \neq 0$  then
7:      $N = N - 1$  ▷ Ensure even dimensions
8:   Make the image matrix square.
9:   Convert values of  $f_i$  into double type to produce  $f$  for mathematical operations.

```

Preprocessing for the Zernike Moments

In the beginning, to compute the magnitude of the Zernike moment, there must be a preprocessing for the Zernike moment. Algorithm 1 illustrates this preprocessing.

Algorithm 2 Specifying The Order (p) And Repetition (q) Of Zernike

```

1: procedure SPECIFYORDERREPETITION( $p, q$ ) ▷ Order and Repetition
2:   Create two vectors  $n$  and  $m$ , where  $n$  is the Zernike order and  $m$  is the Zernike repetition.
3:   Compute length of Zernike set ( $m$  or  $n$ ) based on:
       $L_z = \text{length}\{Z_{P_0 P}\}$ 
       $= \text{length}\{Z_{pq} | P_0 \leq p \leq P; \text{mod}(p - |q|, 2)\}$ 
       $\{p\} = \{P_0, \dots, P\}$  ▷ Range of Zernike moments under test

```

Example: $H_0 = 1, H = 4$, Using the above conditions, get:

- $n = [1, 2, 2, 3, 3, 4, 4, 4]$
- $m = [1, 0, 2, 1, 3, 0, 2, 4]$
- $L_z = 8$ Number of Zernike moments

Specifying the Order and Repetition

In the first step to extract the features by Zernike moments, it is necessary to specify the order (p) and repetition (q) to get the set of polynomials. The flowchart in Figure 3.13 illustrates this process.

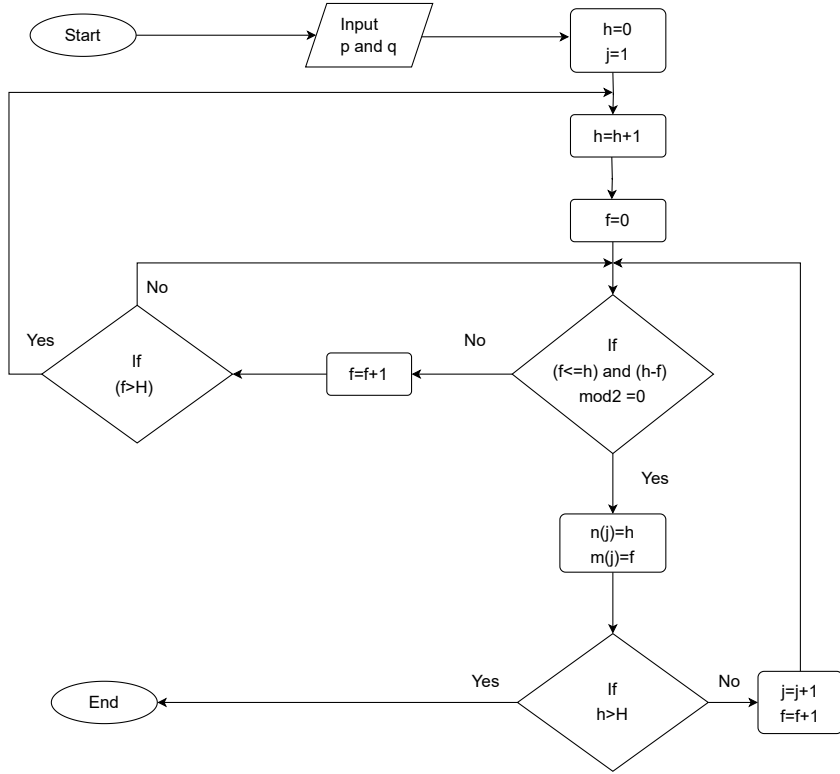


Figure 3.13: Flowchart To Get The Set Of Polynomials

Table 3.1 shows the 8 orders of the moments when $p = 1$ and $q = 4$. When applying Algorithm 1, the output of this algorithm is a set of polynomials, which are represented as vectors of n and m .

Table 3.1: List of The 8 First Orders Zernike Moments

n	m	Zernike moments
1	1	$A_{1,1}$
2	0	$A_{2,0}$
2	2	$A_{2,2}$
3	1	$A_{3,1}$
3	3	$A_{3,3}$
4	0	$A_{4,0}$
4	2	$A_{4,2}$
4	4	$A_{4,4}$

Compute the Radial Polynomials

Solving the radial central moments represents the first step in determining the Zernike central moments. The radial moment for the condition $(p - q = \text{even})$ is applied in the Zernike moments, which are considered solely. $R_{pq}(r)$ denotes the radial central moments with order p and repetition q .

The computation of radial and Zernike moments requires the linear transformation that maps coordinates from (i, j) , where $i, j = 0, 1, 2, \dots, N - 1$, to an appropriate domain $(x, y) \in \mathbb{R}^2$. The new domain should be inside a unit circle since radial and Zernike moments are expressed in polar coordinates (r, θ) with $|r| \leq 1$.

Figure 3.14 shows the process of coordinate transformation. The algorithm to compute the radial polynomials with complex radial moments works on the center of the image. The measurement vector with different orders utilized here is the magnitude invariant to rotation. With order value p and repetition value q , the Zernike moment is a feature extractor from the image.

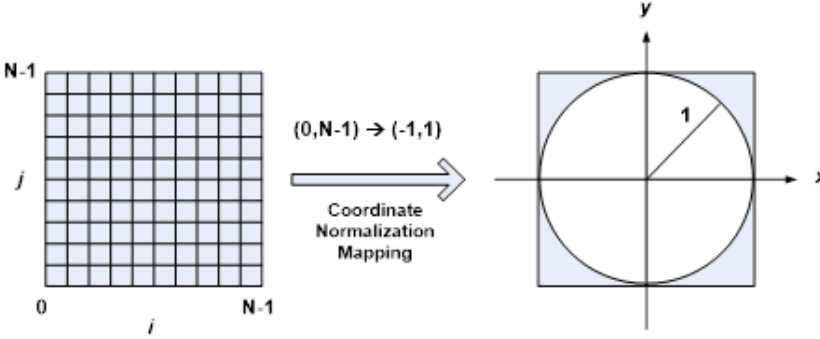


Figure 3.14: Two Vectors Normalization Plan For Radial And Zernike Moments (a) Discrete Image For The Space Of The Size $(N \times N)$; (b) The Map Used For The Coordinate Normalization $(0, N - 1) \rightarrow (-1, +1)$

After specifying the Order (p) and Repetition (q), the next step is to compute the radial polynomials, Algorithm 3 and Figure 3.15 illustrates this process.

Algorithm 3 Compute The Radial Polynomials

- 1: **procedure** COMPUTERADIALPOLYNOMIALS(p, q) = R_{pq}
- 2: **Step One:** Call Algorithm 1 to create two vectors n and m , where n is the Zernike order and m is the Zernike repetition.
- 3: **Step Two:** Compute r_{ij} value from the following equation:

$$r_{ij} = \sqrt{X^2 + Y^2} \quad (3.1)$$

where

$$X = \frac{2}{N-1} \cdot I - 1, \quad Y = \frac{2}{N-1} \cdot J - 1 \quad (3.2)$$

- 4: **Step Three:** Calculate the Radial Polynomials:
 - 5: **for** $k = 0$ **to** $(n - |m|)/2$ **do**
 - 6: $S = (-1)^k \cdot \frac{(n-k)!}{k! \left(\frac{n+|m|}{2} - k\right)! \left(\frac{n-|m|}{2} - k\right)!} \cdot r^{n-2k}$
 - 7: $R_{pq} = R_{pq} + S \cdot r^{p-2k}$
-

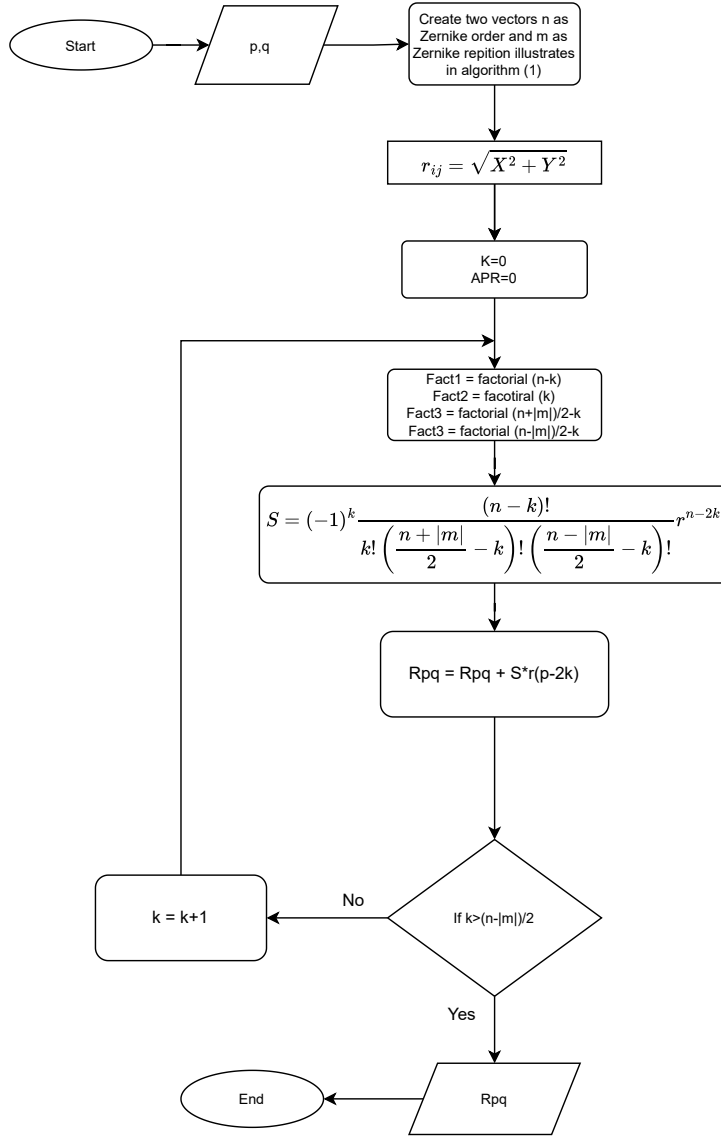


Figure 3.15: Flowchart To Compute The Radial Polynomials

Compute Zernike Basis Functions

This step needs to use radial polynomial, which is complex-valued 2-D Zernike basis functions, and the value of x and y ; also, the iris must be even to compute the Zernike basis function. They are defined in a unit circle and illustrated in Algorithm 4 and the flowchart in Figure 3.16.

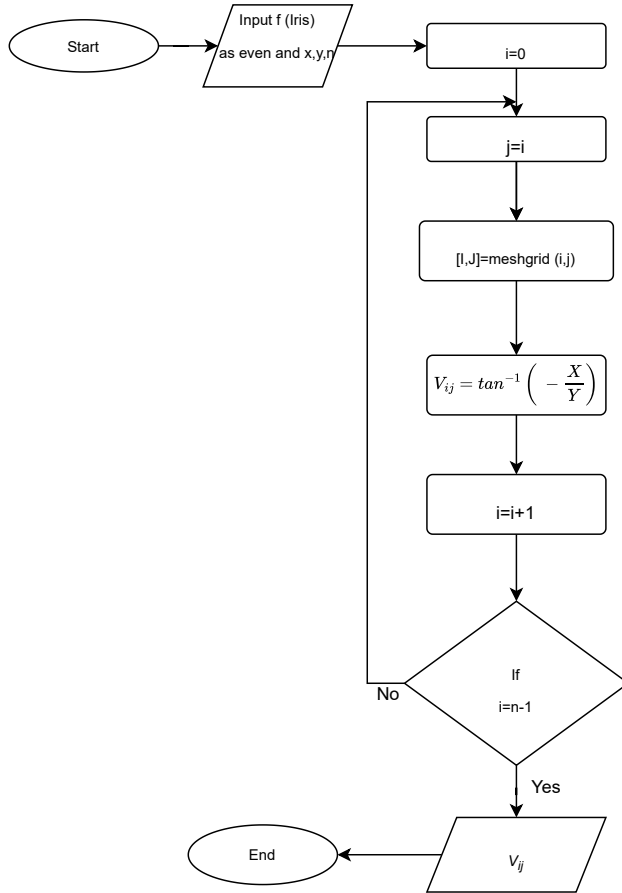


Figure 3.16: Flowchart To Compute Zernike Basis Functions

Algorithm 4 Compute The Zernike Basis Functions

