

HYBRID RUBIK'S CUBE BASED ENCRYPTION FRAMEWORK FOR CANCELABLE BIOMETRICS

Submitted in partial fulfillment of the requirements for the award of
Bachelor of Engineering degree in Computer Science and Engineering

By

SALADI HARI RAMA NAIDU (Reg.No:40111111)
SAI PAVAN KRISHNA JAGARLAMUDI (Reg.No:40111099)



DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SCHOOL OF COMPUTING

SATHYABAMA

**INSTITUTE OF SCIENCE AND TECHNOLOGY
(DEEMED TO BE UNIVERSITY)**

CATEGORY -1 UNIVERSITY BY UGC

Accredited with Grade "A++" by NAAC, | 12B Status by UGC | Approved by AICTE

**JEPPIAAR NAGAR, RAJIV GANDHI SALAI,
CHENNAI - 600119**

APRIL- 2024



SATHYABAMA
INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Category - I University by UGC

Accredited "A++" by NAAC | Approved by AICTE

www.sathyabama.ac.in

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

BONAFIDE CERTIFICATE

This is to certify that this Project Report is the Bonafide work of **SAI PAVAN KRISHNA JAGARLAMUDI (Reg.No - 40111099)** and **SALADI HARI RAMA NAIDU (Reg.No - 40111111)** who carried out the Project entitled "HYBRID RUBIK'S CUBE BASED ENCRYPTION FRAMEWOR FOR CANCELABLE BIOMETRICS" under my supervision from November 2023 to April 2024.

Internal Guide

Dr. D. POORNIMA M.E., Ph.D.

Head of the Department

Dr. L. LAKSHMANAN M.E., Ph.D.

HEAD OF THE DEPARTMENT

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

SATHYABAMA

INSTITUTE OF SCIENCE AND TECHNOLOGY

(DEEMED TO BE UNIVERSITY)

Jeppiaar Nagar, Rajiv Gandhi Salai,

Chennai - 600 129

Submitted for Viva voce Examination held on 12/9/24

Internal Examiner

External Examiner

DECLARATION

I, SAI PAVAN KRISHNA JAGARLAMUDI (Reg. No – 40111099) hereby declare that the Project Report entitled “HYBRID RUBIK’S CUBE BASED ENCRYPTION FRAMEWORK FOR CANCELABLE BIOMETRICS” done by me under the guidance of Dr.D.POORNIMA, M.E., Ph.D., is submitted in partial fulfillment of the requirements for the award of Bachelor of Engineering degree in **Computer Science and Engineering**.

DATE: 10-4-2024

PLACE: Chennai

sai pavan krishna
SIGNATURE OF THE CANDIDATE

ACKNOWLEDGEMENT

I am pleased to acknowledge my sincere thanks to **Board of Management of Sathyabama Institute of Science and Technology** for their kind encouragement in doing this project and for completing it successfully. I am grateful to them.

I convey my thanks to **Dr. T. Sasikala M.E., Ph. D., Dean**, School of Computing, **Dr.L.Lakshmanan M.E., Ph.D.**, Head of the Department of Computer Science and Engineering for providing me necessary support and details at the right time during the progressive reviews.

I would like to express my sincere and deep sense of gratitude to my Project Guide **Dr.D.Poornima, M.E., Ph.D.**, for her valuable guidance, suggestions, and constant encouragement paved way for the successfully completion of my project work.

I wish to express my thanks to all Teaching and Non-teaching staff members of the **Department of Computer Science and Engineering** who were helpful in many ways for the completion of the project.

ABSTRACT

The need for high-speed and secure multi-biometric systems has grown in the last few decades. The progress in Internet applications and computer networks has prompted new problems with security and privacy. Having a reliable and secure means for storing biometrics is a necessary issue. Hence, data encryption and network security have become significant. Then, because of the fast progress in network development, humans can easily and arbitrarily distribute or access digital data from networks. Therefore, ownership security has become a significant issue for individuals, and it requires much interest. Thus, there is a significant threat to copyright owners and digital multimedia producers to conserve multimedia from intruder prospection to avert loss in transmitted data. Ten years ago, most modern security systems depended on biometrics. Unfortunately, these systems have suffered for a long time from hacking trials. If the biometric databases have been hacked and stolen, the saved biometrics will be lost forever. Thus, there is a bad need to develop new cancelable biometric systems. In a classical authentication system, a user can easily change a password if it is compromised. However, the user's biometrics are limited and unique, and if a user biometric is compromised, it will be impossible to change it in a particular system or at least difficult. Cancelable biometrics can be generated with intentional, repeatable distortions of biometric signals based on transforms. Hence, biometric templates are compared in the transform domain. Cancelable biometrics depend on transforming data to replace a single biometric template in the same or different systems. This paper presents a cancelable biometric system that depends on a novel hybrid encryption framework based on the Rubik' cube technique. It allows simultaneous encryption of multiple images. The suggested hybrid enrollment stage begins with chaotic Baker map permutation in the Cipher FeedBack (CFB) operation mode, Advanced Encryption Standard (AES) or Ron's Code (RC6) algorithm as a first stage for encrypting the multiple images, separately. After that, the output encrypted images are passed to the second encryption stage via Rubik's cube technique. Chaotic, RC6, or AES encrypted face images are utilized as the Rubik's cube faces. For the RC6 and AES algorithms, they add a degree of diffusion, while the chaotic algorithm adds a degree of permutation.

TABLE OF CONTENTS

CHAPTER NO.	TITLE	PAGE NO.
	ABSTRACT	v
	TABLE OF CONTENTS	vi
	LIST OF FIGURES	viii
1	INTRODUCTION	1
2	LITERATURE SURVEY	6
	2.1 Inferences from Literature Survey	8
	2.2 Open problems in Existing System	9
	2.3 Aim & Objectives	10
3	REQUIREMENTS ANALYSIS	12
	3.1 Feasibility Studies/Risk Analysis of the Project	12
	3.2 Software Requirements Specification Document	13
4	DESCRIPTION OF PROPOSED SYSTEM	15
	4.1 Selected Methodology or process model	23
	4.1.1 Hybrid encryption process	24
	4.1.2 Hybrid decryption process	29
	4.2 Architecture / Overall Design of Proposed System	31
	4.3 Project Management Plan	34
5	IMPLEMENTATION DETAILS	39
6	RESULTS AND DISCUSSION	37
	6.1 Input Design	37
	6.2 Output Design	37
	6.3 System Testing	38

7	CONCLUSION AND FUTURE WORK	37
	REFERENCES	40
	APPENDIX	42
	A. SOURCE CODE	42
	B. SCREENSHOTS	55
	C. RESEARCH PAPER	61
	D. CERTIFICATES	69

LIST OF FIGURES

FIGURE NO	FIGURE NAME	PAGE NO.
4.1	Encryption process of RC'6	17
4.2	Encryption and Decryption round of RC'6	18
4.3	AES Encryption process	19
4.4	Encryption process of CBC Mode	20
4.5	Encryption process of CFB Mode	21
4.6	Encryption process of OFB Mode	21
4.7	Conversation of 3D Rubik's cube into 2Dimage	22
4.8	Symmetric key encryption process	24
4.9	Asymmetric key encryption process	25
4.10	The proposed hybrid enrollment stage	27
4.11	Proposed framework implementation	31
4.12	Three degrees of encryption	32
5.1	Cancelable Biometric System	35
B.1	Encryption after kernal algorithm and after rubik's cube	55
B.2	Histograms for false faces in kernal and rubik's cube	56
B.3	Encryption kernal algorithm and rubik's cube algorithm for true	57
B.4	Histograms for true faces in kernal and rubik's cube	58
B.5	Correlation value for the false and true faces	59
B.6	Processing times comparison for different algorithms	60

CHAPTER 1

INTRODUCTION

Recent years have seen an exponential growth in the use of various cancelable biometric technologies. Unlike credit cards and passwords, which can be attacked and reissued when compromised, biometrics are permanently associated with a user's unique characteristics and cannot be replaced. Biometrics are either signals or images presented from persons for discrimination between them. The most common biometrics are fingerprints, faces, iris, and speech signals. The basic idea of the operation of biometric systems is to collect the biometrics from some authorized persons. Generally, face recognition depends on the geometric determination of eyes, nose, and mouth in the face images. All traditional biometric systems depend on acquiring a signal or an image from the users or subscribers, and then matching it with those stored in a pre-arranged database. The main disadvantage of this traditional trend is that each person has to provide his original biometrics for access. The enrollment data are saved in the database. This means that if the database is stolen, the original biometrics will be lost forever, and hence the system will not be suitable for utilization again.

Moreover, the users whose biometrics have been stolen will not use them again in other systems. A new trend towards more secure biometrics is to use cancelable biometrics for persons. These biometrics can be changed easily in emergency cases without changing the system at all. Moreover, encryption techniques can be used with all types of biometrics to generate cancelable templates. Different algorithms have been presented in the literature due to the importance and popularity of the recognition systems. Recent cancelable biometric concepts have been adopted for face, fingerprint, and iris recognition by using random kernels to generate cancelable image templates. This approach is called the Minimum Average Correlation Energy (MACE) filter approach. It adopts the correlation coefficient as a tool for verification without automatic classifiers.

Encryption is one of the most favorable methods to secure digital multimedia files in the domains of copyright protection and data authentication. Data encryption has more implementations in different fields such as medical imaging.

Communication, telemedicine, multimedia systems, and military communications. A biometric authentication system automatically identifies or verifies a person using physical, biological, and behavioral characteristics such as the face, iris, fingerprints, hand geometry, or voice. Compared to traditional methods of verification and identification (password, card, ID), biometrics are more suitable for users, can reduce attacks, and allow more security. Recently, modern biometric systems have been developed to enhance the security. The basic concept of cancelable biometrics is to use another version of the original biometric template created through a one-way transform or an encryption scheme to keep the original biometric safe and away from utilization in the system.

In this paper, our proposed cancelable biometric system is based on Rubik's cube technique, which depends on the puzzle concept. A puzzle is a problem that tests the ability of the solver's mind, imagination, skill, and cleverness. Puzzles were and are still classified as games and entertainment tools, but we can also consider them in mathematical problems or logical quizzes in different cases. People with a high attitude may be better than others when they are dealing with those puzzles. Also, those quizzes may be solved easily by computer programs. The proposed cryptosystem is based on Rubik's cube technique mixed with chaotic Baker map, RC6, and AES algorithms. The Initialization Vector (IV) of the chaotic map works as the main key, and the block is a square lattice of $N \times N$ pixels. The RC6 and AES algorithms will be used in this paper. The RC6 algorithm has six simple and basic operations, which are addition (+), subtraction (-), left rotation (>>>), right rotation (<<<), XOR ($\oplus$), and multiplication (*). The utilization of the multiplication process increases the diffusion per round greatly, leading to more security, fewer rounds, and finally, large throughput. The RC6 algorithm uses a key table consisting of $2r + 4w$ -bit words. The Rubik's cube technique adopted in this paper is used to achieve further permutation in the encryption process over its faces. It will be used to encrypt a group of images, simultaneously. The images implemented on these nine faces are encrypted with the chaotic, RC6 or AES technique. The proposed hybrid encryption framework will guarantee both diffusion and permutation in the encrypted images. Hence, the proposed biometric cryptosystem based on Rubik's cube principle can achieve good encryption and hiding ability and resist different types of attacks. Several encryption algorithms based on chaotic map, AES, and RC6 algorithms have been proposed to protect digital data.

This paper introduces cancelable face recognition systems. The first and classical cancelable face recognition system consists of random kernel generation, convolution, and authentication stages. The matcher can decide and classify the encrypted faces as accepted or rejected based on a threshold value. The quantitative evaluations of this proposed cancelable face recognition system depends on EER, False Acceptance Rate (FAR), False Rejection Rate (FRR), Area under Receiver Operating Characteristic (ROC) curve (AROC), and decidability. Another cancelable face recognition system based on the Rubik's cube technique is also presented. We present a novel image encryption enrollment stage based on Rubik's cube technique, achieving good encryption and perfect hiding ability and resisting different attacks.

The rest of this paper is organized as follows. the review and related studies are introduced, the proposed hybrid encryption framework is presented. In Section 4, the cancelable biometric templates are presented. Section 5 is devoted to system architecture. Section 6 presents the authentication metrics of both schemes. Section 7 gives the simulation results of the paper. Section 8 offer the concluding remarks and future research directions, The RC6 algorithm depends on four processing registers, and each register works with 32 bits to handle 128 bits as a total size of the input and output blocks. Therefore, the RC6 is parameterized by the word size (w) in bits, the number of rounds (r), and the encryption/decryption key length in bytes (b). shows the different stages of the RC6 encryption algorithm. Hence, the proposed biometric cryptosystem based on Rubik's cube principle can achieve good encryption and hiding ability and resist different types of attacks. Several encryption algorithms based on chaotic map, AES, and RC6 algorithms have been proposed to protect digital data against different cryptographic attacks. This paper introduces cancelable face recognition systems. The first and classical cancelable face recognition system consists of random kernel generation, convolution, and authentication stages. The matcher can decide and classify the encrypted faces as accepted or rejected based on a threshold value.

Why do people need several security and privacy concerns with biometrics? The answer of this question can be summarized as follows:

1. Passwords and cryptographic keys are known only to the user, and hence secrecy can be maintained. In contrast, biometrics such as voice, face, signature, and fingerprints can be easily recorded and misused without the user's knowledge.

2. If a hacker gets access to the biometric samples and has the ability to present them to a system emulating a human presence, there..

Traditional identity authentication methods are based on what is physically possessed, such as ID cards, and what can be mentally stored in the memory, such as passwords and keys. For instance, the shortfalls of both are that ID cards can easily be lost or forged, while passwords and keys can either be easily guessed or forgotten, respectively. Short passwords are often preferred for memory requirements, but easily guessed by others. On the other hand, long passwords (commonly known as keys)..

The design of architectures and algorithms for multimedia security needs much effort to be suitable for embedded and mobile objects with low computational resource constraints. However, the high complexity and cost are significant problems in a cryptography system. In general, conventional encryption algorithms are complex and they require several rounds for encryption and they are computationally expensive leading to much energy loss in multimedia devices. The implementation of lightweight encryption algorithms is an attractive topic for securing multimedia content. Perception, network/transport, and application layers are the three main layers of any IoT wireless architecture. Each layer is susceptible to different kind of attacks. So, the need for security systems and analysis is the main task in any wireless network design to detect the different types of attacks. The main advantages of Latin Square are that they allow experiments using a small number of runs and they provide better efficiency than other conventional cryptosystems. The Latin Square algorithm uses a short key to produce large-sized squares, and then it constructs the encryption primitives using Latin square keys for encrypting video frames. So, we can easily adjust the Latin Squares' sizes and the key length.

particular application. The simulation results with MATLAB show that the whole scheme is simple and suitable for secure IoT multimedia communication compared to the state-of-the-art algorithms.

The rest of this paper is arranged as follows presents a review of some of the previous and related works gives an explanation of the proposed compression-encryption scheme. Compared to traditional methods of verification and identification (password, card, ID), biometrics are more suitable for users, can reduce attacks, and allow more security. Recently, modern biometric systems have been developed to enhance the security. The basic concept of cancelable biometrics is to use another version of the original biometric template created through a one-way transform or an encryption scheme to keep the original biometric safe and away from utilization in the system. Therefore, ownership security has become a significant issue for individuals, and it requires much interest. Thus, there is a significant threat to copyright owners and digital multimedia producers to conserve multimedia from intruder prospection to avert loss in transmitted data. Ten years ago, most modern security systems depended on biometrics. Unfortunately, these systems have suffered for a long time from hacking trials. If the biometric databases have been hacked and stolen, the saved biometrics will be lost forever. Thus, there is a bad need to develop new cancelable biometric systems. In a classical authentication system, a user can easily change a password if it is compromised. However, the user's biometrics are limited and unique, and if a user biometric is compromised, it will be impossible to change it in a particular system or at least difficult. Cancelable biometrics can be generated with intentional, repeatable distortions of biometric signals based on transforms. Hence, biometric templates are compared in the transform domain. Cancelable biometrics depend on transforming data to replace a single biometric template in the same or different systems. This paper presents a cancelable biometric system that depends on a novel hybrid encryption framework based on the Rubik' cube technique. It allows simultaneous encryption of multiple images. The suggested hybrid enrollment stage begins with chaotic Baker map permutation in the Cipher FeedBack (CFB) operation mode, Advanced Encryption Standard (AES) or Ron's Code (RC6) algorithm as a first stage for encrypting.

CHAPTER 2

LITERATURE SURVEY

In the literature survey of Hybrid Rubik's Cube Based Encryption Framework For Cancelable Biometrics, we explore key papers published in the years 2022 and 2023, each contributing unique insights and methodologies to this field.

The literature survey conducted by William et al. [1] focuses on crime analysis using a computer vision approach with machine learning. The authors explore the use of computer vision techniques to analyze crime-related data and apply machine learning algorithms for crime prediction. The study demonstrates the potential of this approach in improving crime prevention and law enforcement.

Dakalbab et al. [2] present a systematic literature review on artificial intelligence and crime prediction. The researchers investigate the existing studies on the use of AI in predicting criminal activities and identify the trends, challenges, and future directions in this field. The review provides valuable insights into the potential of AI in combating crime and enhancing public safety.

Srinivasulu Raju et al. [3] discuss the analysis and prediction of crime using various machine learning techniques. The authors examine the application of machine learning algorithms in predicting crime patterns and patterns. Their research provides a comprehensive overview of the use of machine learning in criminal analysis and its potential impact on law enforcement.

Lombo et al. [4] propose a crime detection and analysis system that utilizes machine learning and natural language processing techniques to analyze social media messages. The study highlights the importance of leveraging social media data for crime prevention and demonstrates the efficacy of machine learning algorithms in analyzing textual data for criminal activities.

Norouzi [5] presents a spatial, temporal, and semantic crime analysis approach using information extraction from online news. The author explores the use of information

extraction techniques to extract relevant crime-related information from online news sources. The study demonstrates the utility of such approaches in understanding crime patterns and trends.

SH et al. [6] propose a prediction and prevention analysis system that utilizes machine learning algorithms to detect and analyze crime data. The authors apply various machine learning algorithms to predict crime incidents and evaluate their effectiveness in preventing crimes. Their research contributes to the development of predictive policing strategies

Travaini et al. [7] conduct a systematic review of advanced methodologies for recidivism risk prediction in the field of criminal justice. The researchers analyze the existing literature on machine learning techniques for recidivism risk prediction and discuss the strengths and limitations of different approaches. The review provides a comprehensive overview of advanced methodologies in this area.

Adhikary et al. [8] propose an edge-assisted crime prediction and evaluation framework that combines machine learning algorithms with edge computing technology. The authors investigate the potential of edge computing in enhancing crime prediction and evaluation processes. Their research contributes to the development of efficient and scalable crime prediction systems.

Kshatri et al. [9] propose a crime detection approach using big data analytics and machine learning. The authors explore the use of big data analytics techniques to analyze crime-related data and apply machine learning algorithms for crime detection. The study demonstrates the effectiveness of this approach in identifying criminal activities.

Gandapur [10] presents an end-to-end video surveillance-based deep learning model for detecting and preventing criminal activities. The author introduces a comprehensive model that combines video surveillance and deep learning techniques to detect criminal activities in real-time. The research provides insights into the application of deep learning in video.