```

1: procedure COMPUTEZERNIKEBASISFUNCTIONS(f, X, Y)    ▷ Iris image and
   coordinates from Algorithm 2
2:   Input:  $f$                                           ▷ Iris image (even dimension  $N$ )
3:   Input:  $X, Y$                                        ▷ From Algorithm 2
4:   Output:  $V_{ij}$                                      ▷ Zernike basis functions
5:   Step One: Make the Iris image as a mesh grid:
6:   for  $i = 0$  to  $N - 1$  do
7:      $j = i$ 
8:      $[I, J] = \text{meshgrid}(i, j)$ 
9:   Step Two: Calculate the Zernike basis functions:

```

$$V_{ij} = \tan^{-1}\left(-\frac{Y}{X}\right) \quad (3.3)$$

Compute the Zernike Moments

The amplitude values results of the Zernike moments when the order p and repetition q can be illustrated in the Algorithm 5.

Algorithm 5 Compute The Zernike Moments

```

1: procedure COMPUTEZERNIKEMOMENTS(f, p, q)    ▷ Iris image, order, and
   repetition
2:   Input:  $f$                                           ▷ Iris image
3:   Input:  $p, q$                                        ▷ Order and Repetition
4:   Output:  $Z_{pq}$                                      ▷ Amplitude of Zernike moments
5:   Step One: Pre-processing for Zernike moment:
6:     i. Convert image  $f$  into gray level.
7:     ii. Find the size of the image ( $N$  and  $M$ ).
8:     iii. Find smaller values of  $N$  and  $M$ .
9:     iv. Make  $N$  and  $M$  even using the following condition:
10:  if  $\text{mod}(N, 2) \neq 0$  then
11:     $N \leftarrow N - 1$                                 ▷ Ensure even dimensions
12:    v. Make image matrix square.
13:    vi. Convert the values of image  $f$  into double type for math operations.
14:  Step Two: Execute Algorithm 2 to compute the radial polynomials.
15:  Step Three: Call Algorithm 3 to compute Zernike basis functions.
16:  Step Four: Calculate the Zernike moments amplitude:

```

$$Z_{pq} = \frac{p+1}{N-1} \sum_{i=0}^{N-1} \sum_{j=0}^{N-1} R_{pq}(r_{ij}) \cdot e^{-jp\theta_{ij}} \cdot f(i, j) \quad (3.4)$$

3.2.4 Authentication Stage

The authentication has two stages. The face is considered one of the most important axes in daily social interactions, as it conveys emotions and a person's

identity among thousands of people. Additionally, changes such as aging, wearing glasses, changing head movements, or altering hairstyles are considered, with ten cases taken for each person to account for all possible variations.

Convolutional Neural Network (CNN)

We identified that an individual's eye recognition is important in security systems, credit card control, and guilty identification applications. This method, tested on different images, provided the highest accuracy for image recognition, which is used in security systems.

First, some data are derived from the images. We cover the standard space of the eye with 270×70 pixels (i.e., the length from the left eye to the right eye). There are 40 different humans, each with 10 different faces, so we have 400 labels. The number of image channels is one because the images are black and white. The aim is first to separate the face from the eyes. Sometimes, in some images, the eyes are not visible, for example, if the person is wearing glasses or their eyes are closed. For this reason, this process may result in errors. In the proposed method, we recognize these images and eliminate them from the dataset.

In the following histogram chart, we expect 10 images for each person. However, some persons have no images or have fewer images. Therefore, those with fewer than 5 images are eliminated from the dataset; as a result, we have 37 persons whose images are available.

The CNN is configured with six layers in the training process, including input, convolution, activation function, and pooling. In the first layer, the input consists of images with a size of 270×70 pixels. The second layer is for convolution, where the filter size is 3×3 , and the number of filters is 96. Padding is also applied. The third layer performs the activation function, such as the ReLU function, which is used to minimize errors in the output. The bias and weights are updated in every layer, and the modified parameters are passed as input to the next layer.

The CNN training algorithm is similar to 2D image training; however, this article's proposed training involves 1D convolution for various layers, as shown in Figure 3.17. Below, the CNN is fully connected in a 1×1 feature map, allowing for the scalar form of weight and bias updates, which significantly affect CNN implementation.

Max pooling (2×2) is performed in the pooling layer. Several parameters must be considered, such as the filter dimensions, the number of filter dimensions, and the pooling process (as shown in Table 3.2). The filter dimensions are chosen to achieve maximum pooling so that the max-pooled output can be passed to the activation function. The remaining layers aim to minimize the error to the lowest possible value.

Dataset

Each image is a grayscale image where each pixel has a value between 0 and 255. The images of this dataset include variations such as brightness, facial expression (open and closed eyes, smile), images with/without glasses, scale, distance, and head angle rotation. One sample of pictures with different expressions is shown in Figure 3.18.

Initialization of Parameters

At first, 80% of data are considered training data, and 20% of data are considered testing data. Table 3.3 shows the initial values for the testing and training percentage, eye space, and the number of image channels.

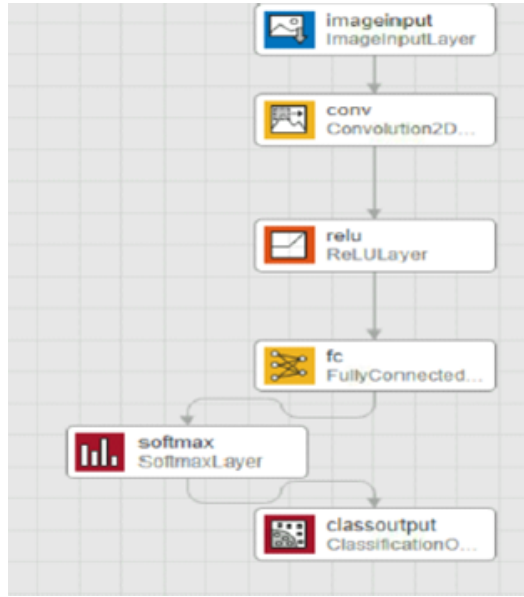


Figure 3.17: CNN Layer-By-Layer Sampling Overview

Table 3.2: Different Parameters In Our CNN

Analysis Results				
	Name	Type	Activations	Learnable
1	Image Input 128*128*3 images	Input Image	128*128*3	-
2	Conv 96 3*3*3 convolution	convolution	128*128*96	Weight 3*3*3*96 Bias 1*2*96 96 is number of filters
3	Relu	Relu	128*128*96 96 is number of filters	-
4	FC 10 fully connected	Fully connected	1*1*10	Weight 10*1572864 Bias 10*1
5	Softmax	Softmax	1*1*10	
6	Classoutput Crossentropyex	Classification output	-	

Table 3.3: Initial Values For The Parameters

Parameters	Initial values
Training data percentage	0.8
Testing data percentage	0.2
Eye space	70*270
The number of image channels	1

Different parameters, such as accuracy, will be computed based on the confusion matrix. Each eye accuracy is computed, and the mean is taken as the last accuracy. The last accuracy and the Error rate parameters can be computed depending on true and false predictions. The improvement process of neural network training with 60 repetitions is shown in Figure 3.19.



Figure 3.18: One Sample of ORL Dataset With Different Expressions. Source: University Of Cambridge.

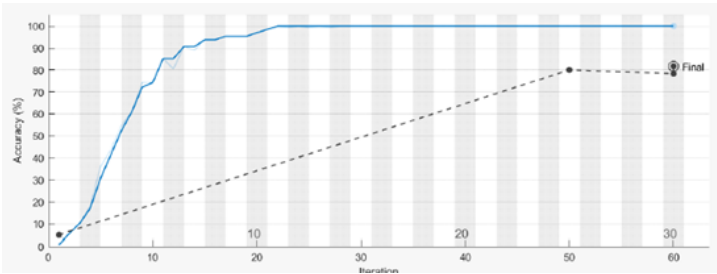


Figure 3.19: Improvement Process Of The Neural Network Training

Table 3.4 shows the comparison results of the proposed method.

Table 3.4: The Results Of Eye Recognition For The Images From Database

Proposed method		Previous papers	
Error Rate	Accuracy	Error Rate	Accuracy
18.30%	81.67%	21.00%	79.00%

Recognition was becoming the main problem for various applications. Eyes can be segmented after removing other parts such as skin, background, and the different parts of the face in an image. Several techniques applied with various studies such as template matching, Eigenvectors & Hough Transform. In the proposed method, eyes will be specified in an input image with a CNN neural network, which will be used in security systems to identify persons. To compare localization performance with the eye detector, the experiment is performed over 370 images in a dataset of ORL (Kleijnung et al., 2010), the rate of recognition is 81.67%, and the error is 18.33%. However, the proposed framework obtains the highest accuracy for the database, still this suffers from common biometric identification bottleneck that contains the proper identification failure when external visual attributes of eyes or faces change that will affect the identification of persons in the security systems. In future work, I will address the problems above.

Zernike Moments (ZM)

After extracting the features from the iris (eight features), detecting only eight features is to draw the interpolation curve of B-spline or Bézier, which requires at least four points of x and y coordinates. Therefore, I now reorder the first four features as x coordinates and the remaining features as y coordinates. So, the control points are (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , (x_4, y_4) . These points will be enrolled in the system's database with the user's information, such as ID, and will make the authentication process by matching the new four points with the points stored in the system's database. Algorithm 6 illustrates the process of authentication.

Algorithm 6 Biometric Authentication Process

```

1: Input: per // as a biometric person
2: Output: Authorization decision
3: Step One: Enrollment
4:   System checker  $\leftarrow$  per.ID
5: for  $i = 0; \dots; i++$  do
6:   Template[ $i$ ]  $\leftarrow$  Extract eight features from per by Zernike moments al-
     gorithm and reorder these features to obtain four control points.
7:   System-Database  $\leftarrow$  Template[ $i$ ]
8: Step Two: Create personal feature template for someone (per) who claims
     that he/she is q:
9:   per.tmp  $\leftarrow$  Extract features of per // send per's features template to con-
     gruence
10: if per.tmp == Template[ $i$ ] then
11:   per is q
12: else
13:   per does not exist

```

3.2.5 Encryption Stage

The key generation for encryption and decryption is the soul of any encryption algorithm and is the most critical issue when dealing with cryptography. Any encryption algorithm's importance lies in obtaining a strong key. Therefore, the main goal of this dissertation is to get proper encryption keys for the AES algorithm that will be used to encrypt the data and apply the RSA algorithm to protect the AES encryption key by encrypting this key using the modified RSA algorithm. Since choosing large prime numbers p and q for the RSA algorithm generates a strong encryption key, it will produce the best level of security in the system.

The control points extracted from the steps above will be used to generate the encryption keys. In the proposed system, two techniques are used to protect the data before sending it to cloud computing: symmetric and asymmetric techniques. Therefore, the proposed system can be divided into two parts. The first part generates 128 bits as the symmetric key for AES-128 to encrypt the data before storing it in cloud computing. The second part generates the asymmetric key used in the RSA algorithm to protect the symmetric key used to encrypt the data. The process of generating the encryption keys is described in the next section.

Generate the AES-128 Encryption Key

In this step, the system will generate the symmetric key, which will be used in the AES-128 algorithm to encrypt the data before storing it in cloud computing. To generate the AES encryption key, the control points will be used to draw the curve of the B-spline, which is designed from the above steps: (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , (x_4, y_4) . The x and y coordinates will be taken from each point, and the last eight digits from each coordinate will be extracted. The reason for detecting only eight digits is to ensure the generation of 128 bits. After that, each pair of digits will be converted to binary, and 128 bits will be obtained from these binary values. These 128 bits will represent the AES encryption key, which will encrypt the data before storing it in cloud computing. Algorithm 7 illustrates the generation of the AES encryption key.

Algorithm 7 Generate The AES Encryption Key

```

1: Input:  $(x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4)$  ▷ Control points
2: Output: AESKEY[i] ▷ AES Encryption Key
3: Step One: Detect the last eight digits from each control point, which result
   in one integer number  $A(i)$ .
4: for  $j = 1$  to 8 do
5:    $as \leftarrow \text{length}(x(j))$  ▷ as represents the length of  $x$ 
6:    $bb \leftarrow as - 6$  ▷ Determine the last eight order
7:   for  $i = bb$  to  $as$  do
8:      $A(i) \leftarrow x(i)$  ▷ Store last eight digits in  $A(i)$ 
9:  $B \leftarrow$  collecting  $A(i)$ s
10: Step Two: From  $B$ , convert each two digits into binary form for each control
    point.
11:  $K \leftarrow 1, ii \leftarrow 1$ 
12: while not end of  $B$  do
13:    $Totaldig(ii) \leftarrow [B(k), B(k+1)]$  ▷ Take only two digits
14:    $k \leftarrow k + 2$ 
15:    $aa_1(ii) \leftarrow \text{decimalToBinaryVector}(Totaldig(ii))$  ▷ Convert two digits to
    binary form
16:    $ii \leftarrow ii + 1$ 
17:    $Totalbin(ii) \leftarrow [aa_1(ii)]$ 
18: Step Three: Detect only 128 bits from Step Two and convert each 8 bits to
    hexadecimal to obtain AESKEY, representing the AES encryption key.
  
```

Generate the RSA Encryption Key

In this step, the system generates the asymmetric encryption key, which will be used in the RSA algorithm to encrypt the symmetric key generated in the previous step and send it securely. To generate the RSA encryption key, two large prime numbers p and q must be used. The Bézier system uses the proposed curve to obtain the first prime number p . The points required for Bézier curve interpolation are needed to draw the Bézier curve, as discussed in Section 2.13. Therefore, the control points extracted in the previous steps will represent the control points of the Bézier curve used to draw the curve containing thousands of points. From these points, only the prime numbers will be selected. Algorithm 8 illustrates how the Bézier curve works.

Algorithm 8 Bézier Curve

-
- 1: **Input:** $n + 1$ control points $(x_0, y_0), (x_1, y_1), \dots, (x_n, y_n)$
 - 2: **Output:** Bézier curve $B(t)$
 - 3: **Step One:** Compute Control Points for Bézier Curves
 - 4: For a Bézier curve, we typically use 4 control points, so $n = 3$. Let's denote these control points as $P_0 = (x_0, y_0)$, $P_1 = (x_1, y_1)$, $P_2 = (x_2, y_2)$ and $P_3 = (x_3, y_3)$.
 - 5: **Step Two:** Evaluate the Bernstein Polynomials
 - 6: The Bernstein polynomials for a Bézier curve are:

$$B_{0,3}(t) = (1 - t)^3$$

$$B_{1,3}(t) = 3t(1 - t)^2$$

$$B_{2,3}(t) = 3t^2(1 - t)$$

$$B_{3,3}(t) = t^3$$

- 7: **Step Three:** Compute the Bézier Curve Points
- 8: The Bézier curve is defined by parameter t having a range from 0 to 1.
- 9: **Step Four:** Algorithm Steps Using Given Variables
- 10: **1. Initialization:**
- 11: Given control points: $(x_0, y_0), (x_1, y_1), (x_2, y_2), (x_3, y_3)$
- 12: **2. For each parameter t in $[0, 1]$, compute the Bézier point $B(t)$:**

$$B(t) = (x(t), y(t))$$

$$x(t) = (1 - t)^3 x_0 + 3t(1 - t)^2 x_1 + 3t^2(1 - t)x_2 + t^3 x_3$$

$$y(t) = (1 - t)^3 y_0 + 3t(1 - t)^2 y_1 + 3t^2(1 - t)y_2 + t^3 y_3$$

After detecting the prime numbers, the system will select the nearest more significant two prime numbers from x and y , and from these two primes, choose the larger one. The second prime number q will be selected from the nearest Mersenne prime number compared with the first prime number, and this prime will be expanded by using $M_q = 2^q - 1$ when q is a prime number. M_p represents the Mersenne prime number used in the RSA algorithm. Algorithm 9 illustrates selecting the prime number from the Bézier curve points p .

After obtaining the largest prime numbers, we can now use the RSA algorithm to encrypt the AES encryption key to enhance the level of security in cloud computing. Figure 3.20 illustrates the flowchart for implementing the MRSA algorithm.

The strength of the RSA algorithm depends on the magnitude of p and q as prime numbers. The proposed approach allows us to obtain huge prime numbers and produces a very large private key. This key may be accessed through digits, providing a high level of security to protect the encryption key used to encrypt the data before storing it in cloud computing.

Algorithm 9 Determine The Largest Primes As p And q For RSA Algorithm