2.1.INFERENCES FROM LITREATURE SURVEY

The proposed encryption scheme is based on the chaotic Baker's map, Rubik's cube, AES, and RC6 algorithms. This is used to concurrently encrypt a collection of photos. The chaotic-AES, RC6 encryption method is used to protect the images in this collection. The proposed architecture will guarantee that the results of the encryption will be both permuted and diffused. As a result, the suggested biometric standard built on the Rubik's cube algorithm may achieve respectable encryption and provide protection against various threats. This proposed method enhances permutation, diffusion, and circulation to the encrypted biometric template, making it difficult for the hackers to break in. In actual-time execution, the Rubik's cube algorithm requires less time. In order to identify which method takes the least amount of time to process, numerous algorithms were benchmarked in different execution orders.

A growing security trend in the field of biometric authentication is cancellable biometrics. An optical image encryption technique is used by the suggested approach. It performs well and offers respectable encryption and cancelability. The suggested technique can withstand a variety of cryptanalytical attack according to the security analysis. It is suggested to use a novel hybrid chaotic map to improve performance of the encryption methods. The proposed mathematical function has qualities that are superior to those of conventional maps. Encryption is one of the most favorable methods to secure digital multimedia files in the domains of copyright protection and data authentication. Data encryption has more implementations in different fields such as medical imaging, Internet communication, telemedicine, multimedia systems, and military communications. A biometric authentication system automatically identifies or verifies a person using physical, biological, and behavioral characteristics such as the face, iris, fingerprints, hand geometry, or voice. Generally, face recognition depends on the geometric determination of eyes, nose, and mouth in the face images. All traditional biometric systems depend on acquiring a signal or an image from the users or subscribers, and then matching it with those stored in a pre-arranged database.

2.2. OPEN PROBLEMS IN EXISTING SYSTEM

Although the hybrid encryption has revolutionized computer vision, current approaches have several major problems: typical vision datasets are labor intensive and costly to create while teaching only a narrow set of visual concepts; standard vision models are good at one task and one task only, and require significant effort to adapt to a new task; and models that perform well on benchmarks have disappointingly poor performance on stress tests, casting doubt on the entire deep learning approach to computer vision.

Most of the existing search engines are done based on the text to text similarity analysis only. This results in wastage of images which doesn't have textual descriptions or captions. They are not Zero-Shot learners. Each search application requires special computation for data specific training.

Moreover, the users whose biometrics have been stolen will not use them again in other systems. A new trend towards more secure biometrics is to use cancelable biometrics for persons. These biometrics can be changed easily in emergency cases without changing the system at all. Moreover, encryption techniques can be used with all types of biometrics to generate cancelable templates. Different algorithms have been presented in the literature due to the importance and popularity of the recognition systems. Recent cancelable biometric concepts have been adopted for face, fingerprint, and iris recognition by using random kernels to generate cancelable image templates. This approach is called the Minimum Average Correlation Energy (MACE) filter approach. It adopts the correlation coefficient as a tool for verification without automatic classifiers. The correlation values are used to measure the similarity between a test pattern and stored biometric templates. The higher the score, the higher the similarity between patterns is. Access to the system is granted only if the score for a test person is higher than a certain threshold. ROC analysis is used to quantify how accurately a system can discriminate between two states. The ROC curve shows the trade-off between a true positive fraction (TPF) and a false-positive fraction. The project's proposed hybrid encryption technique adds more permutations and diffusions to the recorded biometric data. The positive achievements of the proposed hybrid architecture in securely storing the encryption of biometric images were validated results.

2.3 AIM & OBJECTIVES

Aim:

The aim of this project is to develop a robust encryption framework utilizing a hybrid approach based on Rubik's Cube principles for enhancing the security of cancelable biometrics systems.

Objectives:

Research and Analysis:

Conduct an in-depth review of existing cancelable biometrics techniques and encryption methods

Analyze the strengths and weaknesses of current encryption frameworks and cancelable biometrics systems.

Investigate the principles behind Rubik's Cube manipulation and its potential applications in encryption.

Design and Development:

Design a hybrid encryption framework that integrates Rubik's Cube principles with cancelable biometrics techniques.

Develop algorithms for generating cancelable biometric templates and transforming them into Rubik's Cube configurations.

Implement encryption and decryption algorithms based on Rubik's Cube operations to ensure secure biometric data transmission and storage.

Evaluation and Optimization:

Evaluate the performance of the proposed framework in terms of security, computational efficiency, and usability.

Conduct experiments to measure the resistance of the framework against common attacks such as brute-force and dictionary attacks.

Optimize the algorithms and parameters to enhance the overall efficiency and security.

Integration and Compatibility:

Ensure compatibility and interoperability of the encryption framework with existing

biometric authentication systems and databases.

Develop interfaces for seamless integration with various biometric modalities such as fingerprint, iris, and face recognition.

Test the framework's compatibility with different platforms and environments, including mobile devices and cloud computing.

User Acceptance and Usability:

Conduct usability studies to evaluate the user experience and acceptance of the proposed encryption framework.

Gather feedback from users and stakeholders to identify potential areas for improvement and refinement.

Incorporate user-friendly features and interfaces to enhance the accessibility and adoption of the system.

Documentation and Knowledge Dissemination:

Document the design, implementation, and evaluation processes of the encryption framework.

Prepare technical reports, research papers, and presentations for publication in relevant.

By achieving these objectives, the project aims to contribute to the advancement of secure and privacy-preserving biometric authentication systems, thereby addressing the growing concerns regarding data privacy and security in various applications.

- Eliminate the requirement for storing original biometrics in order to gain access.
- Propose hybrid encryption framework to guarantee both diffusion and permutation in the encrypted images.
- Propose biometric cryptosystem based on Rubik's cube principle to achieve good encryption and hiding ability and resist different types of attacks.
- Takes an input related to biometrics, encrypts the data using various encryption algorithms and apply rubik's cube encryption algorithm.

CHAPTER 3

REQUIREMENT ANALYSIS

3.1 FEASIBILITY STUDIES/RISK ANALYSIS OF THE PROJECT

The feasibility of the project is analyzed in this phase and business proposal is put forth with a very general plan for the project and some cost estimates. During system analysis the feasibility study of the proposed system is to be carried out. This is to ensure that the proposed system is not a burden to the company. For feasibility analysis, some understanding of the major requirements for the system is essential.

Three key considerations involved in the feasibility analysis are

- **ECONOMICAL FEASIBILITY**
- **TECHNICAL FEASIBILITY**
- **SOCIAL FEASIBILITY**

ECONOMICAL FEASIBILITY

This study is carried out to check the economic impact that the system will have on the organization. The amount of fund that the company can pour into the research and development of the system is limited. The expenditures must be justified. Thus the developed system as well within the budget and this was achieved because most of the technologies used are freely available. Only the customized products had to be purchased.

TECHNICAL FEASIBILITY

This study is carried out to check the technical feasibility, that is, the technical requirements of the system. Any system developed must not have a high demand on the available technical resources. This will lead to high demands on the available technical resources. This will lead to high demands being placed on the client. The developed system must have a modest requirement, as only minimal or null changes are required for implementing this system.

SOCIAL FEASIBILITY

The aspect of study is to check the level of acceptance of the system by the user. This includes the process of training the user to use the system efficiently. The user must not feel threatened by the system, instead must accept it as a necessity. The level of acceptance by the users solely depends on the methods that are employed to educate the user about the system and to make him familiar with it. His level of confidence must be raised so that he is also able to make some constructive criticism, which is welcomed, as he is the final user of the system.

3.2 SOFTWARE REQUIREMENTS SPECIFICATION DOCUMENT

Functional and non-functional requirements:

A software requirements specification (SRS) is a document that describes what the software will do and how it will be expected to perform.

Functional Requirements:

A Functional Requirement (FR) is a portrayal of the assistance that the product should offer. It describes a product framework or its parts. It tends to be a computation, information control, business process, client association, or whatever other explicit usefulness which characterizes which work a framework is probably going to perform. Functional Requirements are likewise called Functional Specifications.

- Search engine results the images based on the intention of the search phrase.
- The Algorithm should captionate the image if there is no previous caption or textual description and store it in a database or repository.

Non-functional requirements:

Non-functional requirements are requirements that are not directly related to a particular feature of the system. You can see system properties such as reliability, performance and time. Here are some of the non-functional requirements:

- Search results should be closely related and diversified.
- They doesn't too much specific to search phrase.

- It shouldn't take much time to produce results.
- Information shouldn't be lost.
- Should help to meet the Organizational needs.
- Should help to meet the user needs.

HARDWARE & SOFTWARE REQUIREMENTS

H/W CONFIGURATION:

- ◆ 8GB Ram
- ◆ Multicore CPU
- ◆ Intel i3 or Ryzen-3 or above Processor.
- ◆ NVIDIA 1650 GTX GPU

S/W CONFIGURATION:

- ◆ Numpy
- ◆ matplotlib.pyplot
- ◆ pillow-PIL
- ◆ Opencv

CHAPTER 4

DESCRIPTION OF PROPOSED SYSTEM

The need for high-speed and secure multi-biometric systems has grown in the last few decades. The progress in Internet applications and computer networks has prompted new problems with security and privacy. Having a reliable and secure means for storing biometrics is a necessary issue. Hence, data encryption and network security have become significant. Thence, because of the fast progress in network development, humans can easily and arbitrarily distribute or access digital data from networks. Therefore, ownership security has become a significant issue for individuals, and it requires much interest. Thus, there is a significant threat to copyright owners and digital multimedia producers to conserve multimedia from intruder prospection to avert loss in transmitted data. Ten years ago, most modern security systems depended on biometrics. Unfortunately, these systems have suffered for a long time from hacking trials. If the biometric databases have been hacked and stolen, the saved biometrics will be lost forever. Thus, there is a bad need to develop new cancelable biometric systems. In a classical authentication system, a user can easily change a password if it is compromised. However, the user's biometrics are limited and unique, and if a user biometric is compromised, it will be impossible to change it in a particular system or at least difficult. Cancelable biometrics can be generated with intentional, repeatable distortions of biometric signals based on transforms. Hence, biometric templates are compared in the transform domain. Cancelable biometrics depend on transforming data to replace a single biometric template in the same or different systems. This paper presents a cancelable biometric system that depends on a novel hybrid encryption framework based on the Rubik' cube technique. It allows simultaneous encryption of multiple images. The suggested hybrid enrollment stage begins with chaotic Baker map permutation in the Cipher FeedBack (CFB) operation mode, Advanced Encryption Standard (AES) or Ron's Code (RC6) algorithm as a first stage for encrypting the multiple images, separately. After that, the output encrypted images are passed to the second encryption stage via Rubik's cube technique. Chaotic, RC6, or AES encrypted face images are utilized as the Rubik's cube faces. For the RC6 and AES algorithms, they add a degree of diffusion, while the chaotic algorithm adds a degree of permutation. Moreover, the Rubik's cube technique adds more permutations to the encrypted images, simultaneously.

The encrypted images are used in the cancelable biometric system. The simulation results prove that the hybrid proposed encryption framework is efficient. Moreover, it has strong robustness and security.

- 1.RC6-Process
- 2.AES-Algorithm
- 3.ChaoticCBC
- 4.Rubik's cube technique

1.RC6-Process:

The RC6 algorithm depends on four processing registers, and each register works with 32 bits to handle 128 bits as a total size of the input and output blocks. Therefore, the RC6 is parameterized by the word size (w) in bits, the number of rounds (r), and the encryption/decryption key length in bytes.

Like RC5, RC6 is a fully parameterized family of encryption algorithms. A version of RC6 is more accurately specified as RC6- $w/r/b$ where the word size is w bits, encryption consists of a nonnegative number of rounds r , and b denotes the length of the encryption key in bytes. Since the AES submission is targeted at $w = 32$ and $r = 20$, we shall use RC6 as shorthand to refer to such versions. When any other value of w or r is intended in the text, the parameter values will be specified as RC6- w/r . Of particular relevance to the AES effort will be the versions of RC6 with 16-, 24-, and 32-byte keys. For all variants, RC6- $w/r/b$ operates on units of four w -bit words using the following six basic operations. The base-two logarithm of w will be denoted by $\lg w$.

- $a + b$: integer addition modulo 2^w
- $a - b$: integer subtraction modulo 2^w
- $a @ b$: bitwise exclusive-or of w -bit words
- $a * b$: integer multiplication modulo 2^w
- $a \lll b$: rotate the w -bit word a to the left by the amount given by the least significant $\lg w$ bits of b
- $a \ggg b$: rotate the w -bit word a to the right by the amount given by the least significant $\lg w$ bits of b

Note that in the description of RC6 the term "round" is somewhat analogous to the usual DES-like idea of a round: half of the data is updated by the other half; and the two are then swapped. In RC5, the term "half-round" was used to describe this style of action, and an RC5 round was deemed to consist of two half-rounds. This seems to have become a potential cause of confusion, and so RC6 reverts to using the term "round" in the more established way.

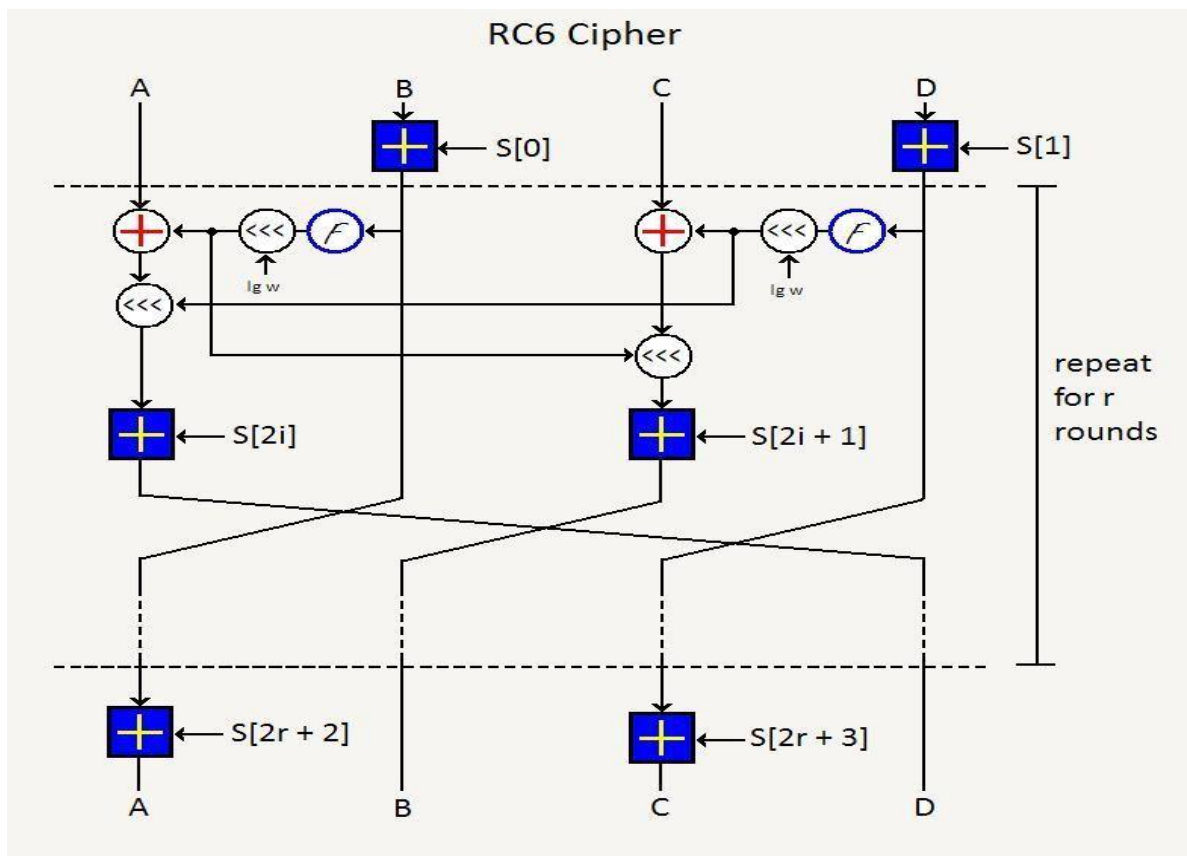


Fig:4.1. Encryption Process of RC'6

RC6 is an evolutionary improvement of RC5, designed to meet the requirements of the Advanced Encryption Standard (AES). Like RC5, RC6 makes essential use of data-dependent rotations. New features of RC6 include the use of four working registers instead of two, and the inclusion of integer multiplication as an additional primitive operation. The use of multiplication greatly increases the diffusion achieved per round, allowing for greater security, fewer rounds, and increased throughput. I found an article about it online and fulfilled .

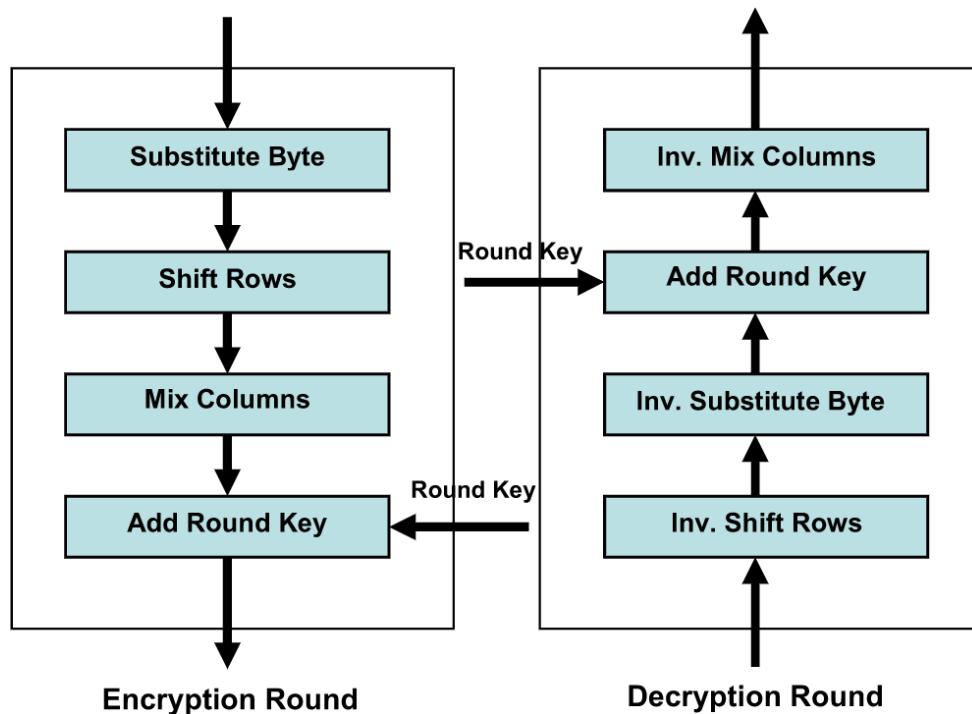


Fig:4.2. Encryption and Decryption round of RC62.

2.AES-Algorithm:

The symmetric encryption algorithm with the most widespread application is called AES, or Advanced Encryption Standard. It is mostly utilized to encrypt and safeguard electronic data. Because it is quicker and more effective than DES (Data encryption standard), it was used to replace DES. Three block ciphers make up the AES encryption system, which is used to encrypt data. NIST (National Institute of Standards and Technology) created AES in 1997. It was created to take the place of DES, which was sluggish and open to many assaults. In order to fix DES's flaws, a new encryption algorithm was developed. AES was subsequently released on November 26, 2001.

The United States National Institute of Standards and Technology (NIST) developed the Advanced Encryption Standard (AES) as a standard for the encryption of electronic data in 2001. Despite being more difficult to build, AES is still commonly used because it is substantially stronger than DES and triple DES.