```

1: Input:  $x_p$  // represents the x coordinate points
2: Input:  $y_p$  // represents the y coordinate points
3: Output:  $p$  and  $q$  as large prime numbers
4: Step One: Determine the primes from the B-spline curve:
5: for  $i = 1$  to  $n$  do //  $n$  represents the number of  $x$  and  $y$  coordinates
6:   if  $\text{isprime}(x_p(i)) = 1$  then
7:      $prx(i) \leftarrow xp(i)$ 
8:   if  $\text{isprime}(y_p(i)) = 1$  then
9:      $pry(i) \leftarrow yp(i)$ 
10: Step Two: Selecting the prime number
11:  $d \leftarrow 1,000,000,000$  // initial  $d$  as a very large number
12: for  $k = 1$  to  $\text{length}(prx)$  do
13:   for  $j = 1$  to  $\text{length}(pry)$  do
14:      $d_0 \leftarrow |prx(k) - pry(j)|$ 
15:     if  $d_0 < d$  then
16:        $d \leftarrow d_0$ 
17:        $p_0 \leftarrow prx(k)$ 
18:        $p_1 \leftarrow pry(j)$ 
19: if  $p_0 > p_1$  then
20:    $p \leftarrow p_0$ 
21: else
22:    $p \leftarrow p_1$ 
23: Step Three: Selecting the nearest prime from Mersenne prime numbers to
    obtain the large prime number used in the RSA algorithm.
24:  $Mq \leftarrow 2^q - 1$ 

```

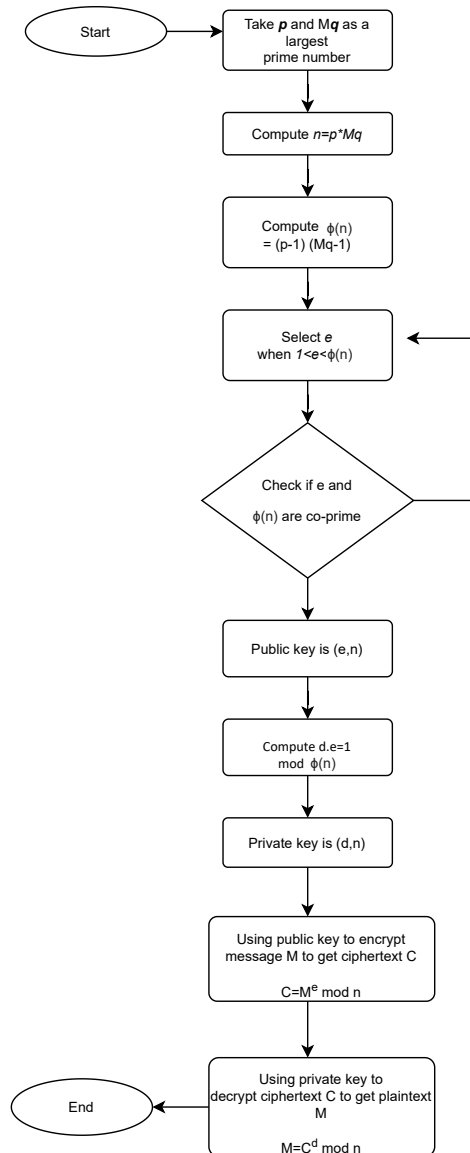


Figure 3.20: Flowchart To Compute Zernike Basis Functions

Chapter 4

Performance and Evaluation of the Proposed System

The contents of this chapter are published in article P6 and presented at conference C1.

This chapter evaluates and explains the proposed system implementation, which is divided into two parts.

4.1 Introduction

The first part is the authentication part that contains many steps, which are pre-processing the iris, extracting the eight features, and reordering these features to represent them as control points for B-spline or Bézier curves, and then matching these features with the features that are stored in the database to make verification that will be illustrated in the following sections. The second part is also divided into two stages. The first stage is to generate the AES encryption key, which will be used to encrypt the data. The second stage generates the RSA encryption key to encrypt the AES encryption key. This system was applied to the MMU and CASIA iris database.

The system was tested using MATLAB 2015 on a laptop with a Core i5 3320M CPU (2.60 GHz), 6 GB of RAM, and a 64-bit operating system.

4.2 Description of Dataset

This section describes the main features of the dataset. The first dataset is the CASIA iris dataset, and the second is the MMU dataset. These databases of iris images were captured in one acquisition session with

1. Flat optical sensor.
2. Sweeping thermal sensor.
3. Flat capacitive sensor.

4.3 Implementation of the Proposed System

The proposed system depends on the iris biometric to enhance the security in cloud computing. Therefore, in the beginning, there should be preprocessing of the iris by determining the iris and pupil localization, then the partial iris pattern, and then the iris divided into 3×3 subregions where the pupil is centered. After enhancing the iris, the proposed system consists of two stages: the first stage is the authentication stage, and the second stage is the confidentiality stage. These stages will be illustrated in the study cases presented in the next section.

Preprocessed Technique of Iris Preprocessing

This section demonstrates the main steps of the proposed iris preprocessing. This technique is the first step of the proposed system. As mentioned earlier, it consists of many steps. The first step of the preprocessing is to process the iris by determining the iris and pupil localization. The next step to preprocessing the iris image is to determine the partial iris pattern, and then the iris is divided into 3×3 sub-regions where the pupil is centered. The cases below illustrate these processes in detail.

Case study 1

The first case where the proposed system will be applied is an iris image from the MMU iris database.

1. Iris size.
2. Type of file.
3. AES Encryption Key, size (128 bits) to encrypt the data.
4. RSA Encryption Key, size (variable) to encrypt the AES Encryption Key.

Preprocessing the Eyes The first step after entering the iris image into the proposed system is to preprocess the iris by determining the iris and pupil localization, then localize the partial iris pattern, and finally divide the iris into 3×3 subregions where the pupil is centered. Figure 4.1a illustrates the iris and pupil localization. Figure 4.1b illustrates the partial iris pattern determination process. The final step is dividing the iris into 3×3 subregions where the pupil is centered; Figure 4.1c illustrates this step.

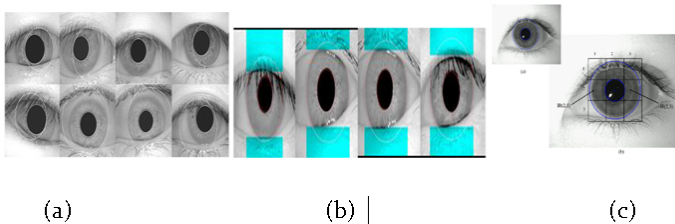


Figure 4.1: Preprocessing Of Iris (a) Iris And Pupil Localization (b) The Partial Iris Pattern (c) The Iris Region Is Divided Into Nine Subregions

Extract the Features After preprocessing the iris, the system can extract features from this enhanced image by applying Zernike moments. Zernike moments can detect the fine details of objects in the image.

However, they are susceptible to noise. On the other hand, they are a very efficient image feature extraction method and are invariant to image rotation. Therefore, to obtain eight features using the Zernike moment, it is necessary to detect the order (p) and the repetition (q) to get the set of polynomials. Experience shows that $p = 3$ and $q = 5$ give the best features for the proposed system. Table 4.1 illustrates this set of polynomials.

Table 4.1: List Of The First 8 Orders Of Zernike Moments

n	m	Zernike moments
1	1	$A_{1,1}$
2	0	$A_{2,0}$
2	2	$A_{2,2}$
3	1	$A_{3,1}$
3	3	$A_{3,3}$
4	0	$A_{4,0}$
4	2	$A_{4,2}$
4	4	$A_{4,4}$

This set of polynomials gives the eight features, which will be reordered to get the four points representing the control points of the B-spline. Table 4.2 illustrates these features and control points.

Table 4.2: Eight Features And Control Points

Features	Value of Features	Control Points
Z_1	1.017072	(1.017072, 1.653543)
Z_2	1.653412	(1.653412, 2.393891)
Z_3	1.744543	(1.744543, 3.255656)
Z_4	1.847913	(1.847913, 8.708148)
Z_5	1.653543	-
Z_6	2.393891	-
Z_7	3.255656	-
Z_8	8.708148	-

Authentication Stage The proposed system will use the control points for authentication and confidentiality. Therefore, to process authentication, the control points are used to be congruent with the features in the database. Table 4.3 illustrates this congruence.

The authentication process is performed by matching the first control point of the iris features with the first control points for all of the iris features in the database. This process is repeated for the other control points to ensure the user's identity. Consequently, when the features are congruent with any database row, the system authorizes the user to use cloud computing services. This represents the first approach in this dissertation to enhance the level of security in cloud computing.

Table 4.3: Congruence Control Points With Database

1.017072	1.653412	1.744543	1.847913	1.653543	2.393891	3.255656	8.708148
12.59776	18.73098	19.56652	32.7656	32.76805	26.37178	65.11989	20.04399
11.8573	18.8453	20.6162	21.98433	20.61446	14.4648	27.03797	10.26192
14.7314	44.70006	55.3483	62.42363	14.73702	93.8137	34.9924	17.2721
31.31452	34.64823	48.8476	76.49455	34.64321	10.44282	37.3523	14.0025
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$

Table 4.3 illustrates that the above process is performed on the same iris features and extracts four values for the left side and four other values for the right side of the same iris using Zernike Moments (ZM).

Using ZM gives fractional numbers, representing the features for this part of the iris with defined parameters. These values are used eight times. The fractional values are as follows:

Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8
1.653412	1.744543	1.017072	1.847913	1.653543	2.393891	3.255656	8.708148

We multiply these fractional numbers by 100,000 to extend their values (magnitudes) and convert them into integer values, excluding the fractions. Each value is restricted to 4 digits:

Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8
1653	1744	1017	1847	1653	2393	3255	8708

After this, we arrange the first four numbers in ascending order to represent the x-coordinate axis, while the other four numbers are taken as the y-coordinate axis:

Z_1	Z_2	Z_3	Z_4	Z_5	Z_6	Z_7	Z_8
1017	1653	1744	1847	1653	2393	3255	8708

These four points, consisting of ordered pairs of numbers, will be used as control points to draw the curve of the B-spline and Bézier-curve interpolator methods. Each pair of these values represents data for control point $K(x_i, y_i)$ where $i = 1 : 4$, as shown in Table 4.4, on the curve to be approximated by the interpolators used with B-spline, specifically:

$$\begin{aligned}
& [(Z_1, Z_5), (Z_2, Z_6), (Z_3, Z_7), (Z_4, Z_8)] \\
& = [(1017, 1653), (1653, 2393), (1744, 3255), (1847, 8708)]
\end{aligned}$$

The result of applying the B-spline interpolator for $(F1, F2, F3, F4)$ is illustrated in Figure 4.2. Through this drawing, we can obtain the final values that will be used to extract the keys. As I mentioned earlier, using a predefined step, I generate the points of the curve one by one in the $[0, 1]$ closed interval starting at parameter 0. If none of the calculated points satisfy the requirements until the end of the interval, a different step is applied and the generation started again from 0 until I find a good starting point.

Therefore, after applying the authentication stage, the proposed system will take the original features to establish the confidentiality stage.

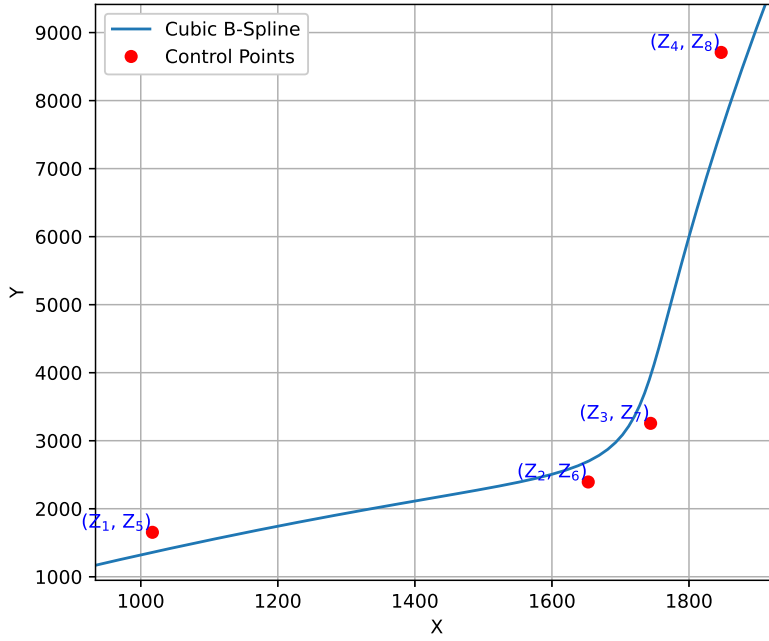


Figure 4.2: B-Spline Curve

Table 4.4: Select The Last Eight Digits To Obtain 128 Bits

Last eight digit of x	Binary Form
1017.19153781	00010011 00001111 00100101 01010001
1520.61634755	00111101 00111111 00101111 00110111
Last eight digit of y	Binary Form
1653.66624762	01000010 00111110 00101111 00111110
1922.78266287	01001110 00011010 00111110 01010111

Confidentiality Stage First Step The next stage is the confidentiality process, which consists of two steps. The first step is generating the encrypted key, which will be used to encrypt the data using the AES algorithm before storing it in the cloud computing data center. The second step is protecting the encrypted key that encrypts the data. This process can be done by encrypting the encryption key with the RSA algorithm.

From the table above, take only the first 128 bits and ignore the remaining bits. The result is:

```

00010011 00001111 00100101 01010001
01000010 00111110 00101111 00111110
00111101 00111111 00101111 00110111
01001110 00011010 00111110 01010111

```

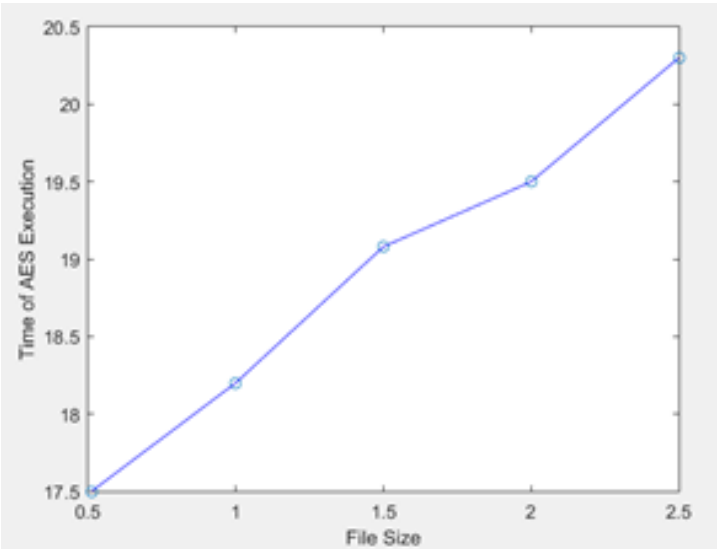


Figure 4.3: Period When Implemented the AES Algorithm

After this process, these bits are converted to hexadecimal by taking consecutive eight bits and converting them to hexadecimal, obtaining the symmetric key for the AES algorithm to encrypt the data. This process is illustrated in Table 4.5 before sending the data to cloud storage.

Table 4.5: 128-Bit Hexadecimal Encrypted Key

Hexadecimal Values			
13	F	25	51
22	3E	2F	3E
3D	3F	2F	37
4E	1A	3E	57

After generating the encrypted key from iris, this key represents a unique encryption key for every person. The key generated by this method will be used to encrypt the data before sending it to cloud computing. This is because this process represents the second approach in this dissertation to enhance the level of security in cloud computing. Table 4.6 and Figure 4.2 illustrate the time for encryption for the different sizes of files.

Secnd step

Table 4.6: Implementation Of AES Algorithm

File Size (Megabit)	Time of AES Execution (ms)
0.512	17.5
1	18.2
1.5	19.08
2	19.5
2.5	20.3

This step will protect the AES-encrypted key by utilizing the points as control points extracted by Zernike moments from the iris. In addition, this key is encrypted using the MRSA algorithm, which requires two prime numbers. The first prime number is obtained from the Bézier curve, while the second is derived from Mersenne prime numbers.

Before using the control points in the Bézier curve, the proposed system multiplies and rounds off each coordinate of x and y by a constant value, such as 10,000, to expand the value of each point. Table 4.7 illustrates this process.

Table 4.7: Expand The Value Of Control Points

X Coordinate of Control Points	Y Coordinate of Control Points
1017	1653
1653	2393
1744	3255
1847	8708

The mechanism for selecting the first prime number p involves comparing the prime numbers from the x -values and y -values, selecting the two most significant values, and then choosing the larger one to represent the prime number p . The second prime number is selected from the Mersenne prime numbers by comparing the value of p with the nearest more significant Mersenne prime number, representing the second prime number M_q . This prime number is used in the RSA algorithm to encrypt the encryption key employed in the AES algorithm.

The prime number p , selected from the points of the Bézier curve in this case, is 1847, and the nearest more significant Mersenne prime number is 8707. This number produces the second prime number by applying the Mersenne prime formula:

$$M_q = 2^{8707} - 1 = (1.169963321862486603522673270657 \times 10^{2621}) - 1$$

The length of the private key is 2622 digits.

These prime numbers (p and M_q) are used in the RSA algorithm to generate the public key and private key, which are then utilized to protect the encrypted key of the AES algorithm. In general, the relationship between the length of the private key and the time required to generate this key is illustrated in Figure 4.4.

Case study 2

The second case applies the proposed system to an iris image from the CASIA iris database.

1. Iris size (Kbytes).
2. Type of file.
3. AES Encryption Key, size (128 bits) to encrypt the data.
4. RSA Encryption Key, size (variable) to encrypt the AES Encryption Key.

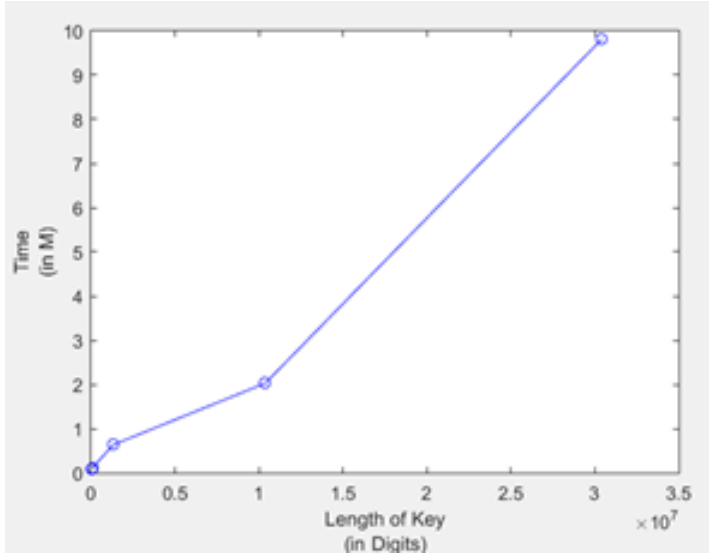


Figure 4.4: Relation Between the Length Of Private Key And Time

Preprocessing the Eyes The first step in the proposed system for preprocessing is to ensure that the iris image is converted into a square format:

$$x = x_1(1 : N, 1 : N)$$

Where:

- x_1 represents the original iris image.
- N is the minimum dimension of the iris image.

After this step, the proposed system preprocesses the iris. Figure 4.5 illustrates the preprocessing of the iris.

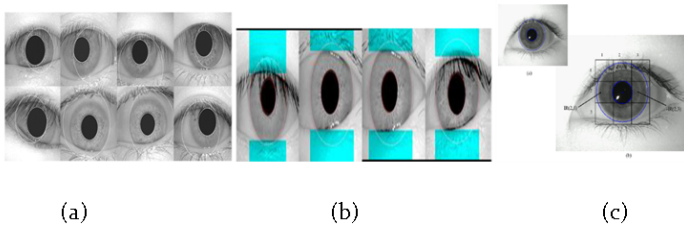


Figure 4.5: Preprocessing Of Iris (a) Iris And Pupil Localization (b) The Partial Iris Pattern (c) The Iris Region Is Divided Into Nine Subregions

Extract the Features The proposed system extracts eight features from the iris using Zernike moments and reorders them to obtain four control points for drawing the B-spline curve. Table 4.8 illustrates these features and their corresponding control points.

Table 4.8: Eight Features And Control Points For Case Study 2

Feature	Value of Feature	Control Points
Z_1	1113.336	(1113.336, 1113.957)
Z_2	1590.908	(1590.908, 4653.255)
Z_3	2162.809	(2162.809, 1326.555)
Z_4	9923.468	(9923.468, 2335.987)
Z_5	1113.957	-
Z_6	4653.255	-
Z_7	1326.555	-
Z_8	2335.987	-

Authentication Stage The control points are utilized for authentication to ensure congruence with the stored features in the database. Table 4.9 illustrates this congruence.

Table 4.9: Congruence Of Control Points With Database

1798.324	1803.471	2783.828	9560.785	2783.6892	2598.767	1277.892	2082.352
2703.776	5639.068	7221.632	7757.561	5639.852	1082.078	1996.989	4622.349
2776.762	2493.503	6756.662	6332.435	6217.465	2493.473	2492.797	1759.122
1597.843	2338.062	3203.731	4581.633	4581.772	6399.361	5704.294	1736.133
1113.336	1590.908	2162.809	9923.468	1113.957	4653.255	1326.555	2335.987
...	...	...	...	...	...	...	...

Confidentiality Stage To generate the encryption key used in the AES algorithm to encrypt the data before storing it in cloud computing, take the last eight digits from each point of control point and convert every two digits into binary form. Then, it selects only 128 bits as an encryption key. Table 4.10 illustrates the result of this process.

Table 4.10: Select The Last Eight Digits To Obtain 128 Bits For Case Study 2

Last eight digit of x	Binary Form
9892,76853264	01001100 01010101 00100000 01000000
9920,67191566	01000011 00010011 00001111 01000010
Last eight digit of y	Binary Form
2331,08685890	00001000 01000100 00111010 01011010
2334,69743006	01000101 01001010 00011110 00000110

01001100	01010101	00100000	01000000
01000011	00010011	00001111	01000010
00001000	01000100	00111010	01011010
01000101	01001010	00011110	00000110

The final step to obtain the AES-128 encryption key is to convert these 128 bits to hexadecimal, which will be used as an encryption key for the AES algorithm to encrypt the data. This process is illustrated in Table 4.11 before sending the data to cloud storage.

Table 4.11: 128-Bit Hexadecimal Encrypted Key
Hexadecimal Values

4C	55	20	40
43	13	F0	42
08	44	3A	5A
45	4A	1E	06

The second step in the confidentiality of the proposed system is to protect the encryption key by encrypting it with the MRSA algorithm. Therefore, two prime numbers must be obtained initially to apply the MRSA algorithm. The first prime number could be obtained from using the Bézier curve by utilizing the control points to draw the curve of a Bézier curve.

The first prime number can be obtained from the curve's points of the Bézier-curve ($p = 9923$). The second prime number can be obtained by comparing the first prime number with the nearest Mersenne prime number ($q = 4673$). We apply the Mersenne prime number to obtain a huge prime number.

$$M_q = 2^{9923} - 1 = (5.2548491497447161918815538985055 \times 10^{9907}) - 1$$

The length of the private key is 2988 digits.

4.4 Evaluate the Result of the Proposed System

The proposed system's strength must be evaluated in two parts. The first part measures the randomness of an encrypted key for AES-128. The second part improves the proposed system to generate a large prime number.

4.4.1 Case Study 1

The AES key encryption that was produced in Case Study 1 was

00010011	00001111	00100101	01010001
01000010	00111110	00101111	00111110
00111101	00111111	00101111	00110111
01001110	00011010	00111110	01010111

Frequency Test

The test statistic is given by:

$$X^2 = \frac{(n_0 - n_1)^2}{n}$$

Where the good sequence must satisfy:

$$0 < X^2 < 3.84$$

Where:

- n : total digits of the key sequence

- n_0 : number of zeros in the key sequence
- n_1 : number of ones in the key sequence

Given:

$$n = 128, \quad n_0 = 60, \quad n_1 = 68$$

We compute:

$$X^2 = \frac{(60 - 68)^2}{128} = \frac{64}{128} = 0.4923077$$

The key succeeds this test since $X^2 = 0.4923077 < 3.84$.

Frequency Mono Bit Test

The frequency mono-bit test is used to ensure the randomness of a sequence based on the proportion of 0's and 1's in the entire sequence. The number of zeros and ones expected in a random sequence should be the same. To succeed in this test, the P-value must be greater than 0.01. The steps of this test are as follows:

1. Convert the zeros into (-1) and the ones into $(+1)$ for the key sequence, and sum them to produce:

$$S_n = x_1 + x_2 + \cdots + x_n$$

where $S_n = 8$.

2. Compute:

$$S_{\text{obs}} = \frac{|S_n|}{\sqrt{n}} = \frac{8}{\sqrt{128}} = \frac{8}{11.31} = 0.70733$$

3. Compute the P-value:

$$P\text{-value} = \text{erfc} \left(\frac{S_{\text{obs}}}{\sqrt{S_n}} \right) = \text{erfc} \left(\frac{0.70733}{\sqrt{8}} \right) = 0.7257210 > 0.01$$

Since the P-value is more significant than 0.01, this key successfully passes the frequency monobit test.

Run Test

This test determines whether the number of 0s and 1s in the sequence is approximately the same as expected in a truly random sequence. When the P-value is greater than 0.01, the test is satisfied. The steps of this test are as follows:

1. Compute the pretest proportion:

$$\pi = \frac{\sum_j \zeta_j}{n} = \frac{60}{128} = 0.4687$$

2. Compute τ :

$$\tau = \frac{2}{\sqrt{n}} = \frac{2}{\sqrt{128}} = 0.1768$$

3. Check the condition:

$$|\pi - \frac{1}{2}| \leq \tau \quad \Rightarrow \quad -0.0313 \leq 0.1768$$

4. Compute $v_n(\text{obs})$:

$$v_n(\text{obs}) = \sum_{k=1}^{n-1} r(k) + 1,$$

where $r(k) = 0$ if $\zeta_k = \zeta_{k+1}$, $r(k) = 1$ otherwise

Thus, $v_n(\text{obs}) = 60$.

5. Compute the P-value:

$$P\text{-value} = \text{erfc} \left(\frac{|v_n(\text{obs}) - 2n\pi(1 - \pi)|}{2\sqrt{2n\pi(1 - \pi)}} \right)$$

$$P\text{-value} = \text{erfc} \left(\frac{|60 - 2 \times 128 \times 0.4687 \times (1 - 0.4687)|}{2\sqrt{2 \times 128 \times 0.4687 \times (1 - 0.4687)}} \right)$$

$$P\text{-value} = 0.449 > 0.01$$

This key successfully passes the run test since the P-value is more significant than 0.01.

Entropy

Entropy is an important parameter to test the strength of the encryption key. When measuring the entropy of this key using a MATLAB function, the value of entropy is:

$$\text{Entropy} = 9.971804 \times 10^{-1}$$

4.4.2 Case Study 2

The AES key encryption that was produced in Case Study 2 was