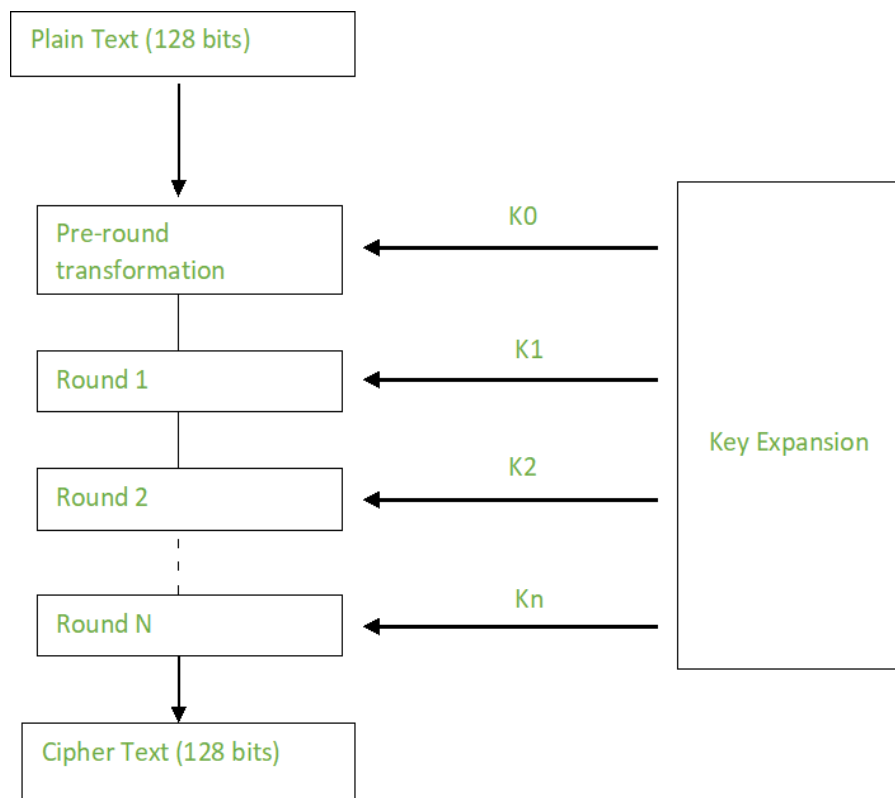


Fig:4.3. AES Encryption process.

Encryption is a technique for making data messages or files unreadable, guaranteeing that only a person with the proper authorization may access that data. Data is encrypted using sophisticated algorithms, and the same data is then decrypted using a key that is supplied by the message's sender. Information is kept secret and confidential by encryption while it is being stored or sent. Any unauthorized access will just reveal a disorganized collection of bytes.

Two Belgian cryptographers, Joan Daemen and Vincent Rijmen, created the AES variation of the Rijndael block cypher and presented a proposal to NIST as part of the AES selection process. A family of cyphers called Rijndael has various key and block sizes. NIST chose three Rijndael family members for AES, each having a 128-bit block size but three distinct key lengths: 128, 192, and 256 bits. The algebraic foundation of AES is pretty straightforward. Nicolas Courtois and Josef Pieprzyk published a theoretical assault in 2002 called the "XSL attack" that claimed to demonstrate a flaw in the AES algorithm, largely as a result of the nonlinear components' lack of complexity.

3. Chaotic CBC

CBC mode has several advantages. In fact, this mode encrypts the same plaintext differently with different initialization vectors. In addition, the encryption of each block depends on the preceding block and therefore, if the order of the cipher text blocks is modified, the decryption will be impossible and the recipient realizes the problem. Furthermore, if a transmission error affects the encrypted block C_i , then only the blocks m_i and m_{i+1} are affected, the other blocks will be determined correctly. CBC has been the most commonly used mode of operation. Its main drawbacks are that encryption is sequential (i.e., it cannot be parallelized), and that the message must be padded to a multiple of the cipher block size. One way to handle this last issue is through the method known as cipher text stealing. Note that a one-bit change in a plaintext or IV affects all following cipher text blocks. Decrypting with the incorrect initialization vector causes the first block of plaintext to be corrupted, but subsequent plaintext blocks will be correct. This is because a plaintext block can be recovered from two adjacent blocks of cipher text. As a consequence, decryption can be parallelized. Note that a one-bit change on the cipher text causes complete corruption of the corresponding block of plaintext, and inverts.

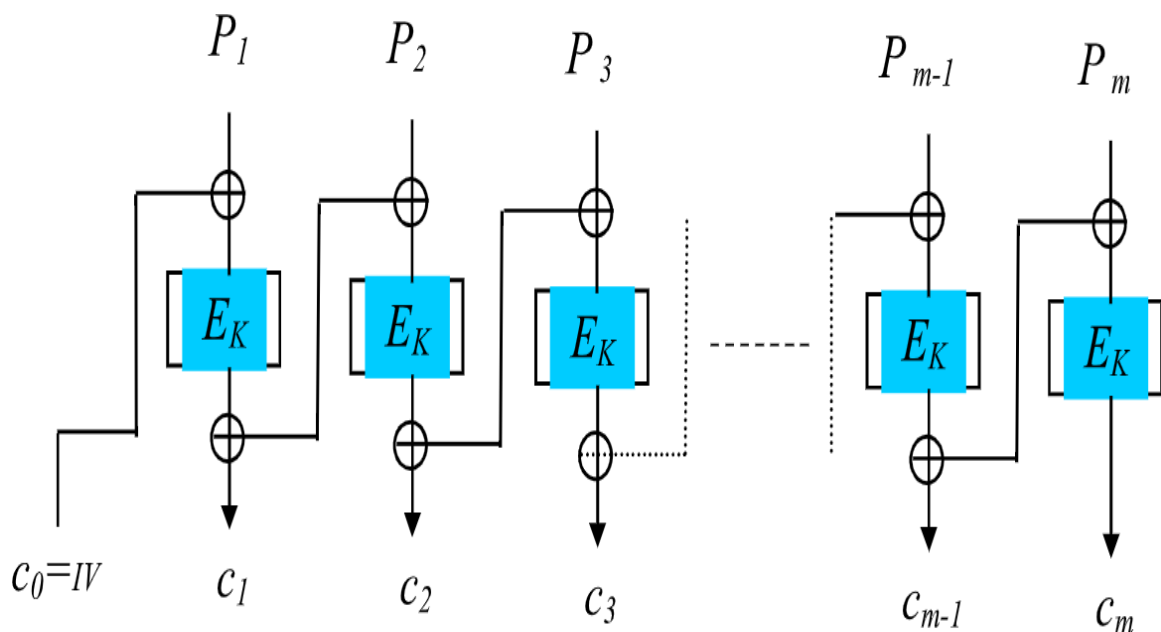


Fig:4.4. Encryption process of CBC mode

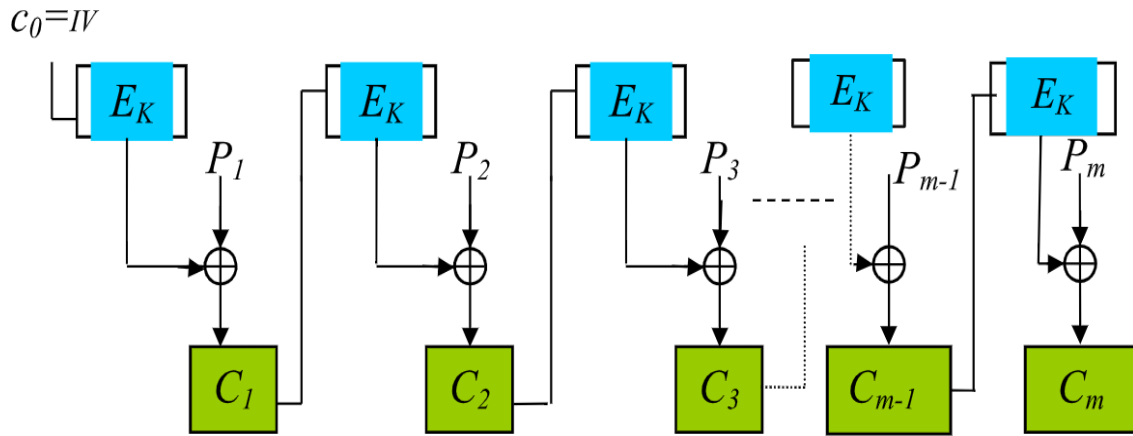


Fig:4.5. Encryption of CFB Mode

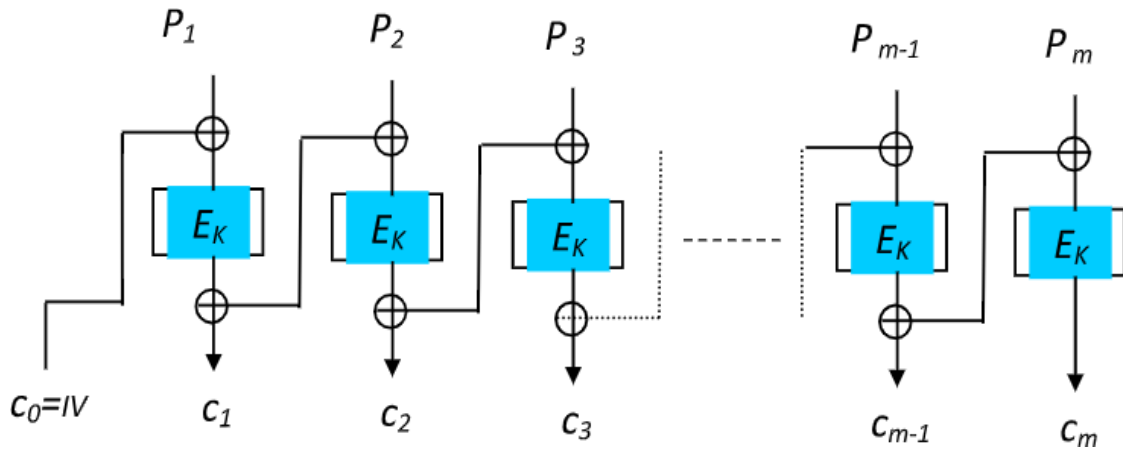


Fig:4.6. Encryption process of OFB Mode

The main objective of this series of articles regarding the chaotic topological behavior of the CBC mode of operation is to understand in which extent this mode depends on its inputs. More precisely, is it possible to understand this dependence, in such a way that the effects of a modification of the IV and/or the message can be predicted? Ifso, this kind of weakness could be considered in the design of specific attacks, whileif the converse is proven, that is to say, if the mid-to-long term effects of a slight modification of the input cannot be predicted, that chaotic dependence will make such attacks inefficient.

4. Rubik's cube algorithm

Apply Rubik's cube technique:

- a. Sum of each row is calculated.
- b. Applying modulus of 2 on the calculated sum.
- c. If the sum of the row is equal to '0', right circular shift is performed.
- d. Sum of each column is calculated.
- e. Applying modulus of 2 on the calculated sum.
- f. If the sum of the row is equal to '0', up circular shift is performed.
- g. Encryption of biometrics such as faces, fingerprints, iris with chaotic.

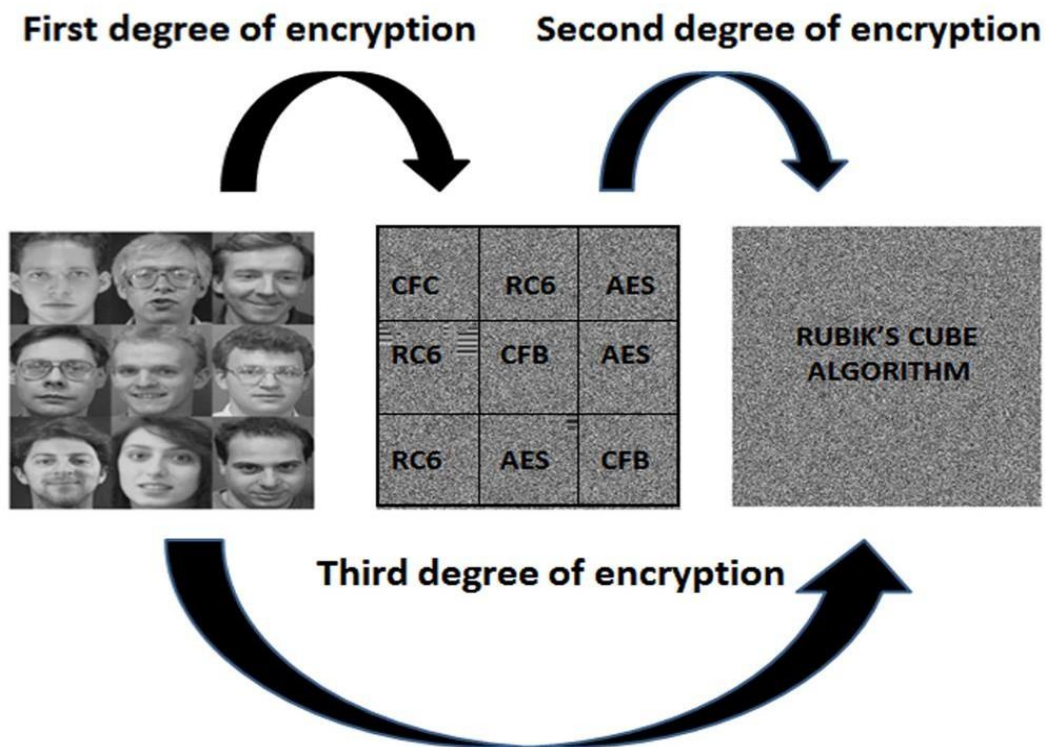


Fig:4.7. Conversation of 3D rubik's cube into 2D Image

4.1 SELECTED METHODOLOGY OR PROCESS MODEL

- **Bio hashing:** A Biometric template protection scheme that uses a user-specific key to transform the original biometric features into a secure hash.
- **Hybrid Methods:** Hybrid methods in cancelable biometrics are methods that combine two or more techniques to generate a cancelable biometric template.
- **Salting Methods:** It is a technique to protect biometric data by adding random noise (called salt) to it before encrypting it.
- **Bio Convolution:** It is a technique to distort biometric data by applying a random convolution filter to it before encrypting it.
- Encrypt the faces with chaotic, AES, or RC6 algorithm.
- Set the encrypted faces over the Rubik's cube faces.
- Convert all faces into a 2D image with a larger size.
- Randomize the converted 2D image with Rubik's cube technique.

Encryption of biometrics such as faces, fingerprints, iris with chaotic baker's AES or RC6 algorithm.

Apply Rubik's cube technique:

- Sum of each row is calculated.
- Applying modulus of 2 on the calculated sum.
- If the sum of the row is equal to '0', right circular shift is performed.
- Sum of each column is calculated.
- Applying modulus of 2 on the calculated sum.
- If the sum of the row is equal to '0', up circular shift is performed.
- Performed for all rows and columns.

4.1.1 HYBRID ENCRYPTION PROCESS

Hybrid encryption is a cryptographic technique that combines the advantages of both symmetric and asymmetric encryption schemes to provide a secure means of data transmission or storage. Let's delve into more detail on how it works:

Symmetric Encryption:

Symmetric encryption uses a single key to both encrypt and decrypt data. It's fast and efficient for large volumes of data. However, the challenge lies in securely sharing the secret key between the communicating parties. If an attacker intercepts the key during transmission, they could decrypt the data.

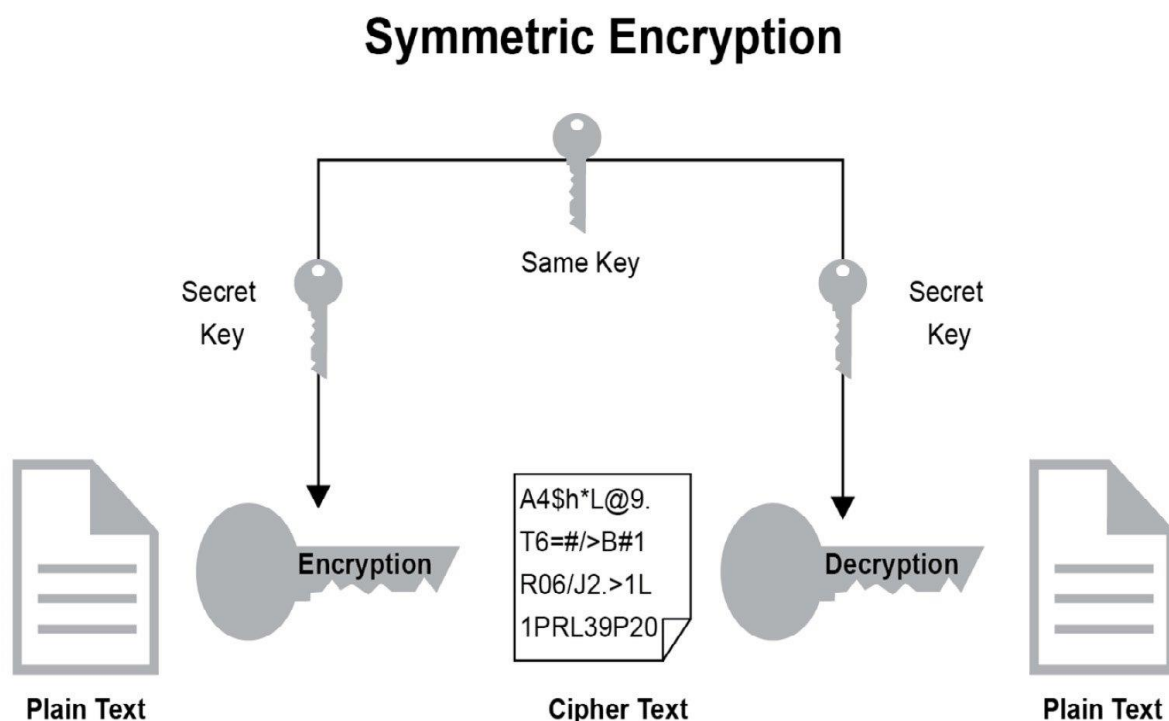


Fig: 4.8. Symmetric key encryption process

Symmetric ciphers have historically been susceptible to known-plaintext attacks, chosen-plaintext attacks, differential cryptanalysis and linear cryptanalysis. Careful construction of the functions for each round can greatly reduce the chances of a successful attack. It is also possible to increase the key length or the rounds in the encryption process to better protect against attack. This, however, tends to increase the processing power and decrease the speed at which the process runs due to the amount of operations the system needs to do.

For example, a 128 bit AES cipher would not be secure against such an attack as it would reduce the time required to test all possible iterations from over 10 quintillion years to about six months. By contrast, it would still take a quantum computer the same amount of time to decode a 256 bit AES cipher as it would a conventional computer to decode a 128 bit AES cipher. For this reason, AES-256 is believed to be "quantum resistant".

Asymmetric Encryption:

Asymmetric encryption, also known as public-key encryption, uses a pair of keys: a public key and a private key. The public key is widely distributed and used for encryption, while the private key is kept secret and used for decryption. This eliminates the need to securely share a secret key, as anyone can encrypt data using the recipient's public key, but only the recipient possessing the corresponding private key can decrypt it.