```
01001100  01010101  00100000  01000000
01000011  00010011  00001111  01000010
00001000  01000100  00111010  01011010
01000101  01001010  00011110  00000110
```

Frequency Test

We can calculate:

$$n = 128, \quad n_0 = 84, \quad n_1 = 44.$$

The test statistic is:

$$X^2 = \frac{(84 - 44)^2}{128}$$

$$X^2 = 12.5$$

This key fails the test since $X^2 = 12.5 > 3.84$.

Frequency Mono Bit Test

To perform the frequency mono-bit test:

1. Convert the zeros into (-1) and the ones into $(+1)$ for the key sequence, and sum them to produce:

$$S_n = x_1 + x_2 + \cdots + x_n$$

where $S_n = -40$.

2. Compute:

$$S_{\text{obs}} = \frac{|S_n|}{\sqrt{n}} = \frac{40}{\sqrt{128}} = 3.535$$

3. Compute the P-value:

$$P\text{-value} = \text{erfc} \left(\frac{S_{\text{obs}}}{\sqrt{S_n}} \right) = \text{erfc} \left(\frac{3.535}{\sqrt{-40}} \right)$$

This key fails the test because the test statistic is not valid due to the negative number inside the square root.

Run Test

This test aims to determine whether the number of 0s and 1s in the sequence is approximately the same as expected in a truly random sequence. The steps are as follows:

1. Compute the pre-test proportion:

$$\pi = \frac{\sum_j \zeta_j}{n} = \frac{84}{128} = 0.6562$$

2. Compute τ :

$$\tau = \frac{2}{\sqrt{n}} = \frac{2}{\sqrt{128}} = 0.1768$$

3. Check the condition:

$$\left| \pi - \frac{1}{2} \right| \leq \tau \quad \Rightarrow \quad 0.1562 \leq 0.1768$$

4. Compute $v_n(\text{obs})$:

$$v_n(\text{obs}) = \sum_{k=1}^{n-1} r(k) + 1, \quad \text{where } r(k) = 0 \text{ if } \zeta_k = \zeta_{k+1}, \quad r(k) = 1 \text{ otherwise}$$

Thus, $v_n(\text{obs}) = 63$.

5. Compute the P-value:

$$P\text{-value} = \text{erfc} \left(\frac{|v_n(\text{obs}) - 2n\pi(1 - \pi)|}{2\sqrt{2n\pi(1 - \pi)}} \right)$$

$$P\text{-value} = \text{erfc} \left(\frac{|63 - 2 \times 128 \times 0.6562 \times (1 - 0.6562)|}{2\sqrt{2 \times 128 \times 0.6562 \times (1 - 0.6562)}} \right)$$

$$P\text{-value} = 46.9 > 0.01$$

This key successfully passes the run test since the P-value is more significant than 0.01.

Entropy

Entropy is an important parameter to test the strength of the encryption key. When measuring the entropy of this key, the value of entropy is:

$$\text{Entropy} = 92.8$$

The summary of the randomness test is illustrated in Table 4.12.

Table 4.12: Summary Of Randomness Tests

Key in Hexadecimal Form	Frequency Test	Frequency Mono Test	Run Test	Entropy
130F2555523E3F3E3D3F2F334E1F3E57	2	0.617	1	0.988
332F2535563E2B2E353B2B3346563E57	0.5	0.723	0.118	0.997
272F255D5A1E2D3635372D334A1F365B	0.281	0.759	0.00893	0.998
180F255D523E3F363D3F27335E5E3E5F	3.125	0.576	0.591	0.982
4C55204043130F4208443A5A454A1E06	12.5	-	46.9	92.8

The above tests prove that the proposed system, which uses the control points of B-spline and Bézier curves, has generated strong encryption keys for the AES-128 algorithm. Therefore, this approach enhances the security of AES-128 used in cloud computing.

Chapter 5

Theses

This chapter presents the core contributions of this dissertation, which introduces a hybrid security framework for cloud computing that combines biometric authentication and advanced cryptographic techniques. The system utilizes iris recognition for feature extraction via Convolutional Neural Networks (CNNs) and Zernike moments and applies B-spline and Bézier curve interpolation to generate control points. These are then used to create secure encryption keys for symmetric (AES-128) and asymmetric (RSA) encryption.

5.1 Authentication and Biometric Contributions

1. Developed a novel cloud authentication framework based on iris biometrics, leveraging CNNs or compact control points derived from Zernike moments and B-spline interpolation [P2, C1]. This reduced computational overhead by storing only four control points per user (Section 3.2.4).
2. Enhanced iris feature extraction by employing rotation-invariant Zernike moments, improving robustness and accuracy in authentication scenarios [P4].
3. Applied B-spline and Bézier curve techniques to generate high-entropy, non-periodic control points, improving both the randomness and compactness of biometric representations [P6, C1].

5.2 Cryptographic Contributions

The techniques and findings presented in this section are thoroughly discussed in Section 3.2.5.

5.2.1 Biometric-Based AES Key Generation

1. Proposed a secure method for AES-128 key derivation from biometric data using B-spline and Bézier curve control points, ensuring key invariance under iris rotation due to Zernike moment properties [P1, P4].
2. Strengthened AES key security by validating the generated keys through statistical randomness and entropy analysis, confirming resistance to brute-force attacks [P6].
3. Secured AES key transmission by integrating RSA encryption, mitigating key exchange vulnerabilities [C1].

5.2.2 RSA Public Key Generation and Modified RSA (MRSA)

1. Designed a novel RSA key generation approach where the first prime (p) is derived from B-spline-based iris features, and the second (q) from a Mersenne prime formula, enabling strong and biometric-tied public keys (Section 3.2.5).
2. Introduced a Modified RSA (MRSA) scheme capable of handling extremely large primes (up to 2988 digits), enhancing asymmetric cryptographic strength (Section 3.2.5).
3. Demonstrated a comprehensive hybrid encryption model combining symmetric (AES-128) and asymmetric (RSA) cryptographic methods for secure cloud-based data transmission [C1].

5.3 Performance Overview

The proposed system demonstrates the ability to handle very large prime numbers, supporting key lengths of up to 2988 digits for secure encryption. A noted limitation is the increased execution time required to encrypt AES-128 keys using the RSA algorithm, primarily due to the high computational cost of large prime generation and exponentiation. Despite this, the trade-off between security and performance is justified by the significant increase in key strength and randomness achieved through the combined use of B-spline, Bézier curves, and Mersenne primes.

Chapter 6

Summary

Cloud internet technology has emerged to provide various electronic computing services. According to the researchers, there are basic risks anytime the users log onto a computer and access the internet, but these risks are more significant when computing in the cloud. Security algorithms for advanced encryption and decryption can be developed to enhance security over the cloud network.

Some cryptography techniques mainly depend on prime numbers generated using weak algorithms such as an RNG (Random Number Generator) and SRNG (Pseudorandom Number Generator). These algorithms generate random numbers, and then primes are chosen from these numbers after some tests.

The main goal of this dissertation is to enhance the two stages of information security in cloud computing and to improve confidentiality and authentication. Two processes can achieve confidentiality. First, we generate an AES encryption key from the iris biometric to encrypt the data. Second, I use a Bézier curve interpolation to reduce periodicity to generate a large prime number as a powerful key that will be used in the modified RSA algorithm called the Mersenne RSA (MRSA) algorithm to encrypt the key of the AES algorithm. From an authentication point of view, we can do that by using features of the iris that will be considered control points of Bézier curves as unique identifications of users.

The proposed approach extracts the features from the iris by applying Zernike moments as a good feature representation. These features are used as control points to draw the curve of a B-spline and to generate a 128-bit AES encryption key, which is used to encrypt the data before sending or storing it in the cloud. Also, these control points will be used in the B-spline interpolation to get the prime numbers that will be expanded using Mersenne prime numbers. Larger prime numbers are used to create the public and private keys in the modified RSA algorithm (MRSA). MRSA algorithm will be used to protect the AES private key.

In addition, these features are used as a tool for authentication, which is an additional step to enhance the security in the proposed system. Thus, this dissertation proposes two encryption techniques: asymmetric and symmetric keys with RSA and AES algorithms to enhance data security in cloud computing.

The proposed system can obtain the largest prime number, 29923 digits, and then convert each digit of the prime number coming from Mersenne prime numbers to binary of four bits. The q component of the final key will be 11.952. The final key, as a prime number, will be used to generate both the encryption and decryption keys in the modified RSA (MRSA) algorithm.

The proposed approach is robust against iris rotation and noise. The iris datasets used to evaluate the performance of the proposed approach are the Biometric Recognition MMUD dataset and the CASIA iris dataset.

Suggestions for Future Work

1. Implementing Self-Organizing Maps (SOM) for user identification in cloud computing.
2. Utilizing multi-biometrics for authentication and encryption key generation.
3. Selecting a more efficient classification method to reduce matching time.
4. Developing an optimized RSA variant to minimize time complexity.
5. Exploring alternative encryption algorithms such as elliptic curves to enhance data security.

List of publications

Conferences

- C1** Hussein Lafta Hussein, Ahmed Abdulrudah Abbass, **Jasim H. Lafta**, Robert Tornai: *Novel Method to Generate symmetric key from Iris Using Polynomial Interpolation and Zernike Moment*. The 12th International Conference on Applied Informatics (ICAI 2023), Eger, Hungary, March 2-4, 2023.

Publications

- P1** Abbass, A. A., Al, A. S., Hussein, H. L., **Kaabi, J. H. L.**, Tornai, R.: *American standard code for information interchange mapping technique for text hiding in the RGB and gray images*. *Int. J. Electr. Comp. Eng.*, **12**(3), 2812–2817, 2022. ISSN: 2088-8708.
DOI: <http://dx.doi.org/10.11591/ijece.v12i3.pp2812-2817>
- P2** Abbass, A. A., Hussein, H. L., Shukur, W. A., **Kaabi, J. H. L.**, Tornai, R.: *Efficient Eye Recognition for Secure Systems Using Convolutional Neural Network*. *Webology*, **19**(1), 4967–4978, 2022. ISSN: 1735-188X.
DOI: <http://dx.doi.org/10.14704/WEB/V19I1/WEB19333>
- P3** Naji, S. A., Hussein, H. L., Ihsan, A. M., **Kaabi, J. H. L.**: *Enhancing LSB Algorithm against Brute-Force Attack using ASCII Mapping Technique*. *J Eng Appl Sci*, **14**(16), 5854–5863, 2019. ISSN: 1816-949X.

Conference proceedings

- P4** Naji, S. A., Tornai, R., **Kaabi, J. H. L.**, Hussein, H. L.: *Iris Recognition Using Localized Zernike Features with Partial Iris Pattern*.
In: *New Trends in Information and Communications Technology Applications*. Eds.: Abbas M. Al-Bakry, Safaa O. Al-Mamory, Mouayad A. Sahib, Haitham S. Hasan, George S. Oreku, Thaker M. Nayl, Jaafar A. Al-Dhaibani. Springer, Baghdad, Iraq, pp. 219–232, 2020. (Proceedings, ISSN: 1865-0929). ISBN: 9783030553395
- P5** Hussein, H. L., Abbass, A. A., Naji, S. A., Al, A. S., **Kaabi, J. H. L.**: *Hiding text in gray image using mapping technique*.
J. Phys. Conf. Ser., **1003**, 1–7, 2018. ISSN: 1742-6588.
DOI: <http://dx.doi.org/10.1088/1742-6596/1003/1/012032>
- P6** Hussein, H. L., Abbass, A. A., Naji, S. A., **Lafta, J. H.**, & Tornai, R. (2025). *Extract features using Zernike Moment to generate symmetric encryption key with polynomial interpolations*. In *AIP Conference Proceedings* (Vol. 3264, p. 030028). 3rd International Conference on Mathematics, AI, Information and Communication Technologies: ICMAICT2023. AIP Publishing. <https://doi.org/10.1063/5.0258437>



Registry number: DEENK//2025.PL
Subject: PhD Publication List

Candidate: Jasim Hussein Lafta Kaabi
Doctoral School: Doctoral School of Informatics
MTMT ID: 10071757

List of publications related to the dissertation

Foreign language scientific articles in international journals (3)

1. Abbass, A. A., Al, a. S., Hussein, H. L., **Kaabi, J. H. L.**, Tornai, R.: American standard code for information interchange mapping technique for text hiding in the RGB and gray images.
Int. J. Electr. Comp. Eng. 12 (3), 2812-2817, 2022. ISSN: 2088-8708.
DOI: <http://dx.doi.org/10.11591/ijece.v12i3.pp2812-2817>
2. Abbass, A. A., Hussein, H. L., Shukur, W. A., **Kaabi, J. H. L.**, Tornai, R.: Efficient Eye Recognition for Secure Systems Using Convolutional Neural Network.
Webology. 19 (1), 4967-4978, 2022. ISSN: 1735-188X.
DOI: <http://dx.doi.org/10.14704/WEB/V19I1/WEB19333>
3. Naji, S. A., Hussein, H. L., Ihsan, A. M., **Kaabi, J. H. L.**: Enhancing LSB Algorithm against Brute-Force Attack using ASCII Mapping Technique.
J Eng Appl Sci. 14 (16), 5854-5863, 2019. ISSN: 1816-949X.

Foreign language conference proceedings (3)

4. Hussein, H. L., Abbass, A. A., Naji, S. A., **Kaabi, J. H. L.**, Tornai, R.: Extract features using Zernike Moment to generate symmetric encryption key with polynomial interpolations.
In: 3rd International Conference on Mathematics, AI, Information and Communication Technologies: ICMAICT2023, AIP Publishing, Melville, NY, 030028-1-030028-9, 2025, (AIP Conference Proceedings, ISSN 0094-243X 1551-7616 ; 3264)
5. Naji, S. A., Tornai, R., **Kaabi, J. H. L.**, Hussein, H. L.: Iris Recognition Using Localized Zernike Features with Partial Iris Pattern.
In: New Trends in Information and Communications Technology Applications. Ed.: Abbas M. Al-Bakry, Safaa O. Al-Mamory, Mouayad A. Sahib, Haitham S. Hasan, George S. Oreku, Thaker M. Nayl, Jaafar A. Al-Dhaibani, Springer, Baghdad, Iraq, 219-232, 2020, (Proceedings, ISSN 1865-0929) ISBN: 9783030553395



UNIVERSITY of
DEBRECEN

UNIVERSITY AND NATIONAL LIBRARY
UNIVERSITY OF DEBRECEN

H-4002 Egyetem tér 1, Debrecen
Phone: +3652/410-443, email: publikaciok@lib.unideb.hu

6. Hussein, H. L., Abbass, A. A., Naji, S. A., Al, a. S., **Kaabi, J. H. L.**: Hiding text in gray image using mapping technique.

J. Phys. Conf. Ser. 1003, 1-7, 2018. ISSN: 1742-6588.

DOI: <http://dx.doi.org/10.1088/1742-6596/1003/1/012032>

The Candidate's publication data submitted to the Tudóstér have been validated by DEENK on the basis of the Journal Citation Report (Impact Factor) database.

09 April, 2025

Bibliography

- A, D. L., Saleem, D. M., Swar, B. K., Kumar, S. and Sharma, R. (2022). Cloud computing: The emerging technology. *International Research Journal of Modernization in Engineering, Technology and Science*, 4, 1503. Peer-Reviewed, Open Access, Fully Refereed International Journal. Impact Factor: 6.752, URL <https://www.irjmets.com>.
- Aggarwal, C. C. (2018a). *Neural networks and deep learning*. 1st ed. Springer International Publishing, Basel, Switzerland.
- Aggarwal, C. C. (2018b). *Neural Networks and Deep Learning: A Textbook*. Springer International Publishing. URL <http://dx.doi.org/10.1007/978-3-319-94463-0>.
- Ali, b. H., Mohammedali, M. A., Abdul-Rahaim, L. A. and Al-Kharsan, I. H. (2022). Design of surgical arm robot based on cloud computing. In *2022 5th International Conference on Engineering Technology and its Applications (IICETA)*. IEEE. URL <http://dx.doi.org/10.1109/IICETA54559.2022.9888457>.
- Alliance, C. S. (2012). SecaaS Implementation Guidance, Category 8: Encryption. Tech. rep., Cloud Security Alliance. URL https://downloads.cloudsecurityalliance.org/initiatives/secaas/SecaaS_Cat_8_Encryption_Implementation_Guidance.pdf.
- Alnajdi, S., Dogan, M. and Al-Qahtani, E. (2016). A survey on resource allocation in cloud computing. *International Journal on Cloud Computing: Services and Architecture (IJCCSA)*, 6,.
- Amazon Web Services (2009). Amazon ec2 - amazon elastic compute cloud. URL <http://www.amazon.com/ec2>.
- Ambedkar, B. and Bedi, S. (2011). A new factorization method to factorize rsa public key encryption. *International Journal of Computer Science Issues (IJCSI)*, 8, 242.
- Asha, N. and Rao, G. R. (2013). A review on various resource allocation strategies in cloud computing. *International Journal of Emerging Technology and Advanced Engineering (IJETA)*, 3,.
- Ashalatha, R. and Vaidehi, M. (2012). The significance of data security in cloud: A survey on challenges and solutions on data security. *International Journal of Internet Computing*, 1, 15–18.

- Bakštein, E. (2014). Iris recognition. URL <http://www.mlmU.cz/wp-content/uploads/2014/09/Iris-MLMU.pdf>.
- Balakumar, P. and Venkatesan, R. (2011). Secure biometric key generation scheme for cryptography using combined biometric features of fingerprint and iris. *International Journal of computer science Issues (IJCSI)*, 8, 349.
- Barker, E. B., Barker, W. C., Burr, W. E., Polk, W. T. and Smid, M. E. (2012). *Recommendation for key management, part 1 :: general (revision 3)*. URL <http://dx.doi.org/10.6028/NIST.SP.800-57p1r3>.
- Bassham, L. E., Rukhin, A. L., Soto, J., Nechvatal, J. R., Smid, M. E., Barker, E. B., Leigh, S. D., Levenson, M., Vangel, M., Banks, D. L., Heckert, N. A., Dray, J. F. and Vo, S. (2010). *A statistical test suite for random and pseudorandom number generators for cryptographic applications*. URL <http://dx.doi.org/10.6028/NIST.SP.800-22r1a>.
- Bidgoli, H. (2006). *Handbook of Information Security, Threats, Vulnerabilities, Prevention, Detection, and Management*, vol. 3. John Wiley & Sons.
- Bowyer, K. W., Hollingsworth, K. and Flynn, P. J. (2008). Image understanding for iris biometrics: A survey. *Computer vision and image understanding*, 110, 281–307.
- Chafia, F., Salim, C. and Farid, B. (2010). A biometric crypto-system for authentication. In *2010 International Conference on Machine and Web Intelligence*. IEEE, 434–438.
- Chandra, S., Paul, S., Saha, B. and Mitra, S. (2013). Generate an encryption key by using biometric cryptosystems to secure transferring of data over a network. *IOSR Journal of Computer Engineering (IOSR-JCE)*, 12, 16–22.
- Chen, Y., Paxson, V. and Katz, R. H. (2010). What’s new about cloud computing security? Tech. Rep. UCB/EECS-2010-5. URL <http://www2.eecs.berkeley.edu/Pubs/TechRpts/2010/EECS-2010-5.html>.
- Chong, C.-W., Raveendran, P. and Mukundan, R. (2003). Translation invariants of zernike moments. *Pattern Recognition*, 36, 1765–1773. URL [http://dx.doi.org/10.1016/S0031-3203\(02\)00353-9](http://dx.doi.org/10.1016/S0031-3203(02)00353-9).
- Daugman, J. (1998). Recognizing people by their iris patterns. *Information Security Technical Report*, 3, 33–39. URL [http://dx.doi.org/10.1016/S1363-4127\(98\)80016-2](http://dx.doi.org/10.1016/S1363-4127(98)80016-2).
- Dillon, T., Wu, C. and Chang, E. (2010). Cloud computing: Issues and challenges. In *2010 24th IEEE International Conference on Advanced Information Networking and Applications*. 27–33.
- Feng, Y., Yuen, P. C. and Jain, A. K. (2008). A hybrid approach for face template protection. In *Biometric Technology for Human Identification V*, vol. 6944. SPIE, 58–68.
- Gampala, V., Inuganti, S. and Muppidi, S. (2012). Data security in cloud computing with elliptic curve cryptography. *International Journal of Engineering and Advanced Technology (IJEAT)*, 2, 138–141. Retrieval Number: C0721052312.

- Goldreich, O. et al. (2005). Foundations of cryptography—a primer. *Foundations and Trends® in Theoretical Computer Science*, 1, 1–116.
- Goshwe, N. Y. (2013). Data encryption and decryption using rsa algorithm in a network environment. *International Journal of Computer Science and Network Security (IJCSNS)*, 13, 9.
- Greengard, S. (2010). Cloud computing and developing nations. *Commun. ACM*, 53, 18–20. URL <https://doi.org/10.1145/1735223.1735232>.
- He, K., Zhang, X., Ren, S. and Sun, J. (2016). Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.
- Heaton, J. (2017). Ian goodfellow, yoshua bengio, and aaron courville: Deep learning: The mit press, 2016, 800 pp, isbn: 0262035618. *Genetic Programming and Evolvable Machines*, 19, 305–307. URL <http://dx.doi.org/10.1007/s10710-017-9314-z>.
- Hse, H. and Newton, A. (2004). Sketched symbol recognition using zernike moments. In *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.*, vol. 1. 367–370 Vol.1.
- Ilankumaran, S. and Deisy, C. (2019). Multi-biometric authentication system using finger vein and iris in cloud computing. *Cluster Computing*, 22, 103–117.
- Jagadeesan, A., Thillaikkarasi, T. and Duraiswamy, K. (2010). Cryptographic key generation from multiple biometric modalities: Fusing minutiae with iris feature. *International Journal of Computer Applications*, 2, 16–26.
- Jain, A. K., Bolle, R. and Pankanti, S. (2006). *Biometrics: personal identification in networked society*, vol. 479. Springer Science & Business Media.
- Jain, A. K., Ross, A. and Prabhakar, S. (2004). An introduction to biometric recognition. *IEEE Transactions on circuits and systems for video technology*, 14, 4–20.
- Jouini, M., Aissa, A. B., Rabai, L. B. A. and Mili, A. (2012). Towards quantitative measures of information security: A cloud computing case study. *International Journal of cyber-security and digital forensics (IJCSDF)*, 1, 248–262.
- Kalpana, P. and Singaraju, S. (2012). Data security in cloud computing using rsa algorithm. *International Journal of research in computer and communication technology, IJRCCT, ISSN 2278–5841*.
- Kaur, A. and Singh, S. (2013). Improved storage security scheme usingrsa&twofish algorithm at window azure cloud. *International Journal of Computer Trends and Technology (IJCTT)–volume*, 4,.
- Kaur, B., Singh, S. and Kumar, J. (2018). Iris recognition using zernike moments and polar harmonic transforms. *Arabian Journal for Science and Engineering*, 43, 7209–7218. URL <http://dx.doi.org/10.1007/s13369-017-3057-2>.
- Kaur, S. (2012). Cryptography and encryption in cloud computing. *VSRD International journal of Computer science and Information Technology*, 2, 242–249.

- Khatoon, A. and Ikram, D. A. A. (2014). Performance Evaluation of RSA Algorithm in Cloud Computing Security. *International Journal of Innovation and Scientific Research*, 12, 336–345. URL <http://www.ijisr.issr-journals.org/abstract.php?article=IJISR-14-302-08>.
- Khotanzad, A. and Hong, Y. (1990). Invariant image recognition by zernike moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12, 489–497.
- Kleijnung, T., Aoki, K., Franke, J., Lenstra, A. K., Thomé, E., Bos, J. W., Gaudry, P., Kruppa, A., Montgomery, P. L., Osvik, D. A. et al. (2010). Factorization of a 768-bit rsa modulus. In *Advances in Cryptology–CRYPTO 2010: 30th Annual Cryptology Conference, Santa Barbara, CA, USA, August 15-19, 2010. Proceedings 30*. Springer, 333–350.
- Kumar, M., Puran, M. and Koul, K. (2011). Robust rsa for digital signature. *International Journal of Computer Science Issues*, 8,.
- Kumar, N., Awasthi, Y. and Agarwal, R. P. (2016). Authenticating cloud & data center with iris. *International Journal of Engineering and Technical Research (IJETR)*, 4,. Print ISSN: 2454-4698.