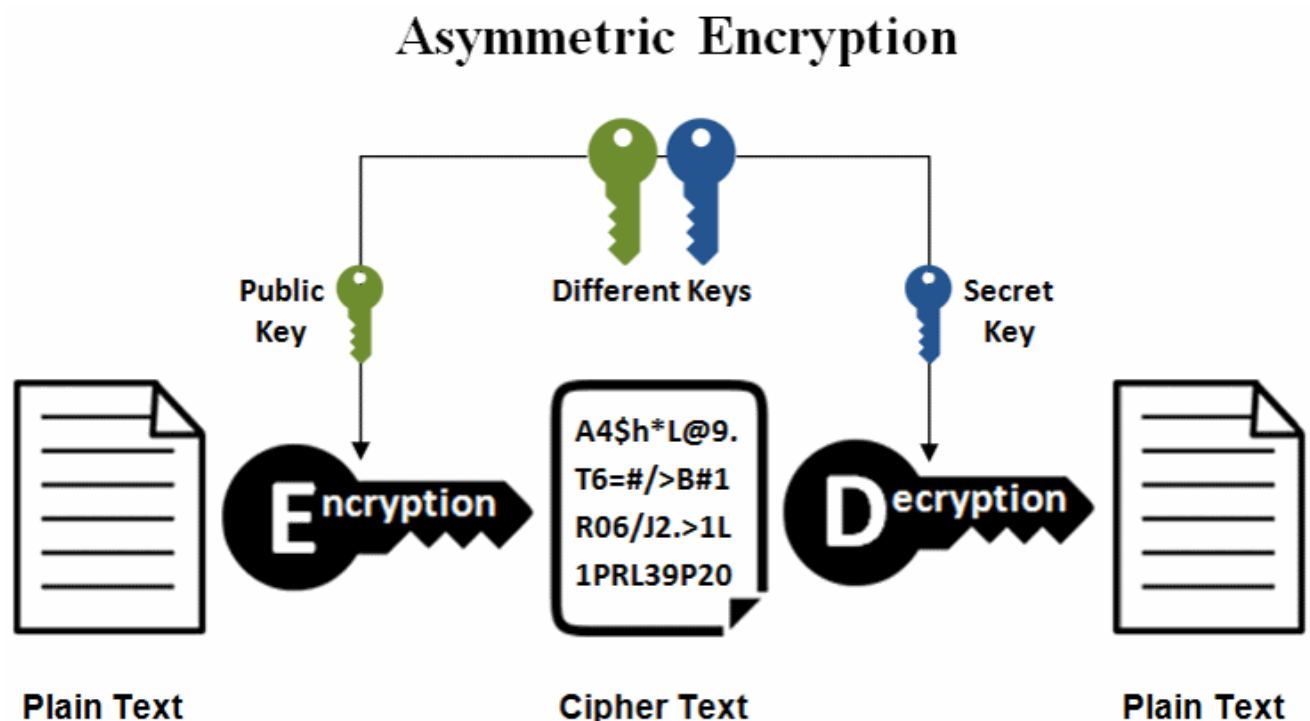


Fig:4.9. Asymmetric key encryption process

A public key is a cryptographic key that can be used by any person to encrypt a message so that it can only be decrypted by the intended recipient with their private key. A private key also known as a secret key is shared only with key's initiator. When someone wants to send an encrypted message, they can pull the intended recipient's public key from a public directory and use it to encrypt the message before sending it. The recipient of the

message can then decrypt the message using their related private key. If the sender encrypts the message using their private key, the message can be decrypted only using that sender's public key, thus authenticating the sender. These encryption and decryption processes happen automatically; users do not need to physically lock and unlock the message. Many protocols rely on asymmetric cryptography, including the transport layer security (TLS) and secure sockets layer (SSL) protocols, which make HTTPS possible.

Hybrid Encryption Process:

1.Key Exchange: The process begins with the sender and receiver establishing communication. They don't need to share any secret keys beforehand.

2.Session Key Generation: The sender generates a random symmetric key, often referred to as a session key. This key is unique to the current session and is used only for encrypting the specific data being transmitted.

3.Symmetric Encryption: The sender encrypts the data using the symmetric session key. Since symmetric encryption is fast, it's ideal for encrypting large amounts of data.

4.Asymmetric Encryption: The sender then encrypts the session key using the recipient's public key. This encrypted session key is sent along with the encrypted data.

5.Transmission: The encrypted data and the encrypted session key are transmitted to the receiver.

6.Decryption: Upon receiving the encrypted data and session key, the recipient first uses their private key to decrypt the session key.

7.Symmetric Decryption: With the session key decrypted, the recipient can now use it to decrypt the actual data encrypted using symmetric encryption.

Asymmetric encryption uses a mathematically related pair of keys for encryption and decryption: a public key and a private key. If the public key is used for encryption, then the related private key is used for decryption. If the private key is used for encryption, then the related public key is used for decryption. The two participants in the asymmetric encryption

workflow are the sender and the receiver. Each has its own pair of public and private keys. First, the sender obtains the receiver's public key. Next, the plaintext message is encrypted by the sender using the receiver's public key. This creates ciphertext. The ciphertext is sent to the receiver, who decrypts it with their private key, returning it to legible plaintext. Because of the one-way nature of the encryption function, one sender is unable to read the messages of another sender, even though each has the public key of the receiver. Asymmetric cryptography is typically used to authenticate data using digital signatures. A digital signature is a mathematical technique used to validate the authenticity

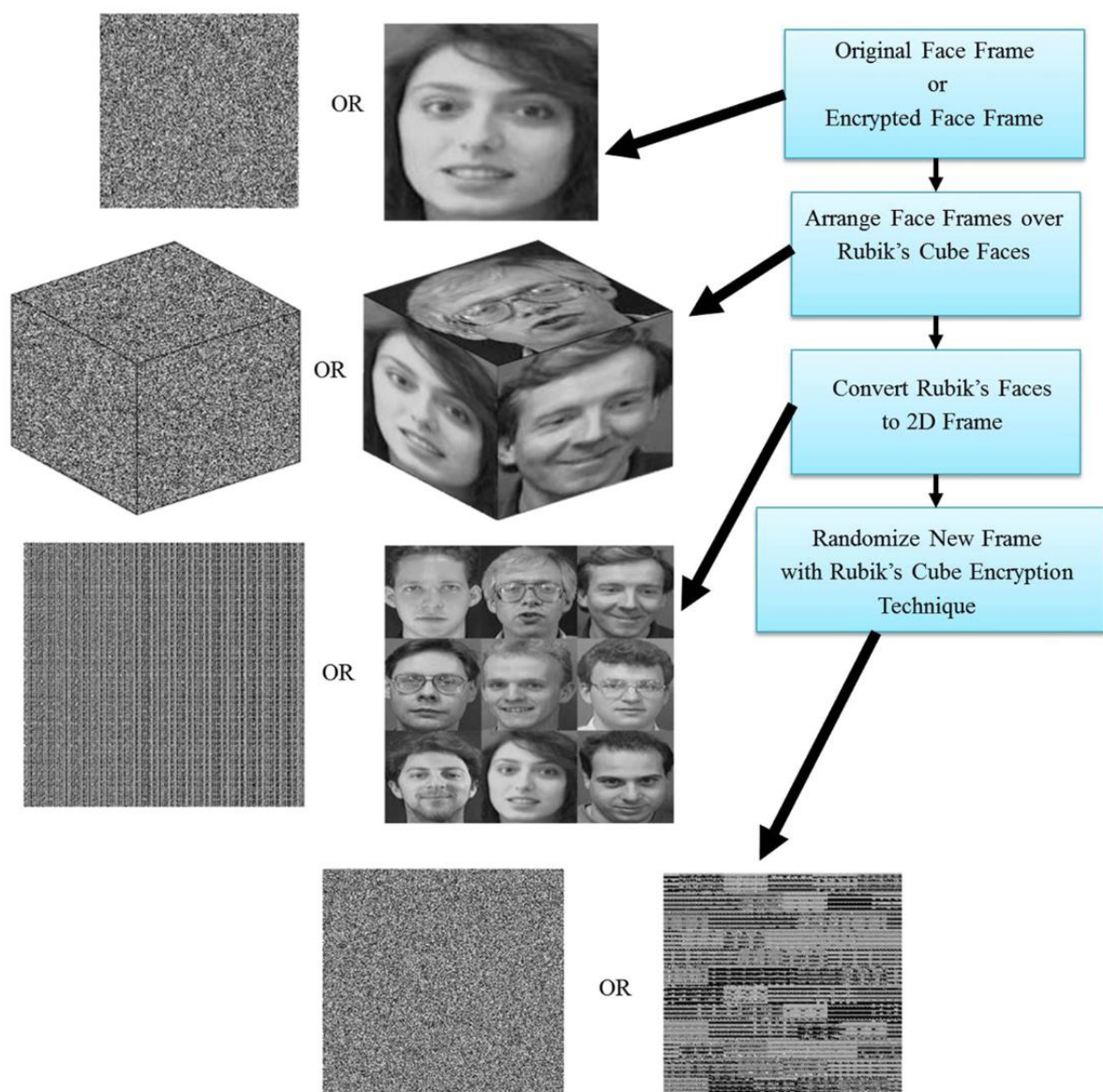


Fig: 4.10. The proposed hybrid enrollment stage

A public key and Private keys are generated randomly using an algorithm, and the keys have a mathematical relationship with each other. The key should be longer in length (128 bits, 256 bits) to make it stronger and make it impossible to break the key even if the other paired key is known. The number of possible keys increases proportionally with the key length, and hence cracking it also becomes tougher. The public key is open to everyone. Anyone can access it and encrypt data with it. However, once encrypted, that data can only be unlocked by using the corresponding private key. As you can imagine, the private key must be kept secret to keep it from becoming compromised.

Advantages of Hybrid Encryption:

Security: Combining symmetric and asymmetric encryption provides a balance between security and efficiency. Asymmetric encryption ensures secure key exchange, while symmetric encryption is efficient for encrypting data.

Efficiency: Symmetric encryption is faster than asymmetric encryption, making it suitable for encrypting large volumes of data. Hybrid encryption leverages the strengths of both encryption types to optimize performance.

Flexibility: Hybrid encryption allows for flexible key management. The symmetric session key is generated for each session, adding an extra layer of security, while the asymmetric keys remain constant, simplifying key management.

In summary, hybrid encryption offers a robust and efficient solution for secure data transmission by combining the strengths of symmetric and asymmetric encryption techniques.

Asymmetric cryptography, which can also be called public-key cryptography, uses private and public keys to encrypt and decrypt the data. The keys are simply large numbers that are paired together; however, they are asymmetric means not identical. Public Key is One of the keys in the pair that can be shared with everyone, whereas Private key is the other key in the pair that is kept secret; it is called the private key. Any key can be used to encrypt a message then the other key, which is not used for encryption, is used for decryption. When we talk about encryption, it's much like a lock on a door. To operate the lock, you need a key. This means only people who hold the key can unlock the door and gain access to whatever it's protecting. In the digital world, a key can come in many forms a is

open to everyone. Anyone can access it and encrypt data with it. However, once encrypted, that data can only be unlocked by using the corresponding private key. As you can imagine, the private key must be kept secret to keep it from becoming compromised. So, only the authorized person, server, machine, or instrument has access to the private key.

What you may not realize is that you're actually using public key encryption right now! Do you see the security padlock icon in your browser or the HTTPS in the website URL? Both of those things indicate you've connected to a website that uses SSL/TLS certificates and the secure TLS protocol. What this does is use asymmetric key encryption to verify the identity of the server and to create symmetric session keys. (Once this is done, your browser and the web server switch to using symmetric encryption for the rest of the session.)

4.1.2 HYBRID DECRYPTION PROCESS

Hybrid decryption is the process of decrypting data that has been encrypted using a hybrid encryption scheme, which combines symmetric and asymmetric encryption techniques. Here's a detailed explanation of how hybrid decryption works:

Receiving Encrypted Data:

The recipient receives the encrypted data along with the encrypted session key. The encrypted data is typically encrypted using symmetric encryption, while the session key is encrypted using asymmetric encryption (recipient's public key).

1.Using Private Key:

The recipient uses their private key, which is kept secret, to decrypt the encrypted session key. Asymmetric encryption ensures that only the recipient possessing the corresponding private key can decrypt the session key.

2.Obtaining Session Key:

Once the session key is decrypted, the recipient obtains the symmetric session key that was originally used by the sender to encrypt the data. This session key is unique to the current communication session.

3.Symmetric Decryption:

With the session key obtained, the recipient now uses it to decrypt the encrypted data. Symmetric decryption is used because it's faster and more efficient than asymmetric decryption, especially for decrypting large volumes of data.

4.Revealing Plain Text:

The recipient now has access to the decrypted data, revealing the original plaintext message that was encrypted by the sender.

Advantages of Hybrid Decryption:

Security: Hybrid decryption ensures security by using asymmetric encryption for key exchange, which eliminates the need to transmit sensitive symmetric keys over insecure channels. Symmetric encryption is then used for efficient bulk data encryption.

Efficiency: By utilizing symmetric encryption for data encryption and decryption, hybrid decryption achieves efficiency in processing large volumes of data. Asymmetric encryption is only used for the secure exchange of session keys.

Flexibility: Hybrid decryption allows for flexible key management. The symmetric session keys are generated for each session, providing an additional layer of security, while the asymmetric keys remain constant, simplifying key management. In summary, hybrid decryption combines the strengths of asymmetric and symmetric encryption techniques to securely decrypt data while optimizing efficiency. It ensures secure communication by leveraging asymmetric.

If a cancellable biometric is conceded, the alteration features are changed, and the same biometrics is associated to a new outline, which is consequently used. The primary idea behind cancelable biometrics is to store the original biometric safe using an alternate adaption of the original biometric pattern created by using a one-way transformation of passwords and keys. For instance, the shortfalls of both are that ID cards can easily be lost or forged, while passwords and keys can either be easily guessed or forgotten, respectively.

4.2 ARCHITECTURE / OVERALL DESIGN OF PROPOSED SYSTEM

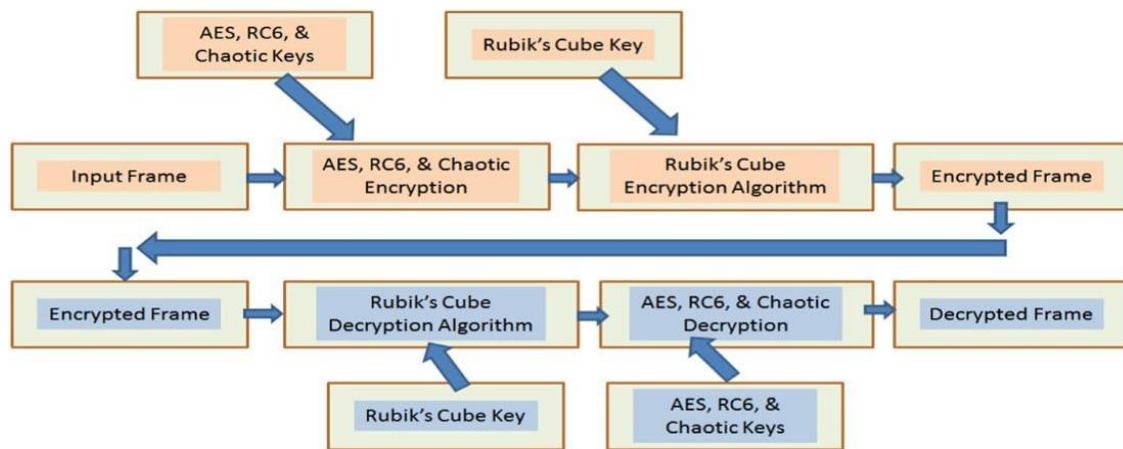


Fig:4.11. Proposed framework implementation

- Encrypt the faces with chaotic, AES, or RC6 algorithm.
- Set the encrypted faces over the Rubik's cube faces.
- Convert all faces into a 2D image with a larger size.
- Randomize the converted 2D image with Rubik's cube technique.

Proposed hybrid enrollment framework:

We use in this paper the Rubik's cube technique with different random keys to randomize the pixels of the original gray-scale images, by changing the positions of the pixels within the encrypted images via the circular-shift step applied on each of the columns and rows. Then, columns and rows are XOR-ed with the secret keys, and these stages can be repeated to improve the security strength of this algorithm.

Why do people need several security and privacy concerns with biometrics? The answer of this question can be summarized as follows:

Traditional identity authentication methods are based on what is physically possessed, such as ID cards, and what can be mentally stored in the memory.

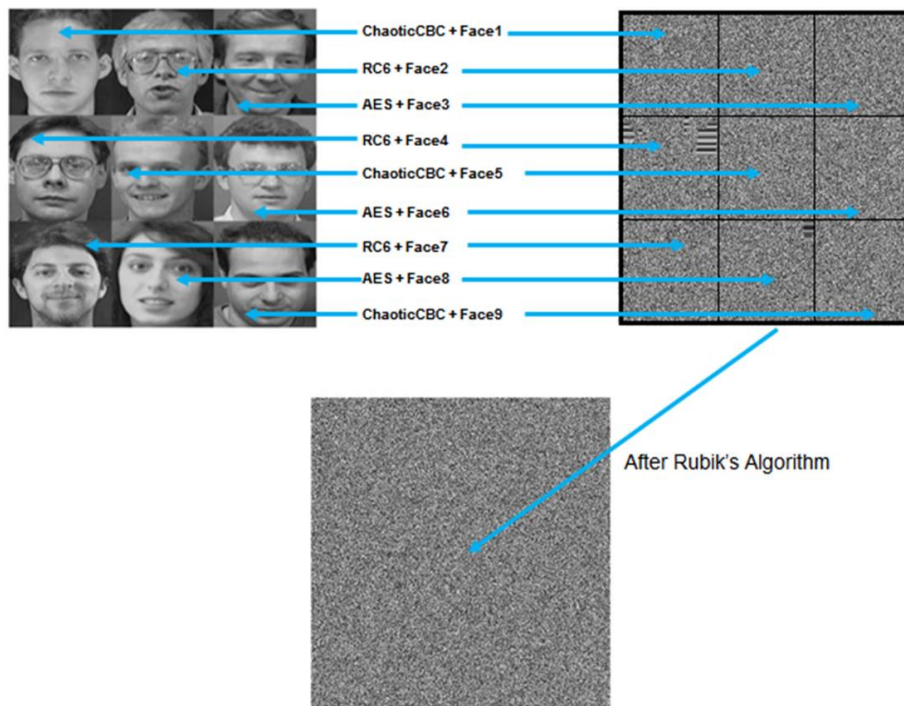


Fig: 4.12. Three degrees of encryption.

The proposed encryption scheme is based on the chaotic Baker's map, Rubik's cube, AES, and RC6 algorithms. This is used to concurrently encrypt a collection of photos. The chaotic-AES, RC6 encryption method is used to protect the images in this collection. The proposed architecture will guarantee that the results of the encryption will be both permuted and diffused. As a result, the suggested biometric standard built on the Rubik's cube algorithm may achieve respectable encryption and provide protection against various threats. As a result, a cancelable biometric system was used. When biometrics are intentionally and thoroughly altered to protect user-specific data, this is referred to as cancelable biometrics. If a cancellable biometric is conceded, the alteration features are changed, and the same biometrics is associated to a new outline, which is consequently used. The primary idea behind cancelable biometrics is to store the original biometric safe using an alternate adaption of the original biometric pattern created by using a one-way transformation of passwords and keys. For instance, the shortfalls of both are that ID cards can easily be lost or forged, while passwords and keys can either be easily guessed or forgotten, respectively. Short passwords are often preferred for memory requirements, but easily guessed by others. On the other hand, long passwords (commonly known as keys).

Asymmetric cryptography, which can also be called public-key cryptography, uses private and public keys to encrypt and decrypt the data. The keys are simply large numbers that are paired together; however, they are asymmetric means not identical. Public Key is One of the keys in the pair that can be shared with everyone, whereas Private key is the other key in the pair that is kept secret; it is called the private key. Any key can be used to encrypt a message then the other key, which is not used for encryption, is used for decryption. When we talk about encryption, it's much like a lock on a door. To operate the lock, you need a key. This means only people who hold the key can unlock the door and gain access to whatever it's protecting. In the digital world, a key can come in many forms a password, code, PIN, or a complex string of computer-generated characters. The public key is open to everyone. Anyone can access it and encrypt data with it. However, once encrypted, that data can only be unlocked by using the corresponding private key. As you can imagine, the private key must be kept secret to keep it from becoming compromised. So, only the authorized person.

It allows simultaneous encryption of multiple images. The suggested hybrid enrollment stage begins with chaotic Baker map permutation in the Cipher FeedBack (CFB) operation mode, Advanced Encryption Standard (AES) or Ron's Code (RC6) algorithm as a first stage for encrypting the multiple images, separately. After that, the output encrypted images are passed to the second encryption stage via Rubik's cube technique. Chaotic, RC6, or AES encrypted face images are utilized as the Rubik's cube faces. For the RC6 and AES algorithms, they add a degree of diffusion, while the chaotic algorithm adds a degree of permutation. Moreover, the Rubik's cube technique adds more permutations to the encrypted images, simultaneously. These biometrics can be changed easily in emergency cases without changing the system at all. Moreover, encryption techniques can be used with all types of biometrics to generate cancelable templates.

It has been the most commonly used mode of operation. Its main drawbacks are that encryption is sequential (i.e., it cannot be parallelized), and that the message must be padded to a multiple of the cipher block size. One way to handle this last issue is through the method known as cipher text stealing. Note that a one-bit change in a plaintext or IV affects all following cipher text blocks. Decrypting with the incorrect initialization vector causes the first block of plaintext to be corrupted correctly.

4.3. PROJECT MANAGEMENT PLAN:

August	Literature survey
September	Data acquisition
October	Data processing
Novemeber	Implemation of code
December	Dataset implementation
March	Implementing the encrypted output final report

CHAPTER 5

IMPLEMENTATION DETAILS

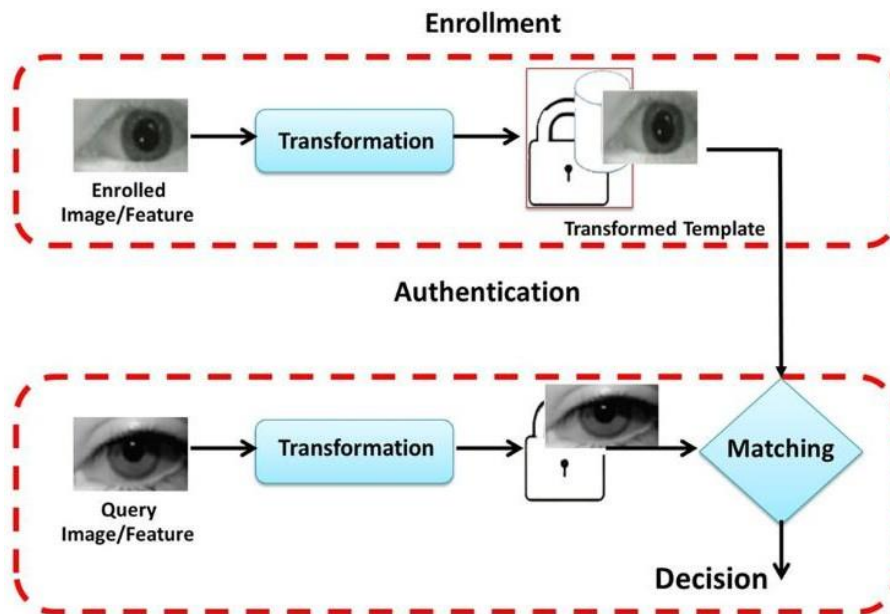


Fig:5.1. Cancelable Biometric System

- Python modules used:
- Numpy - NumPy is a Python library used for working with arrays
- Pillow - PIL adds image processing capabilities
- Matplotlib.pyplot – Python library used for visualization of data, using graphs, charts, plots.
- Pycryptodome - Python package of various cryptographic algorithms.
- opencv - Python library for solving computer vision problems

- Sum of each row is calculated.
- Applying modulus of 2 on the calculated sum.
- If the sum of the row is equal to '0', right circular shift is performed.
- Sum of each column is calculated.
- Applying modulus of 2 on the calculated sum.
- If the sum of the row is equal to '0', up circular shift is performed.
- Performed for all rows and columns.

However, the user's biometrics are limited and unique, and if a user biometric is compromised, it will be impossible to change it in a particular system or at least difficult. Cancelable biometrics can be generated with intentional, repeatable distortions of biometric signals based on transforms. Hence, biometric templates are compared in the transform domain. Cancelable biometrics depend on transforming data to replace a single biometric template in the same or different systems. This paper presents a cancelable biometric system that depends on a novel hybrid encryption framework based on the Rubik' cube technique. It allows simultaneous encryption of multiple images. The suggested hybrid enrollment stage begins with chaotic Baker map permutation in the Cipher FeedBack (CFB) operation mode, Advanced Encryption Standard (AES) or Ron's Code (RC6) algorithm as a first stage for encrypting the multiple images, separately. After that, the output encrypted images are passed to the second encryption stage via Rubik's cube technique. Chaotic, RC6, or AES encrypted face images are utilized as the Rubik's cube faces. For the RC6 and AES algorithms, they add a degree of diffusion, while the chaotic algorithm adds a degree of permutation. Moreover, the Rubik's cube technique adds more permutations to the encrypted images, simultaneously. These biometrics can be changed easily in emergency cases without changing the system at all. Moreover, encryption techniques can be used with all types of biometrics to generate cancelable templates. Different algorithms have been presented in the literature due to the importance and popularity of the recognition systems. Recent cancelable biometric concepts have been adopted for face, fingerprint, and iris recognition by using random kernels to generate cancelable image templates.

CHAPTER 6

RESULTS AND DISCUSSION

6.1 Input Design

A hybrid Rubik's Cube-based encryption framework for cancelable biometrics combines the versatility of Rubik's Cube manipulation with the security of cancelable biometric techniques to create a robust encryption system. In this framework, biometric features such as fingerprints, iris patterns, or facial characteristics are transformed into a unique representation that can be used to manipulate a virtual Rubik's Cube. Here's an overview of the input design for this innovative encryption framework:

Biometric Data Conversion: The first step involves capturing biometric data from the user, such as fingerprints or iris scans. This data is then processed and converted into a unique numerical representation using feature extraction techniques. This representation serves as the basis for generating the initial configuration of the virtual Rubik's Cube.

Rubik's Cube Configuration: The numerical representation derived from the biometric data is used to configure the initial state of the virtual Rubik's Cube. Each element of the numerical representation determines the position or orientation of specific cubelets within the Rubik's Cube. This configuration serves as the encryption key for the data.

Cube Manipulation: To encrypt or decrypt data, users manipulate the virtual Rubik's Cube according to a predefined algorithm. The algorithm specifies a sequence of rotations and movements that alter the configuration of the cube in a controlled manner. The specific sequence of movements depends on the encryption key derived from the user's biometric data.

6.2 Output Design

Our hybrid Rubik's Cube-based encryption framework for cancelable biometrics leverages the complexity and randomness inherent in the Rubik's Cube puzzle to create a robust and secure method for encrypting biometric data. The framework combines the unique features of Rubik's Cube manipulation with advanced cryptographic techniques to ensure both security and privacy in biometric authentication systems.

At the core of our framework is the transformation of biometric data into a representation that can be mapped onto a Rubik's Cube. This transformation process involves encoding biometric features into a format suitable for manipulation on the Rubik's Cube, ensuring that the resulting encryption maintains the distinctiveness of the biometric data while also rendering it cancelable and unlinkable to the original biometric sample.

Once the biometric data is transformed, it undergoes encryption through a combination of Rubik's Cube algorithms and cryptographic methods. The Rubik's Cube algorithms introduce additional layers of randomness and complexity, making it extremely difficult for unauthorized parties to decipher the encrypted biometric data without the proper decryption key. Meanwhile, the cryptographic techniques further enhance security by applying encryption schemes such as AES or RSA to protect the Rubik's Cube-based encryption keys.

6.3 System Design

The hybrid Rubik's Cube-based encryption framework for cancelable biometrics is a sophisticated system designed to enhance security in biometric authentication while preserving user privacy through cancelability techniques.

At its core, the system employs a combination of Rubik's Cube algorithms and biometric data to generate unique encryption keys. The biometric data, such as fingerprints or iris scans, undergoes cancelable transformation processes to produce multiple cancelable templates. These templates are then mapped onto different configurations of a virtual Rubik's Cube, creating a diverse set of encryption keys.

The framework consists of several key components:

Biometric Data Acquisition: Users provide their biometric data, which is then processed and transformed into cancelable templates using techniques like random projection or feature transformation.

Encryption Key Generation: The Rubik's Cube configurations serve as the basis for generating encryption keys. The cube configurations are transformed into cryptographic keys using established algorithms, ensuring strong encryption.

CHAPTER 7

CONCLUSION AND FUTURE WORK

In this work, a hybrid encryption architecture for cancellable biometric devices was created with increased security. This study's main contribution is the addition of the Rubik's cube encryption algorithm to the hybrid architecture that also uses the chaotic baker's map, RC6, and AES encryption techniques.

The project's proposed hybrid encryption technique adds more permutations and diffusions to the recorded biometric data. The positive achievements of the proposed hybrid architecture in securely storing the encryption of biometric images were validated by simulation results.

In the case of using all methods simultaneously, parallel processing can dramatically shorten the calculation durations of the conventional AES, RC6, and chaotic baker's map encryption algorithms:

- Studying different instances of biometric cryptosystems of techniques using Rubik's cube with various keys.
- Usage of asymmetric algorithms.

REFERENCES

- [1] William, P., Shrivastava, A., Shunmuga Karpagam, N., Mohanaprakash, T. A., Tongkachok, K., & Kumar, K. (2023). Crime analysis using computer vision approach with machine learning. In *Mobile Radio Communications and 5G Networks: Proceedings of Third MRCN 2022* (pp. 297-315). Singapore: Springer Nature Singapore.
- [2] Dakalbab, F., Talib, M. A., Waraga, O. A., Nassif, A. B., Abbas, S., & Nasir, Q. (2022). Artificial intelligence & crime prediction: A systematic literature review. *Social Sciences & Humanities Open*, 6(1), 100342.
- [3] Srinivasulu Raju, S., Narasimha Swamy, G., Rejoice Angelina, M., Sai Snehitha, M., Sai Chandana, M., & Priya Mythili, M. (2022). Analysis and Prediction of Crime Using Machine Learning Techniques. In *Smart and Intelligent Systems: Proceedings of SIS 2021* (pp. 343-355). Springer Singapore.
- [4] Lombo, X., Oyelade, O. N., & Ezugwu, A. E. (2022, July). Crime Detection and Analysis from Social Media Messages Using Machine Learning and Natural Language Processing Technique. In *International Conference on Computational Science and Its Applications* (pp. 502-517). Cham: Springer International Publishing.
- [5] Norouzi, Y. (2022, May). Spatial, Temporal, and Semantic Crime Analysis Using Information Extraction From Online News. In *2022 8th International Conference on Web Research (ICWR)* (pp. 40-46). IEEE.
- [6] SH, A. S., Pathmanaban, J., Kingsley, S., & Sriman, B. (2022, November). Prediction and Prevention Analysis Using Machine Learning Algorithms for Detecting the Crime Data. In *2022 1st International Conference on Computational Science and Technology (ICCST)* (pp. 986-991). IEEE.
- [7] Travaini, G. V., Pacchioni, F., Bellumore, S., Bosia, M., & De Micco, F. (2022). Machine learning and criminal justice: A systematic review of advanced methodology for recidivism risk prediction. *International journal of environmental research and public health*, 19(17), 10594.

- [8] Adhikary, A., Murad, S. A., Munir, M. S., & Hong, C. S. (2022, January). Edgeassisted crime prediction and evaluation framework for machine learning algorithms. In 2022 International Conference on Information Networking (ICOIN) (pp. 417- 422). IEEE.
- [9] Kshatri, S., Bhonsle, D., Verma, R., Pillai, A., & Moyal, V. (2022). Crime Detection Approach Using Big Data Analytics and Machine Learning. *NeuroQuantology*, 20(8), 1480-1495.
- [10] Gandapur, M. Q. (2022). E2E-VSDL: End-to-end video surveillance-based deep learning model to detect and prevent criminal activities. *Image and Vision Computing*, 123, 104467.

APPENDIX

A. SOURCE CODE

Kernel

```
import glob
import cv2 as
c import time
start=time.time
e()
def kernal(image, prob):
    op = c.GaussianBlur(image, (51, 51),
    0)return op
k = "true1.jpg"
for file in glob.glob(k):
    print(file)
    img = c.imread(file,1)
    out=kernal(img,.01)
    c.imshow("blurred image",out)
    c.imwrite("kernal_true1.jpg",out)
end=time.time()
print("Execution time:",end-start)
```

RC6-Rubik's Cube Technique

```
import base64 as
b6import math
import cv2 as c
from Crypto.Util.Padding
import * from PIL import
BmpImagePluginimport time
from PIL import
Imageimport numpy
as np import json
start =
time.time()
class RC6:
    def ToXor(self, *Args):
        lgt = len(bin(max(Args))[2:])
        Args = [self.binExp(bin(arg), lgt)[2:] for arg in Args]

    op = '0b'
    for x in range(lgt):
        counter = 0
        for arg in Args:
```

```

        counter      +=
    int(arg[x])  op    +=
    str(counter % 2)
    return int(op, 2)
def binExp(self, bit_string,
length):output = bit_string
    for i in range(1, r):
        t = self.SCircular((B * ((2 * B) % mod + 1) % mod) % mod, w, int(math.log(w)),
'left') u = self.SCircular((D * ((2 * D) % mod + 1) % mod) % mod, w,
int(math.log(w)), 'left')A = (self.SCircular(self.ToXor(A, t), w, u, 'left') + S[2 * i]) %
mod
        C = (self.SCircular(self.ToXor(C, u), w, t, 'left') + S[2 * i + 1]) %
modaa, bb, cc, dd = B, C, D, A
        A, B, C, D = aa, bb, cc, dd
        A = (A + S[2 * r + 2]) %
modC = (C + S[2 * r + 3])
        % mod
        op = "
        op += self.binExp(bin(A), w)[2:]
        op += self.binExp(bin(B), w)[2:]
        op += self.binExp(bin(C), w)[2:]
        op += self.binExp(bin(D),
w)[2:]return (op)
        for block in msg :
            output += self.binEncryption(block,
w, r)return output
Original = c.imread("true1.jpg")
Original_row, Original_col, Original_dep =
Original.shape# Display original image
c.imshow("Original_image", Original)
c.waitKey()
c.imwrite("tooriginalimg.bmp",
Original) Original =
c.imread("tooriginalimg.bmp")
Original =
BmpImagePlugin.BmpImageFile("tooriginalimg.bmp")#
Convert original image data to bytes
Original_bytes = Original.tobytes()
A, B, C, D = aa, bb, cc, dd
        A = (A + S[2 * r + 2]) %
modC = (C + S[2 * r + 3])
        % mod
        op = "
        op += self.binExp(bin(A), w)[2:]
        op += self.binExp(bin(B), w)[2:]
        op += self.binExp(bin(C), w)[2:]

```

```

while len(output) != length + 2:
    output = output[:2] + '0' + output[2:]
return output
def SCircular(self, number, w, bits, side):
    binString = self.binExp(bin(number), w)
    bits %= w
    binString = binString[2:]
    if side == 'left':
        return int('0b' + binString[bits:] + binString[:bits], 2)
    if side == 'right':
        return int('0b' + binString[:-bits] + binString[bits:], 2)
def keysTableGen(self, key, w = 32, r = 20):
    mod = 2 ** w
    while len(key) % w != 0:
        key = key + '0'
    c = int(len(key) / w) # no. of words in a key
    L = [key[i * w : (i + 1) * w] for i in range(c)]
    L = [int('0b' + k, 2) for k in L]
    def odd(number):
        """Rounding to the nearest odd integer"""
        if int(number) % 2 != 0: return int(number)
        else: return int(number) + 1
    f = (math.sqrt(5) + 1) / 2 # golden ratio
    Qw = odd((f - 1) * 2 ** w)
    Pw = odd((math.e - 2) * 2 ** w)
    S = []
    S.append(Pw)
    for p in range(1, 2 * r + 4):
        S.append((S[p - 1] + Qw) % mod)
    A = B = i = j = 0
    V1 = 3 * max(c, 2 * r + 4)
    for q in range(1, V1):
        A = S[i] = self.SCircular((S[i] + A + B) % mod, w, 3, 'left')
        B = L[j] = self.SCircular((L[j] + A + B) % mod, w, (A + B) % mod, 'left')
        i = (i + 1) % (2 * r + 4)
        j = (j + 1) % c
    self.keysTable = S
    return S
def binEncryption(self, msg, w = 32, r = 20):
    mod = 2 ** w
    S = self.keysTable
    A = int('0b' + msg[0:w], 2)
    B = int('0b' + msg[(w):(2 * w)], 2)
    C = int('0b' + msg[(2 * w):(3 * w)], 2)
    D = int('0b' + msg[(3 * w):(4 * w)], 2)

    B = (B + S[0]) % mod
    D = (D + S[1]) % mod

```

```

for i in range(1, r):
    t = self.SCircular((B * ((2 * B) % mod + 1) % mod) % mod, w, int(math.log(w)), 'left')
    u = self.SCircular((D * ((2 * D) % mod + 1) % mod) % mod, w, int(math.log(w)), 'left')
    A = (self.SCircular(self.ToXor(A, t), w, u, 'left') + S[2 * i]) % mod
    C = (self.SCircular(self.ToXor(C, u), w, t, 'left') + S[2 * i + 1]) % mod
    aa, bb, cc, dd = B, C, D, A
    A, B, C, D = aa, bb, cc, dd
A = (A + S[2 * r + 2]) % mod
C = (C + S[2 * r + 3]) % mod
op = ""
op += self.binExp(bin(A), w)[2:]
op += self.binExp(bin(B), w)[2:]
op += self.binExp(bin(C), w)[2:]
op += self.binExp(bin(D), w)[2:]
return (op)
def ConvBytesBin(self, bytes):
    op = bytearray(bytes)
    op = [self.binExp(bin(char), 8)[2:] for char in op]
    op = "".join(op)
    return op
def ConvBinBytes(self, bin):
    output = [int('0b' + bin[block * 8 : (block + 1) * 8], 2) for block in range(int(len(bin) / 8))]
    output = bytes(output)
    return output
def encryption(self, msg, key, w = 32, r = 20):
    self.keysTableGen(key, w = 32, r = 20)
    size = len(msg)
    size = self.binExp(bin(size), 64)
    msg = size[2:] + msg
    while len(msg) % (4 * w) != 0:
        msg += '0'
    msg = [msg[(block * 4 * w) : ((block + 1) * 4 * w)] for block in range(int(len(msg) / (4 * w)))]
    output = ""
    for block in msg :
        output += self.binEncryption(block, w, r)
    return output
Original = c.imread("true1.jpg")
Original_row, Original_col, Original_dep = Original.shape
# Display original image
c.imshow("Original_image", Original)
c.waitKey()
c.imwrite("tooriginalimg.bmp", Original)
Original = c.imread("tooriginalimg.bmp")
Original = BmpImagePlugin.BmpImageFile("tooriginalimg.bmp")
# Convert original image data to bytes
Original_bytes = Original.tobytes()
k = b6.b64encode(bytes("Rc6encrypt", 'utf-8'))
rc = RC6()
msgBin = rc.ConvBytesBin(Original_bytes)