- L., R. R. (1987). A method for obtaining digital signature and public-key cryptosystems. *ACM*, 21, 2. URL <https://cir.nii.ac.jp/crid/1570009749568213120>.
- Lajevardi, S. M. and Hussain, Z. M. (2009). Zernike moments for facial expression recognition. *rn*, 2, 3.
- Lauter, K. (2004). The advantages of elliptic curve cryptography for wireless security. *IEEE Wireless Communications*, 11, 62–67.
- Lecun, Y., Bottou, L., Bengio, Y. and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86, 2278–2324.
- Lei, S., Zishan, D. and Jindi, G. (2010). Research on key management infrastructure in cloud computing environment. In *2010 Ninth International Conference on Grid and Cloud Computing*. IEEE, 404–407. URL <http://dx.doi.org/10.1109/GCC.2010.84>.
- Lewinter, M. and Meyer, J. (2015). *Elementary number theory with programming*. John Wiley & Sons, Nashville, TN.
- Li, C., Li, J. and Gao, D. (2012). Fingerprint verification using weighted zernike moments and fast computation based on multi-disc roi. *International Journal of Digital Content Technology and its Applications*, 6, 546–555. URL <http://dx.doi.org/10.4156/jdcta.vol6.issue20.59>.
- LI, D., SHAN, J., SHAO, Z., ZHOU, X. and YAO, Y. (2013). Geomatics for smart cities - concept, key techniques, and applications. *Geo-spatial Information Science*, 16, 13–24. URL <http://dx.doi.org/10.1080/10095020.2013.772803>.
- Li, Y. and Xu, X. (2009). Revolutionary information system application in biometrics. In *2009 International Conference on Networking and Digital Society*, vol. 1. IEEE, 297–300.

- Lourida, K. G., Mouhtaropoulos, A. and Vakaloudis, A. (2013). Assessing database and network threats in traditional and cloud computing. *International Journal of Cyber-Security and Digital Forensics*, 2, 1–17. URL <https://api.semanticscholar.org/CorpusID:17631673>.
- Ma, L., Tan, T., Wang, Y. and Zhang, D. (2003). Personal identification based on iris texture analysis. *IEEE transactions on pattern analysis and machine intelligence*, 25, 1519–1533.
- Malik, R. and Kumar, P. (2015). Cloud computing security improvement using diffie hellman and aes. *International Journal of Computer Applications*, 118,.
- Marinescu, D. C. (2013). Introduction. In *Cloud Computing*. Elsevier, 1–19.
- Marton, K., Suci, A., Sacarea, C. and Cret, O. (2012). Generation and testing of random numbers for cryptographic applications. *Proceedings of the Rumanian Academy, Series A*, 13, 368–377.
- Masek, L. (2003). *Recognition of Human Iris Patterns for Biometric Identification*. Master’s thesis, University of Western Australia. URL <https://www.peterkovesi.com/studentprojects/libor/LiborMasekThesis.pdf>.
- Mell, P. and Grance, T. (2009). Draft nist working definition of cloud computing. URL <https://csrc.nist.gov/groups/SNS/cloud-computing>.
- Mishra, S. and Priyadarshini, S. B. (2023). Cloud-secured learning: Strengthening e-learning platforms for enhanced accessibility and protection. *PriMera Scientific Engineering*, 3,. Published on 3 September 2023, URL <https://primerascientific.com/pdf/psen/PSEN-03-073.pdf>.
- Mitali, V. K., Sharma, A. et al. (2014). A survey on various cryptography techniques. *International Journal of Emerging Trends & Technology in Computer Science (IJETTCS)*, 3, 307–312.
- Moghaddam, F. F., Alrashdan, M. T. and Karimi, O. (2013). A hybrid encryption algorithm based on rsa small-e and efficient-rsa for cloud computing environments. *Journal of Advances in Computer Networks* 238–241. URL <http://dx.doi.org/10.7763/JACN.2013.V1.47>.
- N.Kumbhar, N., V. Chaudhari, V. and A.Badhe, M. (2012). The comprehensive approach for data security in cloud computing: A survey. *International Journal of Computer Applications*, 39, 23–29. URL <http://dx.doi.org/10.5120/5080-7433>.
- Omeline, L., Goga, J., Pavlovicova, J., Oravec, M. and Jansen, B. (2021). A survey of iris datasets. *Image and Vision Computing*, 108, 104109. URL <http://dx.doi.org/10.1016/j.imavis.2021.104109>.
- O’Shea, K. (2015). An introduction to convolutional neural networks. *arXiv pre-print arXiv:1511.08458*.
- Paar, C. and Pelzl, J. (2014). *Understanding cryptography*. 2010th ed. Springer, Berlin, Germany.

- Prautzsch, H., Boehm, W. and Paluszny, M. (2002). *Bézier and B-Spline Techniques*. Springer Berlin Heidelberg. URL <http://dx.doi.org/10.1007/978-3-662-04919-8>.
- Pund, S. M. and Desai, C. G. (2014). Implementation of rsa algorithm using mersenne prime. *International Journal of Networking and Parallel Computing*, 1, 33–41.
- Qader, H. A., Ramli, A. R. and Al-Haddad, S. A. R. (2007). Fingerprint recognition using zernike moments. *Int. Arab J. Inf. Technol.*, 4, 372–376.
- Rajasekar, B. and Manigandan, S. (2015). An efficient resource allocation strategies in cloud computing. *International Journal of Innovative Research in Computer and Communication Engineering*, 3, 1239–1244.
- Raov, S., Raonk, N. and Kumari, E. (2009). Cloud computing: an overview. *Journal of Theoretical and Applied Information Technology (JATIT)*, 9, 71–76. URL <http://www.jatit.org/volumes/Vol9No1/10Vol9No1.pdf>.
- Rimal, B. P., Choi, E. and Lumb, I. (2009). A taxonomy and survey of cloud computing systems. In *2009 Fifth International Joint Conference on INC, IMS and IDC*. 44–51.
- Ross, A. A. and Flynn, P. (2008). *Handbook of biometrics*. Springer Science+Business Media, LLC.
- Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., Berg, A. C. and Fei-Fei, L. (2015). Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115, 211–252. URL <http://dx.doi.org/10.1007/s11263-015-0816-y>.
- Salman, H. M. (2012). Fuzzy bio-cryptography key generation. In *The 13th International Arab Conference on Information Technology ACIT*. 538–543.
- Saraswathi, K. and Balasubramaniam, D. R. (2010). Biocryptosystems for authentication and network security-a survey. URL <https://api.semanticscholar.org/CorpusID:114193155>.
- Sarkar, M. K. and Chatterjee, T. (2014). Enhancing data storage security in cloud computing through steganography. *International Journal on Network Security*, 5, 13.
- Sattarova Feruza, Y. and Kim, T. H. (2007). It security review: Privacy, protection, access control, assurance and system security. *International journal of multimedia and ubiquitous engineering*, 2, 17–32.
- Schoenberg, I. (1969). Cardinal interpolation and spline functions. *Journal of Approximation Theory*, 2, 167–206. URL [http://dx.doi.org/10.1016/0021-9045\(69\)90040-9](http://dx.doi.org/10.1016/0021-9045(69)90040-9).
- Schoenberg, I. J. (1988). *Cardinal Interpolation and Spline Functions IV. The Exponential Euler Splines*, chap. 3. Birkhäuser Boston, 88–110. URL http://dx.doi.org/10.1007/978-1-4899-0433-1_3.

- Shankar, M. and Akshaya, P. (2014). Hybrid cryptographic technique using rsa algorithm and scheduling concepts. *International Journal of Network Security & Its Applications*, 6, 39.
- Sharma, P., Gupta, A. K. and Vijay, A. (2012). Notice of violation of iee publication principles: Modified integer factorization algorithm using v-factor method. In *2012 Second International Conference on Advanced Computing & Communication Technologies*. IEEE, 423–425. This paper was found to be in violation of IEEE publication principles. See <https://doi.org/10.1109/ACCT.2012.73> for details., URL <http://dx.doi.org/10.1109/ACCT.2012.73>.
- Simonyan, K. (2014). Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*.
- Singh, A., Vaish, A. and Keserwani, P. K. (2014). Information security: Components and techniques. *International Journal of Advanced Research in Computer Science and Software Engineering*, 4,.
- Smart, N. P. (2016). *Cryptography Made Simple*. Springer International Publishing. URL <http://dx.doi.org/10.1007/978-3-319-21936-3>.
- Soofi, A. A., Khan, M. I. and Amin, F.-e. (2014). Encryption techniques for cloud data confidentiality. *International Journal of Grid and Distributed Computing*, 7, 11–20. URL <http://dx.doi.org/10.14257/ijgdc.2014.7.4.02>.
- Stallings, W. (2006). *Cryptography and network security*, 4/E. Pearson Education India.
- Stallings, W. (2013). *Cryptography and network security*. 6th ed. Pearson, Upper Saddle River, NJ.
- Subashini, S. and Kavitha, V. (2011). A survey on security issues in service delivery models of cloud computing. *Journal of Network and Computer Applications*, 34, 1–11. URL <https://www.sciencedirect.com/science/article/pii/S1084804510001281>.
- Subbiah, M., Muthukumaran, D. and Ramkumar, D. (2013). Enhanced survey and proposal to secure the data in cloud computing environment. *International Journal of Engineering Science*, 5,.
- Szegedy, C., Ioffe, S., Vanhoucke, V. and Alemi, A. (2017). Inception-v4, inception-resnet and the impact of residual connections on learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 31, URL <http://dx.doi.org/10.1609/aaai.v31i1.11231>.
- Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V. and Rabinovich, A. (2015). Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 1–9.
- Tanwer, A., Tayal, A., Hussain, M. and Reel, P. S. (2010). Thin apps store for smart phones based on private cloud infrastructure. In *2010 ITU-T Kaleidoscope: Beyond the Internet? - Innovations for Future Networks and Services*. 1–6.

- Teague, M. R. (1980). Image analysis via the general theory of moments*. *J. Opt. Soc. Am.*, 70, 920–930. URL <https://opg.optica.org/abstract.cfm?URI=josa-70-8-920>.
- Teh, C.-H. and Chin, R. (1988). On image analysis by the methods of moments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10, 496–513. URL <http://dx.doi.org/10.1109/34.3913>.
- Tirthani, N. and R, G. (2014). Data security in cloud architecture based on diffie hellman and elliptical curve cryptography. Cryptology ePrint Archive, Paper 2014/049. URL <https://eprint.iacr.org/2014/049>.
- Tisse, C.-l., Martin, L., Torres, L., Robert, M. et al. (2002). Person identification technique using human iris recognition. In *Proc. Vision Interface*. 294–299.
- University, S. (2024). Cs231n: Convolutional neural networks for visual recognition. Accessed: 2017-09-04, URL <https://cs231n.github.io/>.
- Vijayakumar, K. (2022). Security issues and algorithms in cloud computing. *Global J. Adv. Res*, 2, 569–574.
- Wayman, J. L., Jain, A. K., Maltoni, D. and Maio, D. (2005). *Biometric systems: Technology, design and performance evaluation*. Springer Science & Business Media.
- Yan, S. Y. (2012). *Computational number theory and modern cryptography*. Wiley-HEP Information Security Series, John Wiley & Sons, Nashville, TN.
- Yang, J., Xiong, N., Vasilakos, A. V., Fang, Z., Park, D., Xu, X., Yoon, S., Xie, S. and Yang, Y. (2011). A fingerprint recognition scheme based on assembling invariant moments for cloud computing communications. *IEEE Systems Journal*, 5, 574–583.
- Yang, J. and Zhang, X. (2012). Feature-level fusion of fingerprint and finger-vein for personal identification. *Pattern Recognition Letters*, 33, 623–628.
- Zeiler, M. D. and Fergus, R. (2014). *Visualizing and Understanding Convolutional Networks*, chap. 53. Springer International Publishing, 818–833. URL http://dx.doi.org/10.1007/978-3-319-10590-1_53.
- Zhang, M., Yang, B., Zhang, W. and Takagi, T. (2011). Multibiometric based secure encryption, authentication scheme with fuzzy extractor. *Int. J. Netw. Secur.*, 12, 50–57.
- Zhang, Q., Cheng, L. and Boutaba, R. (2010). Cloud computing: state-of-the-art and research challenges. *Journal of internet services and applications*, 1, 7–18.