```

```

keyBin = rc.ConvBytesBin(k)
encrpytedBin = rc.encryption(msgBin, keyBin)
encryptedBytes = rc.ConvBinBytes(encrpytedBin)
# Convert bytes to encrypted image data
opimage = Original.copy()
opimage.frombytes(encryptedBytes)
opimage.save("rc6.bmp")

```

```

#Rubik's Cube :
myImage = Image.open("rc6.bmp");
width, height = myImage.size
pixels = myImage.load()
has_alpha = len(pixels[0,0]) == 4
fill = 1
array = [[fill for x1 in range(width)] for y1 in range(height)]
for y2 in range(height):
    for x2 in range(width):
        if has_alpha:
            r, g, b, a = pixels[x2,y2]
        else:
            r, g, b = pixels[x2,y2]
        array[y2][x2] = r

```

```

KR = [40, 6, 227, 45, 19, 236, 8, 163, 50, 252, 21, 216, 143, 170, 114, 134, 46, 4, 185, 3, 34, 247,175,
253,137, 111, 177, 254, 166, 186, 77, 183, 11, 31, 244, 169, 162, 137, 248,252,63,221,100,7, 28, 152,
228, 5,82, 212, 164, 133, 124, 252, 55, 187, 64, 85, 1, 44, 156, 86, 252, 91, 224, 34, 70, 73, 231, 119,
216, 211, 147,5, 221, 96, 73, 63, 147, 126, 117, 80, 134, 130, 29, 56, 180, 201, 157, 28, 207, 233, 10,
147, 239, 73, 57, 0,214, 33, 32, 150, 72, 127, 119, 230, 102, 169, 232, 10, 111, 217, 235, 132, 78, 197,
53, 142, 94, 79, 16, 240,217, 149, 117, 188, 106, 120, 175, 98, 170, 181, 93, 45, 101, 4, 47, 236, 43,
41, 251, 52, 124, 202, 24, 76,144, 54, 58, 221, 184, 35, 177, 146, 42, 139, 232, 13, 51, 233, 43, 83,
241, 236, 48, 215, 26, 86, 20, 70, 34,217, 212, 4, 181, 45, 17, 121, 91, 202, 155, 240, 141, 86, 141,
192, 178, 175, 149, 173, 223, 116, 226, 87, 76,133, 185, 219, 64, 105, 245, 238, 251, 46, 100, 160,
134, 126, 229, 29, 199, 92, 21, 145, 61, 67, 89, 21, 127,104,243, 183, 215, 220, 148, 6, 245, 230, 178,
195, 19, 192, 136, 99, 27, 157, 189, 108, 54, 107,41]

```

```

KC = [197, 98, 234, 162, 220, 198, 187, 211, 1, 224, 140, 104, 191, 7, 157, 210, 191, 134, 10, 210,
218, 22, 56, 61, 196, 242, 21, 89, 226, 254, 3, 229, 213, 83, 242, 92, 122, 242, 227, 115, 92, 27, 131,
50, 178, 158, 29, 127, 19, 12, 189, 16, 35, 220, 97, 249, 191, 190, 220, 126, 32, 215, 231, 224, 94, 7,
120, 215, 63, 220, 243, 111, 110, 147, 230, 43, 240, 54, 97, 17, 254, 98, 105, 110, 132, 130, 58, 118,
217, 73, 175, 164, 18, 140, 23, 187, 32, 189, 169, 23, 156, 246, 234, 236, 204, 5, 222, 92, 248, 172,
44, 246, 208, 68, 159, 15, 211, 182, 44, 215, 237, 27, 23, 221, 85, 253, 103, 24, 31, 8, 46, 217, 56,
163, 28, 137, 131, 75, 203, 202, 12, 117, 150, 87, 121, 168, 129, 164, 70, 222, 30, 157, 215, 13, 108,
77, 108, 242, 228, 167, 192, 17, 213, 34, 75, 56, 25, 89, 230, 125, 169, 159, 216, 139, 158, 46, 33,
254, 67, 239, 54, 159, 175, 101, 41, 221, 61, 164, 66, 41, 113, 12, 7, 114, 254, 52, 40, 0, 180, 25, 184,
43, 84, 73, 65, 195, 182, 61, 35, 250, 81, 67, 52, 215, 255, 176, 112, 22, 208, 108, 172, 98, 30, 238,
245, 233, 57, 83, 121, 48, 109, 150, 130, 170, 88, 121, 198, 234, 95, 59,81]

```

```

for i in range(height):
    alpha1 = 0
    for j in range(width):
        alpha1 = ((alpha1%2) + (array[i][j] % 2)) % 2

```

```

if(alpha1 == 0):
    for k in range(KR[i]):
        temp2 = array[i][width-1]
        for l in range(width-1, -1, -1):
            array[i][l] = array[i][l-1];
        array[i][0] = temp2;
    else:
        for k in range(KR[i]):
            temp2 = array[i][0]
            for l in range(width-1):
                array[i][l] = array[i][l+1];
            array[i][width-1] = temp2;
for j in range(width):
    beta = 0
    for i in range(height):
        beta = ((beta%2) + (array[i][j] % 2)) % 2
    if(beta == 0):
        for k in range(KC[j]):
            temp2 = array[height - 1][j]
            for l in range(height-1, -1, -1):
                array[l][j] = array[l-1][j];
            array[0][j] = temp2;
    else:
        for k in range(KC[j]):
            temp2 = array[0][j]
            for l in range(height-1):
                array[l][j] = array[l+1][j];
            array[height-1][j] = temp2;
for j2 in range(width):
    for i2 in range(height):
        if((i2%2) != 0 ):
            array[i2][j2] = array[i2][j2]^KC[j2]
        else:
            array[i2][j2] = array[i2][j2]^KC[width-1-j2]
for i in range(height):
    for j in range(width):
        if((j%2) != 0 ):
            array[i][j] = array[i][j]^KR[i]
        else:
            array[i][j] = array[i][j]^KR[height-1-i]
array1 = np.array(array, dtype=np.uint8)
new_image = Image.fromarray(array1)
new_image.save("rc6_rubik'scube_encrypted_true1.jpg")
with open("D:\\KR.txt", "w") as outfile:
    json.dump(KR, outfile)
with open("D:\\KC.txt", "w") as outfile:
    json.dump(KC, outfile)
# Close all windows
c.destroyAllWindows()

```

```

end=time.time()
print("Execution Time -> ", end-start)

```

Chaotic CBC – Rubik’s Cube Technique

```

import sys as s
import cv2 as c
import numpy as np
from Crypto.Cipher import AES
from Crypto.Util.Padding import *
#import Crypto.Random.get_random_bytes as get_random_byte
import PIL.Image as Image
import json
import time
start = time.time()
def bakers(a,b,c):
    x=a
    y=b
    z=c
    if a>=0.0 and a<0.5 and b>=0 and b<0.5:
        x=2*a
        y=2*b
        z=c/2
    elif a>=0 and a<0.5 and b>=0.5 and b<=1:
        x=a*2
        y=(b*2)-1
        z=(c/4)+0.5
    elif a>=0.5 and a<=1 and b>=0 and b<0.5:
        x=(2*a)-1
        y=b*2
        z=(c/4)+0.25
    elif a>=0.5 and a<=1 and b>=0.5 and b<=1:
        x=(2*a)-1
        y=(b*2)-1
        z=(c/4)+0.75
    l=int(x*255)
    m=int(y*255)
    n=int(z*255)
    return l, m, n
myImage = Image.open("true1.jpg")
width, height = myImage.size
pixels = myImage.load()
has_alpha = len(pixels[0,0]) == 4
arr=[]
for y1 in range(height):
    for x1 in range(width):
        if has_alpha:
            r, g, b, a = pixels[x1,y1]
        else:

```

```

    r, g, b = pixels[x1,y1]
    arr.append(r)
for y2 in range(height):
    for x2 in range(width):
        if has_alpha:
            r, g, b, a = pixels[x2,y2]
            k=(255-r)/255
            l=(255-g)/255
            m=(255-b)/255
            r,g,b=bakers(k,l,m)
            value=(r,g,b,a)
            myImage.putpixel((int(x2), int(y2)), value)
        else:
            r, g, b = pixels[x2,y2]
            k=(255-r)/255
            l=(255-g)/255
            m=(255-b)/255
            r,g,b= bakers(k,l,m)
            value=(r,g,b)
            myImage.putpixel((int(x2), int(y2)), value)
myImage=myImage.convert('RGB')
myImage.save("chaotic.jpg")
#Rubik's Cube
myImage = Image.open("chaotic.jpg");
width, height = myImage.size
pixels = myImage.load()
print(pixels[0,0])
has_alpha = len(pixels[0,0]) == 4
fill = 1
array = [[fill for x in range(width)] for y in range(height)]
for y in range(height):
    for x in range(width):
        if has_alpha:
            r, g, b, a = pixels[x,y]
        else:
            r, g, b = pixels[x,y]
        array[y][x] = r
KR = [40, 6, 227, 45, 19, 236, 8, 163, 50, 252, 21, 216, 143, 170, 114, 134, 46, 4, 185, 3, 34, 247,175,
253,137, 111, 177, 254, 166, 186, 77, 183, 11, 31, 244, 169, 162, 137, 248,252,63,221,100,7, 28, 152,
228, 5,82, 212, 164, 133, 124, 252, 55, 187, 64, 85, 1, 44, 156, 86, 252, 91, 224, 34, 70, 73, 231, 119,
216, 211, 147,5, 221, 96, 73, 63, 147, 126, 117, 80, 134, 130, 29, 56, 180, 201, 157, 28, 207, 233, 10,
147, 239, 73, 57, 0,214, 33, 32, 150, 72, 127, 119, 230, 102, 169, 232, 10, 111, 217, 235, 132, 78, 197,
53, 142, 94, 79, 16, 240,217, 149, 117, 188, 106, 120, 175, 98, 170, 181, 93, 45, 101, 4, 47, 236, 43,
41, 251, 52, 124, 202, 24, 76,144, 54, 58, 221, 184, 35, 177, 146, 42, 139, 232, 13, 51, 233, 43, 83,
241, 236, 48, 215, 26, 86, 20, 70, 34,217, 212, 4, 181, 45, 17, 121, 91, 202, 155, 240, 141, 86, 141,
192, 178, 175, 149, 173, 223, 116, 226, 87, 76,133, 185, 219, 64, 105, 245, 238, 251, 46, 100, 160,
134, 126, 229, 29, 199, 92, 21, 145, 61, 67, 89, 21, 127,104,243, 183, 215, 220, 148, 6, 245, 230, 178,
195, 19, 192, 136, 99, 27, 157, 189, 108, 54, 107,41]
KC = [197, 98, 234, 162, 220, 198, 187, 211, 1, 224, 140, 104, 191, 7, 157, 210, 191, 134, 10, 210,

```

218, 22, 56, 61, 196, 242, 21, 89, 226, 254, 3, 229, 213, 83, 242, 92, 122, 242, 227, 115, 92, 27, 131, 50, 178, 158, 29, 127, 19, 12, 189, 16, 35, 220, 97, 249, 191, 190, 220, 126, 32, 215, 231, 224, 94, 7, 120, 215, 63, 220, 243, 111, 110, 147, 230, 43, 240, 54, 97, 17, 254, 98, 105, 110, 132, 130, 58, 118, 217, 73, 175, 164, 18, 140, 23, 187, 32, 189, 169, 23, 156, 246, 234, 236, 204, 5, 222, 92, 248, 172, 44, 246, 208, 68, 159, 15, 211, 182, 44, 215, 237, 27, 23, 221, 85, 253, 103, 24, 31, 8, 46, 217, 56, 163, 28, 137, 131, 75, 203, 202, 12, 117, 150, 87, 121, 168, 129, 164, 70, 222, 30, 157, 215, 13, 108, 77, 108, 242, 228, 167, 192, 17, 213, 34, 75, 56, 25, 89, 230, 125, 169, 159, 216, 139, 158, 46, 33, 254, 67, 239, 54, 159, 175, 101, 41, 221, 61, 164, 66, 41, 113, 12, 7, 114, 254, 52, 40, 0, 180, 25, 184, 43, 84, 73, 65, 195, 182, 61, 35, 250, 81, 67, 52, 215, 255, 176, 112, 22, 208, 108, 172, 98, 30, 238, 245, 233, 57, 83, 121, 48, 109, 150, 130, 170, 88, 121, 198, 234, 95, 59, 81]

```

for i in range(height):
    alpha = 0
    for j in range(width):
        alpha = ((alpha%2) + (array[i][j] % 2)) % 2
    if(alpha == 0):
        for k in range(KR[i]):
            temp2 = array[i][width-1]
            for l in range(width-1, -1, -1):
                array[i][l] = array[i][l-1];
            array[i][0] = temp2;
        else:
            for k in range(KR[i]):
                temp2 = array[i][0]
                for l in range(width-1):
                    array[i][l] = array[i][l+1];
                array[i][width-1] = temp2;
    for j in range(width):
        beta = 0
        for i in range(height):
            beta = ((beta%2) + (array[i][j] % 2)) % 2
        if(beta == 0):
            for k in range(KC[j]):
                temp2 = array[height - 1][j]
                for l in range(height-1, -1, -1):
                    array[l][j] = array[l-1][j];
                array[0][j] = temp2;
            else:
                for k in range(KC[j]):
                    temp2 = array[0][j]
                    for l in range(height-1):
                        array[l][j] = array[l+1][j];
                    array[height-1][j] = temp2;
    for j in range(width):
        for i in range(height):
            if((i%2) != 0 ):
                array[i][j] = array[i][j]^KC[j]
            else:
                array[i][j] = array[i][j]^KC[width-1-j]
    for i3 in range(height):

```

```

for j3 in range(width):
    if((j3%2) !=0 ):
        array[i3][j3] = array[i3][j3]^KR[i3]
    else:
        array[i3][j3] = array[i3][j3]^KR[height-1-i3]
array1 = np.array(array, dtype=np.uint8)
new_image = Image.fromarray(array1)
new_image.save("chaotic_rubik'scube_encrypted_true1.jpg")
with open("D:\\KR.txt", "w") as outfile:
    json.dump(KR, outfile)
with open("D:\\KC.txt", "w") as outfile:
    json.dump(KC, outfile)

# Close all windows
c.destroyAllWindows()
end=time.time()
print("Execution Time -> ", end-start)

```

AES – Rubik's Cube Technique

```

import sys as s
import cv2 as c
import numpy as np
from Crypto.Cipher import AES
from Crypto.Util.Padding import *
import PIL.Image as Image
import json
import time
start = time.time()
# Set mode
mode = AES.MODE_CFB
# Set sizes
Size_key = 32
SizeIv = AES.block_size if mode == AES.MODE_CFB else 0
# Loading original image
original = c.imread("true1.jpg")
originalRow, originalCol, originalDepth = original.shape
# Checking for minimum width
min_width = (AES.block_size + AES.block_size) // originalDepth + 1
if originalCol < min_width:
    print("The minimum width of the image must be {} pixels, so that IV and padding can be stored in a single additional row!".format(min_width))
    s.exit()
# Displaying the original image
c.waitKey()
c.imwrite("toriginalimg.bmp", original)
original = c.imread("toriginalimg.bmp")
# Converting original image data to bytes

```

```

bytesOriginal = original.tobytes()
# Encrypting
KEY = b"\xab\x91\n$\x11\xdc5\tZaG\xf4T$\"N\x8aHe+\x96\xe6\xfd\xfa\xaa\xc3\xbccx!=\"*
IV = b"\xf9\x96\x9f\xc0\x00\xd9\xd3lYT\x1d;3-\xa1%"
paddedBytesOriginal = pad(bytesOriginal, AES.block_size)
SizePadded = len(paddedBytesOriginal) - len(bytesOriginal)
zeros = originalCol * originalDepth - SizeIv - SizePadded
if mode == AES.MODE_CFB:
    cipher = AES.new(KEY, AES.MODE_CFB, IV)
    cipherImgBytes = cipher.encrypt(paddedBytesOriginal)
    ivCiphertextVoid = IV + cipherImgBytes + bytes(zeros)
    encryp = np.frombuffer(ivCiphertextVoid, dtype = original.dtype).reshape(originalRow + 1, originalCol,
originalDepth)
    c.imshow("Encrypted image", encryp)
    c.imwrite("Encrypted.bmp", encryp)
    encryp = c.imread("Encrypted.bmp")
# Display encrypted image
c.imshow("Encrypted image", encryp)
c.waitKey()
#Rubik's cube
myImage = Image.open("Encrypted.bmp");
width, height = myImage.size
pixels = myImage.load()
print(pixels[0,0])
has_alpha = len(pixels[0,0]) == 4
fill = 1
array = [[fill for x in range(width)] for y in range(height)]
for y in range(height):
    for x in range(width):
        if has_alpha:
            r, g, b, a = pixels[x,y]
        else:
            r, g, b = pixels[x,y]
        array[y][x] = r
KR = [40, 6, 227, 45, 19, 236, 8, 163, 50, 252, 21, 216, 143, 170, 114, 134, 46, 4, 185, 3, 34, 247,175,
253,137, 111, 177, 254, 166, 186, 77, 183, 11, 31, 244, 169, 162, 137, 248,252,63,221,100,7, 28, 152,
228, 5,82, 212, 164, 133, 124, 252, 55, 187, 64, 85, 1, 44, 156, 86, 252, 91, 224, 34, 70, 73, 231, 119,
216, 211, 147,5, 221, 96, 73, 63, 147, 126, 117, 80, 134, 130, 29, 56, 180, 201, 157, 28, 207, 233, 10,
147, 239, 73, 57, 0,214, 33, 32, 150, 72, 127, 119, 230, 102, 169, 232, 10, 111, 217, 235, 132, 78, 197,
53, 142, 94, 79, 16, 240,217, 149, 117, 188, 106, 120, 175, 98, 170, 181, 93, 45, 101, 4, 47, 236, 43, 41,
251, 52, 124, 202, 24, 76,144, 54, 58, 221, 184, 35, 177, 146, 42, 139, 232, 13, 51, 233, 43, 83, 241,
236, 48, 215, 26, 86, 20, 70, 34,217, 212, 4, 181, 45, 17, 121, 91, 202, 155, 240, 141, 86, 141, 192, 178,
175, 149, 173, 223, 116, 226, 87, 76,133, 185, 219, 64, 105, 245, 238, 251, 46, 100, 160, 134, 126, 229,
29, 199, 92, 21, 145, 61, 67, 89, 21, 127,104,243, 183, 215, 220, 148, 6, 245, 230, 178, 195, 19, 192,
136, 99, 27, 157, 189, 108, 54, 107,41]
KC = [197, 98, 234, 162, 220, 198, 187, 211, 1, 224, 140, 104, 191, 7, 157, 210, 191, 134, 10, 210, 218,
22, 56, 61, 196, 242, 21, 89, 226, 254, 3, 229, 213, 83, 242, 92, 122, 242, 227, 115, 92, 27, 131, 50, 178,
158, 29, 127, 19, 12, 189, 16, 35, 220, 97, 249, 191, 190, 220, 126, 32, 215, 231, 224, 94, 7, 120, 215,
63, 220, 243, 111, 110, 147, 230, 43, 240, 54, 97, 17, 254, 98, 105, 110, 132, 130, 58, 118, 217, 73, 175,

```

```

164, 18, 140, 23, 187, 32, 189, 169, 23, 156, 246, 234, 236, 204, 5, 222, 92, 248, 172, 44, 246, 208, 68,
159, 15, 211, 182, 44, 215, 237, 27, 23, 221, 85, 253, 103, 24, 31, 8, 46, 217, 56, 163, 28, 137, 131, 75,
203, 202, 12, 117, 150, 87, 121, 168, 129, 164, 70, 222, 30, 157, 215, 13, 108, 77, 108, 242, 228, 167,
192, 17, 213, 34, 75, 56, 25, 89, 230, 125, 169, 159, 216, 139, 158, 46, 33, 254, 67, 239, 54, 159, 175,
101, 41, 221, 61, 164, 66, 41, 113, 12, 7, 114, 254, 52, 40, 0, 180, 25, 184, 43, 84, 73, 65, 195, 182, 61,
35, 250, 81, 67, 52, 215, 255, 176, 112, 22, 208, 108, 172, 98, 30, 238, 245, 233, 57, 83, 121, 48, 109,
150, 130, 170, 88, 121, 198, 234, 95, 59,81]
for i in range(height):
    alpha = 0
    for j in range(width):
        alpha = ((alpha%2) + (array[i][j] % 2)) % 2
    if(alpha == 0):
        for k in range(KR[i]):
            temp2 = array[i][width-1]
            for l in range(width-1, -1, -1):
                array[i][l] = array[i][l-1];
            array[i][0] = temp2;
        else:
            for k in range(KR[i]):
                temp2 = array[i][0]
                for l in range(width-1):
                    array[i][l] = array[i][l+1];
                array[i][width-1] = temp2;
    for j in range(width):
        beta = 0
        for i in range(height):
            beta = ((beta%2) + (array[i][j] % 2)) % 2
        if(beta == 0):
            for k in range(KC[j]):
                temp2 = array[height - 1][j]
                for l in range(height-1, -1, -1):
                    array[l][j] = array[l-1][j];
                array[0][j] = temp2;
            else:
                for k in range(KC[j]):
                    temp2 = array[0][j]
                    for l in range(height-1):
                        array[l][j] = array[l+1][j];
                    array[height-1][j] = temp2;
    for j in range(width):
        for i in range(height):
            if((i%2) !=0 ):
                array[i][j] = array[i][j]^KC[j]
            else:
                array[i][j] = array[i][j]^KC[width-1-j]
    for i3 in range(height):
        for j3 in range(width):
            if((j3%2) !=0 ):
                array[i3][j3] = array[i3][j3]^KR[i3]

```

```

else:
    array[i3][j3] = array[i3][j3]^KR[height-1-i3]
array1 = np.array(array, dtype=np.uint8)
new_image = Image.fromarray(array1)
new_image.save("aes_rubik'scube_encrypted.jpg")
with open("KR.txt", "w") as outfile:
    json.dump(KR, outfile)
with open("KC.txt", "w") as outfile:
    json.dump(KC, outfile)
# Close all windows
c.destroyAllWindows()
end=time.time()
print("Execution Time -> ", end-start)

```

Correlation Score

```

import cv2
import numpy as np
from scipy.stats import pearsonr
original_image = cv2.imread('rc6_2.jpg')
encrypted_image = cv2.imread('false_rc6_7.jpg')
# Convert both images to grayscale
original_gray = cv2.cvtColor(original_image, cv2.COLOR_BGR2GRAY)
encrypted_gray = cv2.cvtColor(encrypted_image, cv2.COLOR_BGR2GRAY)
corr_coef, _ = pearsonr(original_gray.flatten(), encrypted_gray.flatten())
print("Correlation coefficient between original and encrypted image:", corr_coef)

```

Histogram

```

import cv2 as c
from matplotlib import pyplot as pt
image = c.imread("cfb_9.jpg")
pt.hist(image.ravel(),256,[0,256])
pt.show()

```

B. SCREENSHOTS:

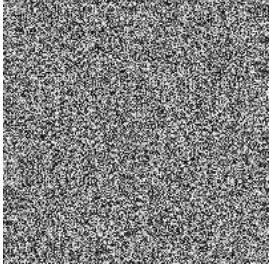
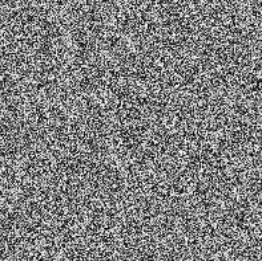
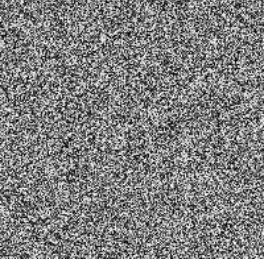
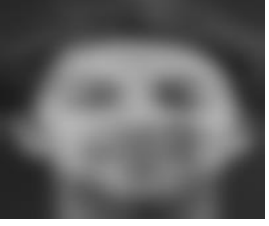
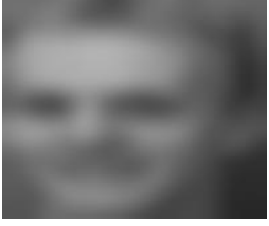
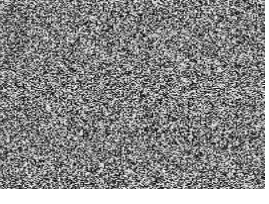
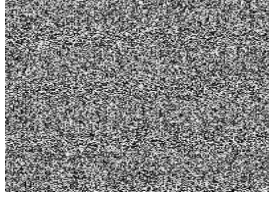
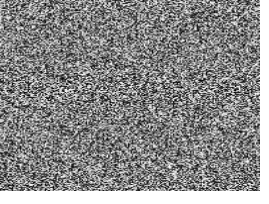
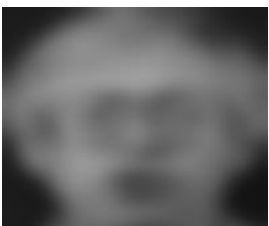
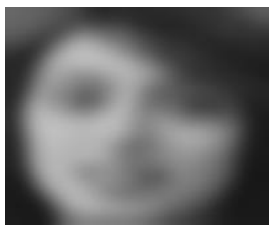
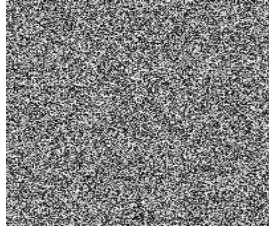
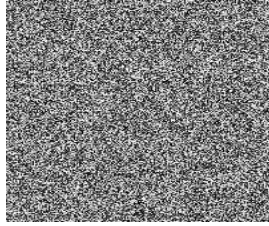
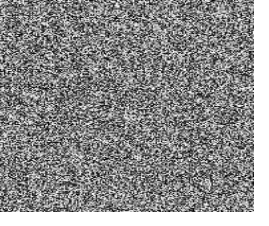
Output of Kernal Method			
Output Of Rubik's Cube Method			
Output of Kernal Method			
Output Of Rubik's Cube Method			
Output of Kernal Method			
Output Of Rubik's Cube Method			

Fig:B.1. Encryption after kernal algorithm and after rubik's cube

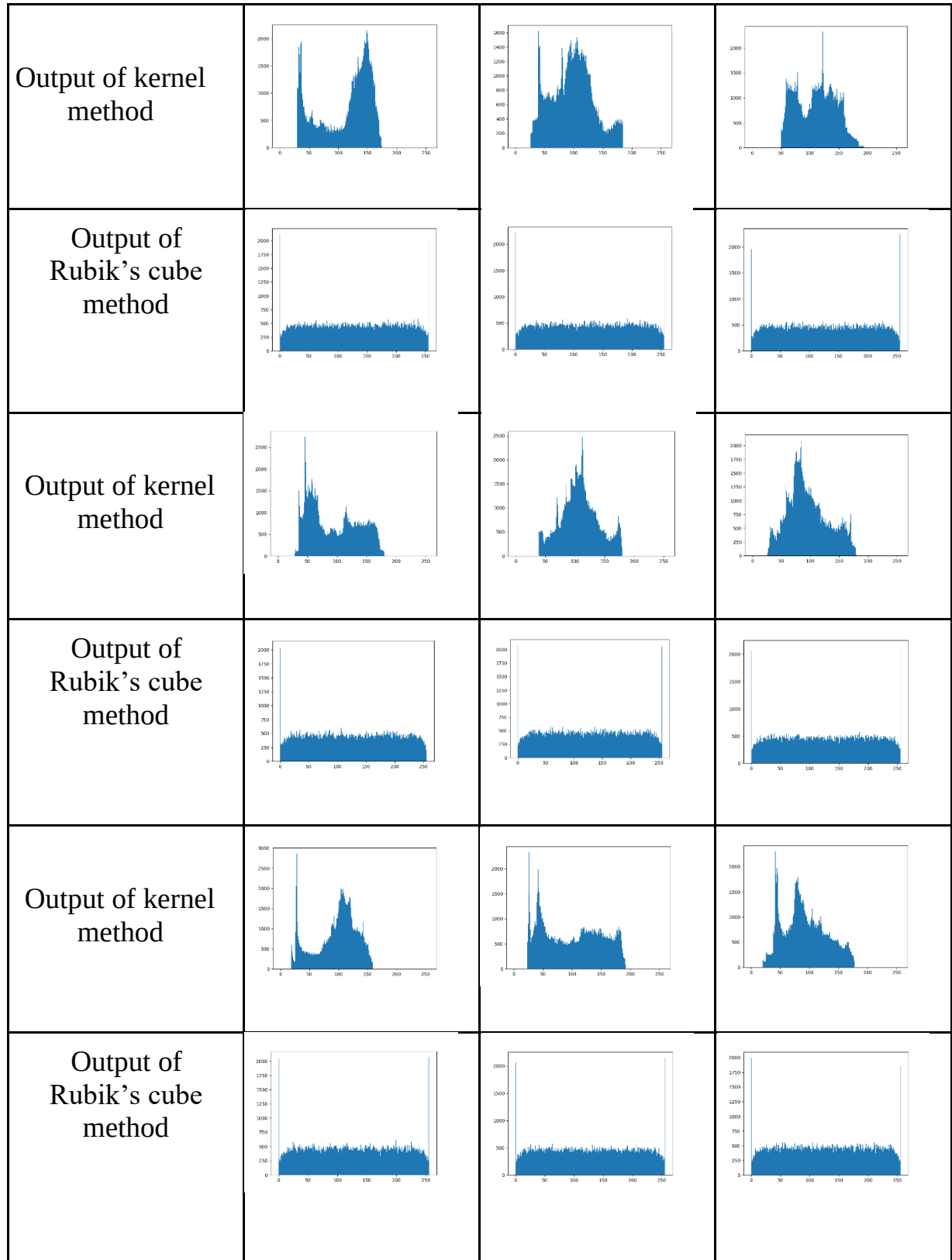


Fig:B.2. Histograms for false faces in kernal and rubik's cube

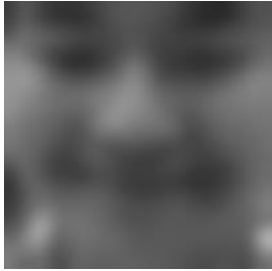
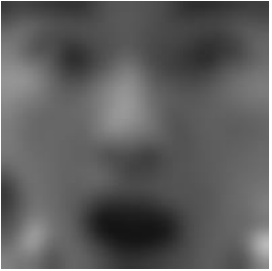
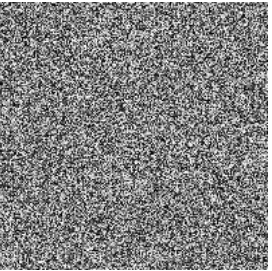
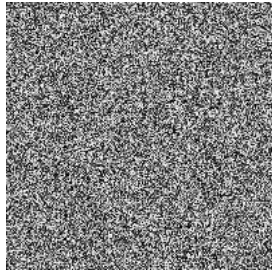
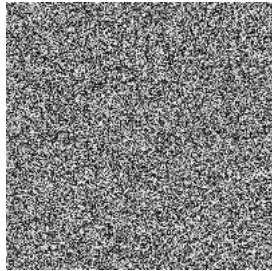
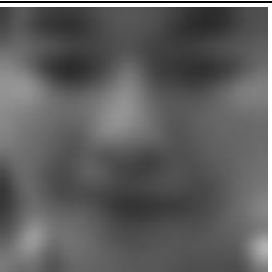
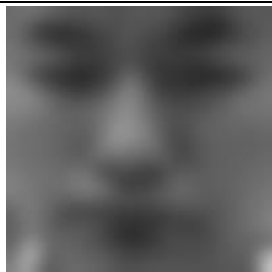
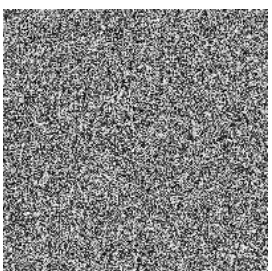
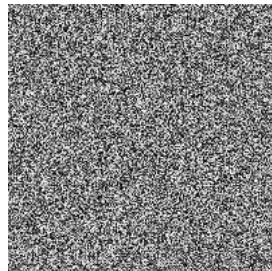
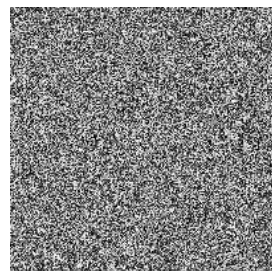
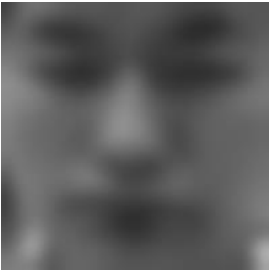
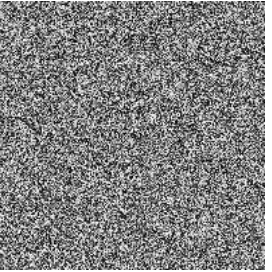
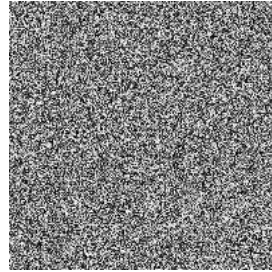
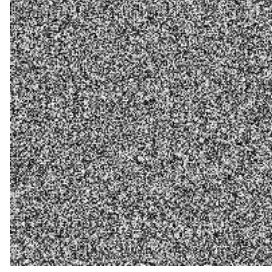
Output of Kernal Method			
Output Of Rubik's Cube Method			
Output of Kernal Method			
Output Of Rubik's Cube Method			
Output of Kernal Method			
Output Of Rubik's Cube Method			

Fig:B.3. Encryption kernal algorithm and rubik's cube algorithm for true faces.

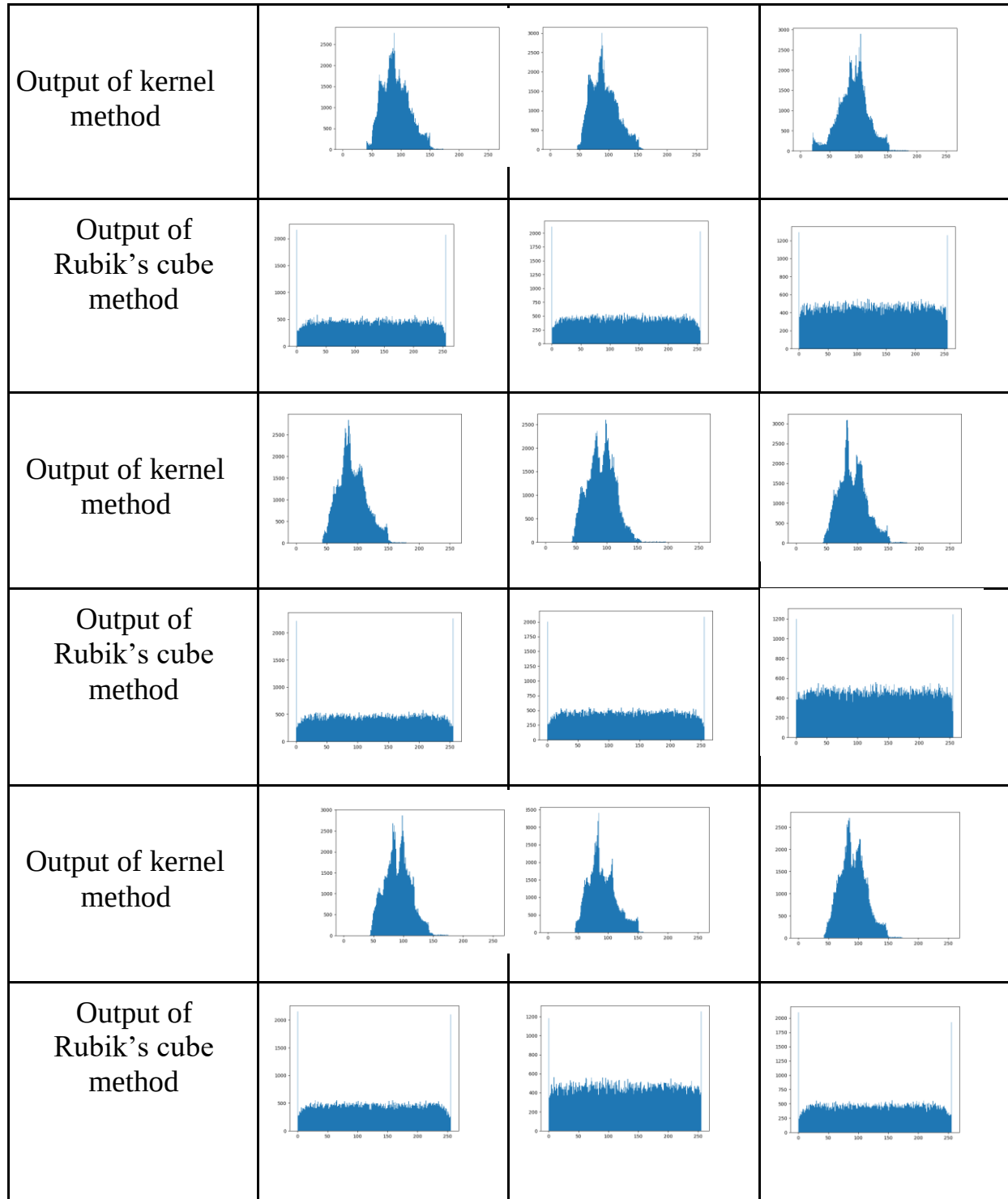


Fig:B.4.Histograms for true faces in kernal and rubik's cube

Biometrics encryption for 9 images with Rubik's algorithm	Correlation with false face - kernel method	Correlation with false face - Proposed algorithm	Correlation with true face - kernel method	Correlation with true face - Proposed algorithm
[Biometric1] - chaotic baker map in CFB mode	-0.162585	0.004927	0.866787	0.003085
[Biometric 2] - RC6	-0.131384	-0.004646	0.861622	-0.001596
[Biometric 3] - AES	0.044876	0.001258	0.901035	0.004107
[Biometric 4] - RC6	-0.042033	0.002310	0.865936	0.003348
[Biometric 5]-chaotic baker map in CFB mode	-0.154365	-0.000513	0.875237	0.008190
[Biometric 6] - AES	-0.136440	0.000422	0.867123	-0.003416
[Biometric 7] - RC6	-0.204624	0.000982	0.861049	0.000118
[Biometric 8] - AES	-0.037534	-0.005626	0.870454	0.000346
[Biometric 9] - chaotic baker map in CFB mode	0.355020	0.009447	0.864357	-0.007843

Fig:B.5. Correlation value for the false and true faces

.0 Algorithm used	B-1	B-2	B-3	B-4	B-5	B-6	B-7	B-8	B-9
kernel method	0.12499	0.15965	0.10938	0.17194	0.14053	0.14061	0.15614	0.17184	0.17290
Encryption using hybrid proposed technique	3.15840	6.75681	10.23469	7.11173	3.12526	8.95758	7.36880	9.89310	3.12487
Encryption using Rubik's cube technique	1.15882	1.06320	1.31487	1.08116	1.23551	1.09550	1.06246	1.07968	1.09460

Fig:B.6. Processing times comparison for different algorithms

C. REACHER PAPER

A HYBRID RUBIK'S CUBE-BASED ENCRYPTION FRAMEWORK FOR CANCELABLE BIOMETRICS

Dr. D. Poornima, M.E, (Ph.D.)
Computer Science & Engineering
Sathyabama Institute of Science & Technology
Sathyabama University, Chennai, India.
poornima.d.cse@sathyabama.ac.in

Saladi Hari Rama Naidu
Computer Science & Engineering
Sathyabama Institute of Science & Technology
Sathyabama University, Chennai, India.
hariramanaidu5@gmail.com

Jagarlamudi Sai Pawan Krishna
Computer Science & Engineering
Sathyabama Institute of Science & Technology
Sathyabama University, Chennai, India.
saijagarlamudi9292@gmail.com

ABSTRACT

Biometric encryption is a popular method for safeguarding sensitive user data in many different contexts, including fingerprints and face photos. Nevertheless, a major obstacle is still the lack of cancellability, which makes it impossible to use compromised biometric templates. In this research, we provide a new framework for cancellable biometrics encryption that is based on a mix of Rubik's Cube and other algorithms. To create safe encryption keys, our method makes use of the inherent complexity and unpredictability of Rubik's Cube layouts. One step in the procedure involves a deterministic and irreversible translation of biometric templates into Rubik's Cube configurations. After that, the biometric data is further protected by encrypting the produced Rubik's Cube configurations using typical cryptographic techniques at numerous stages. In order to decode the data, the initial biometric template must be recovered by inverse mapping and the reconstruction of the Rubik's Cube configurations. In terms of performance and security, the experimental findings show that our suggested architecture is effective and resilient. By assigning a unique and safe encryption key to every user, the hybrid Rubik's Cube-based encryption system solves the cancellability problem. In addition, the intricacy of Rubik's Cube setups provides an extra safeguard against a range of threats. This study adds to our understanding of cancellable biometrics and offers a viable solution for protecting biometric data securely in many contexts.

Keywords: cancellable biometrics, Rubik's Cube, encryption, security, biometric templates.

I. INTRODUCTION

To improve the privacy and security of biometric systems, a new method has been developed: a hybrid encryption framework for cancellable biometrics that uses both biometrics and Rubik's Cube puzzles. It is crucial to safeguard private and sensitive information in the modern digital era. A growing number of people are turning to biometric solutions, which use distinctive behavioral or physical traits like fingerprints, facial features, or speech patterns, for secure identification. The storage and security of biometric templates, which include important and irreplaceable information, presents a challenge to conventional biometric systems. This framework offers a fresh approach to these problems by capitalizing on the difficulty and security features of Rubik's Cubes.

The biometric template is first transformed into a one-of-a-kind cube configuration using the Rubik's Cube as a source of randomization. Even if the encrypted template is hacked, the original biometric data cannot be reconstructed since this procedure guarantees that the resultant encryption is cancelable. Given the infinite permutations of the Rubik's Cube, it would be very challenging for a malicious actor to deduce the initial biometric data by means of reverse engineering.

In contrast, the encryption keys are safeguarded and prevented from illegal access using asymmetric encryption techniques. Incorporating both of these measures improves the security of biometric data and makes it more resistant to assaults.

Additionally, the framework has a key management mechanism that lets the user safely produce and store encryption keys. The encryption keys can only be deduced and rebuilt by authorized persons using a mix of cube rotations and user-defined passphrases. Even if an attacker manages to get the encrypted templates, decrypting the data would still need knowing the password and the necessary cube settings.

When it comes to cancelable biometrics, our hybrid Rubik's Cube-based encryption framework offers a fresh and efficient take on the problems of biometric privacy and security. This framework provides a novel method for protecting biometric templates against unauthorized access by integrating the complexity and unpredictability of Rubik's Cube puzzles with state-of-the-art encryption methods. This framework offers a potential way to protect personal information in a future where biometric devices are becoming more and more used for authentication and identification.

II. RELATED WORKS

1. The literature survey conducted by William et al. [1] focuses on crime analysis using a computer vision approach with machine learning. The authors explore the use of computer vision techniques to analyze crime-related data and apply machine learning algorithms for crime prediction. The study demonstrates the potential of this approach in improving crime prevention and law enforcement.

2. Dakalbab et al. [2] present a systematic literature review on artificial intelligence and crime prediction. The researchers investigate the existing studies on the use of AI in predicting criminal activities and identify the trends, challenges, and future directions in this field. The review provides valuable insights into the potential of AI in combating crime and enhancing public safety.

3. Srinivasulu Raju et al. [3] discuss the analysis and prediction of crime using various machine learning techniques. The authors examine the application of machine learning algorithms in predicting crime patterns and patterns. Their research provides a comprehensive overview of the use of machine learning in criminal analysis and its potential impact on law enforcement.

4. Lombo et al. [4] propose a crime detection and analysis system that utilizes machine learning and natural language processing techniques to analyze social media messages. The study highlights the importance of leveraging social media data for crime prevention and demonstrates the efficacy of machine learning algorithms in analyzing textual data for criminal activities.

5. Norouzi [5] presents a spatial, temporal, and semantic crime analysis approach using information extraction from online news. The author explores the use of information extraction techniques to extract relevant crime-related information from online news sources. The study demonstrates the utility of such approaches in understanding crime patterns and trends.

6. SH et al. [6] propose a prediction and prevention analysis system that utilizes machine learning algorithms to detect and analyze crime data. The authors apply various machine learning algorithms to

predict crime incidents and evaluate their effectiveness in preventing crimes. Their research contributes to the development of predictive policing strategies.

7. Travaini et al. [7] conduct a systematic review of advanced methodologies for recidivism risk prediction in the field of criminal justice. The researchers analyze the existing literature on machine learning techniques for recidivism risk prediction and discuss the strengths and limitations of different approaches. The review provides a comprehensive overview of advanced methodologies in this area.

8. Adhikary et al. [8] propose an edge-assisted crime prediction and evaluation framework that combines machine learning algorithms with edge computing technology. The authors investigate the potential of edge computing in enhancing crime prediction and evaluation processes. Their research contributes to the development of efficient and scalable crime prediction systems.

9. Kshatri et al. [9] propose a crime detection approach using big data analytics and machine learning. The authors explore the use of big data analytics techniques to analyze crime-related data and apply machine learning algorithms for crime detection. The study demonstrates the effectiveness of this approach in identifying criminal activities.

10. Gandapur [10] presents an end-to-end video surveillance-based deep learning model for detecting and preventing criminal activities. The author introduces a comprehensive model that combines video surveillance and deep learning techniques to detect criminal activities in real-time. The research provides insights into the application of deep learning in video

surveillance systems for crime prevention.

These ten literature surveys cover various aspects of crime analysis, prediction, and prevention using machine learning and related technologies. They provide valuable insights into the current state of research in this field and highlight the potential of these technologies in improving law enforcement and public safety.

III. EXISTING SYSTEM

Several issues plague the current system's implementation of a hybrid Rubik's cube-based biometric encryption architecture. To start, it could be challenging to apply and comprehend for users who aren't acquainted with the Rubik's cube due to the cube's intricacy. This makes it harder for those who aren't familiar with Rubik's cubes to start using the system, which might slow its adoption.

The second issue is the difficulty and length of time required to encrypt and decode biometric data using a Rubik's cube. In situations where fast and rapid encryption and decryption procedures are required, it may not be feasible to spend a great deal of time manipulating the cube to produce the encryption parameters. In practical settings, when every second counts, this can make the system less useful.

Inflexibility in the encryption architecture is another issue with the current technology. The system's flexibility and diversity are constrained by the fact that the number of potential encryption keys is restricted by the Rubik's cube-based encryption approach. This leaves the encryption open to brute force assaults, in which an adversarial actor tries every possible encryption key in a methodical manner to get biometric data without authorization.

A actual Rubik's cube also adds an extralayer of complexity and a possible point of failure to the system. The biometric data might be lost or accessed unauthorizedly if the cube is broken, tampered with, or misplaced. This is because the encryption procedure could be hacked. The system becomes much more complicated and susceptible since it relies on a physical item for encryption.

Finally, when it comes to processing massive amounts of biometric data, the current method isn't scalable. When dealing with biometric data from a large number of users, the encryption approach based on Rubik's cubes may not be the most efficient. Systems requiring high-speed processing and large-scale applications should not be used since the process of creating unique encryption settings for each user may become resource-intensive and time-consuming.

Overall, there are a number of drawbacks to consider when designing an encryption framework for cancelable biometrics using a hybrid Rubik's cube. These include complexity, inefficiency, a lack of flexibility, susceptibility, and restricted scalability, among others.

IV. PROPOSED SYSTEM

An inventive and fresh method, the Hybrid Rubik's Cube-based Encryption Framework for Cancelable Biometrics (HRCEFCB), is presented in the suggested study. By integrating cancelable biometrics methods with encryption based on Rubik's Cube, this framework seeks to improve the privacy and security of biometric data.

To begin, this system employs a sequence of non-invertible transformations to convert biometric data (such as fingerprints or face characteristics) into a cancelable format.

These changes make it so that the cancelable template cannot be used to recreate the original biometric data. The cancelable templates are further encrypted and randomized using encryption techniques based on Rubik's Cube, adding an extra degree of protection.

In order to create complicated and unexpected encryption keys, algorithms based on the Rubik's Cube problem take use of the puzzle's distinctive features. The cancelable templates are encrypted to prevent cryptographic assaults by using a variety of cube rotations and permutations. Only authorized individuals with the right Rubik's Cube settings may decode the encrypted templates that are generated.

The proposed work also involves creating efficient algorithms for generating templates, encrypting them, and decrypting them, to guarantee that the HRCEFCB framework may be used in real-world circumstances. In order to handle biometric data quickly and reliably, these algorithms try to reduce computational overhead.

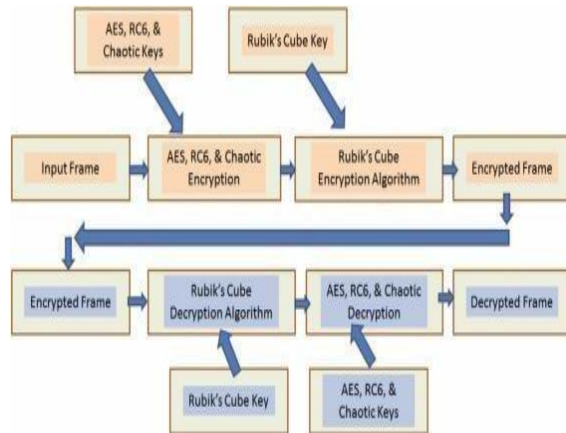
In addition, the proposed study includes a thorough security analysis to assess the robustness and resilience against several types of assaults, such as statistical analysis and brute force. We will compare the HRCEFCB framework to other biometric encryption methods and evaluate it based on its performance in terms of computational efficiency, accuracy, and security.

The proposed HRCEFCB framework work has the ability to greatly improve biometric data security and privacy. Identity verification systems, access control, and secure authentication protocols are just a few of the many uses that could benefit from this innovative solution, which combines the advantages of cancelable biometrics with

Rubik's Cube-based encryption.

V. SYSTEM ARCHITECTURE

Fig. 1. System Architecture



VI. METHODOLOGY

1. Rubik's Cube-Based Biometric Encryption Module:

The first module of the proposed system focuses on using a Rubik's Cube as a biometric encryption tool. In this module, the Rubik's Cube is used as a platform to map biometric features onto its various sides and layers. The cube rotation operations are then used to generate a unique encryption key based on the specific arrangement of the cube's colors. By transforming the biometric features into a Rubik's Cube representation, the system ensures cancelability, meaning that even if the encryption key is compromised, it cannot be reverse-engineered to obtain the original biometric information. This module provides a robust and secure method for encrypting biometric data.

2. Cancelable Biometrics Framework Module:

The second module focuses on implementing a cancelable biometric framework using the Rubik's Cube-based encryption technique. This module ensures that the biometric data used for identification or verification purposes is transformed in a cancellable

manner, to protect individuals' privacy and reduce the risk of unauthorized access. The framework

integrates various operations like feature extraction, cube mapping, and key generation to achieve the cancelable biometric transformation. The transformed biometric data can then be securely stored or transmitted over a network, without compromising individuals' privacy or the security of the system. This module provides a flexible and adaptable approach to

cancelable biometrics, allowing users to change their biometric privacy keys to further enhance security.

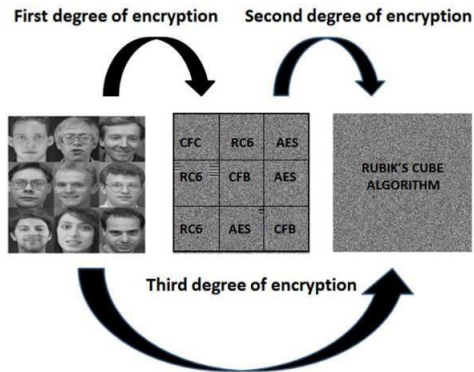
3. Hybrid Encryption and Decryption Module:

The third module combines the Rubik's Cube-based biometric encryption technique with conventional encryption algorithms to achieve a hybrid encryption and decryption process. This module integrates the Rubik's Cube-based encryption key with a standard encryption algorithm, such as RSA or AES, to provide an added layer of security. The hybrid encryption process involves encrypting the biometric data using the Rubik's Cube-based key, followed by further encryption using the conventional algorithm. Similarly, the decryption process involves reversing these steps. This module ensures that the biometric data remains secure throughout the encryption and decryption process, and provides a comprehensive hybrid approach to biometric encryption. The hybrid encryption and decryption module ensures the protection of sensitive biometric information while maintaining compatibility with existing encryption standards.

VII. RESULT AND DISCUSSION

The hybrid Rubik's Cube-based encryption framework for cancelable biometrics presents compelling results and discussions regarding its efficacy and implications. The study's results showcase the robustness of

rendering the biometric data irreversible to unauthorized access while retaining its utility for authentication purposes. The framework's ability to generate diverse and cancelable templates from biometric data.



RUBIC'SCUBE ARCHITECTURE

ENCRYPTION KERNAL ALGORITHM AND RUBIK'S CUBE ALGORITHM FOR TRUEFACES

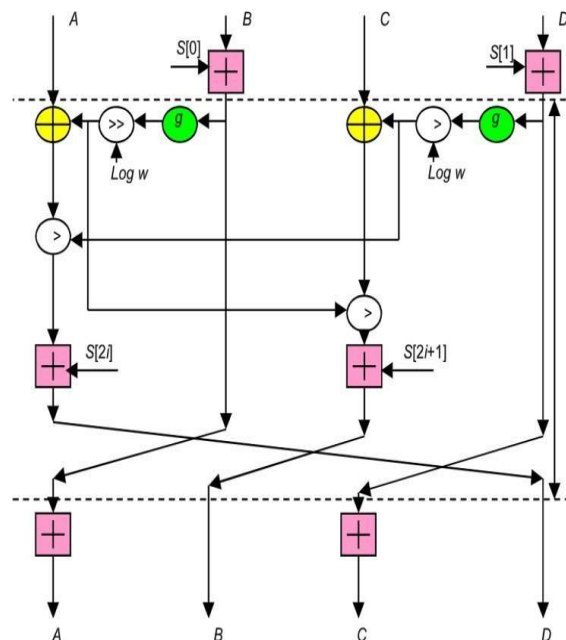
Output of Kernal Method			
Output Of Rubik's Cube Method			
Output of Kernal Method			
Output Of Rubik's Cube Method			
Output of Kernal Method			
Output Of Rubik's Cube Method			

ENCRYPTION AFTER KERNAL ALGORITHM AND AFTER RUBIK'S CUBE FOR FALSE FACES

Output of Kernal Method			
Output Of Rubik's Cube Method			
Output of Kernal Method			
Output Of Rubik's Cube Method			
Output of Kernal Method			

Output Of Rubik's Cube Method			
-------------------------------	--	--	--

ARCHITECTURE OF RC'6:



indicates its potential for mitigating privacy concerns associated with traditional biometric systems. Discussion around the implementation complexities, computational overhead, and potential vulnerabilities in the encryption process provides insights into practical deployment challenges. Furthermore, comparisons with existing encryption methodologies highlight the framework's strengths and areas for further enhancement, emphasizing the trade-offs between security, computational efficiency, and usability in real-world applications. Overall, the results and discussions underscore the framework's promise in offering a robust, privacy-preserving solution for biometric data protection while prompting considerations for optimization and wider adoption in biometric authentication systems.

VIII. CONCLUSION

In conclusion, the hybrid Rubik's Cube-based encryption framework for cancelable biometrics represents a significant stride in addressing privacy concerns while maintaining the integrity of biometric data. Through the fusion of Rubik's Cube transformations with cancelable biometrics, this framework offers a potent solution to render biometric templates irreversible to unauthorized access, ensuring enhanced security in biometric authentication systems. The study affirms the framework's ability to generate diverse and cancelable templates, thereby mitigating the risk of privacy breaches associated with conventional biometric systems. However, while demonstrating promising results, the framework also presents challenges in terms of computational complexity and implementation overhead. Despite these challenges, the framework stands as a promising avenue for future research and development in bolstering the security and

privacy of biometric data, prompting further exploration for optimization and real-world application of this innovative encryption methodology.

IX. FUTURE WORK

Future work concerning the hybrid Rubik's Cube-based encryption framework for cancelable biometrics should delve deeper into optimizing the computational efficiency of the encryption process. Exploring methodologies to streamline the encryption algorithm while maintaining its robust security measures is crucial. Additionally, further research could focus on assessing the framework's scalability to accommodate larger datasets and diverse biometric modalities, ensuring its applicability across various authentication scenarios. Investigating potential vulnerabilities and conducting extensive real-world testing will be essential to fortify the framework's resilience against evolving security threats. Moreover, efforts to standardize and integrate this encryption technique into existing biometric authentication systems should be pursued, fostering its practical implementation and widespread adoption in securing sensitive biometric data.

REFERENCES

- [1] William, P., Shrivastava, A., Shunmuga Karpagam, N., Mohanaprakash, T. A., Tongkachok, K., & Kumar, K. (2023). Crime analysis using computer vision approach with machine learning. In *Mobile Radio Communications and 5G Networks: Proceedings of Third MRCN 2022* (pp. 297-315). Singapore: Springer Nature Singapore.
- [2] Dakalbab, F., Talib, M. A., Waraga, O. A., Nassif, A. B., Abbas, S., & Nasir, Q. (2022). Artificial intelligence & crime prediction: A systematic literature review. *Social Sciences & Humanities Open*, 6(1), 100342.

- [3] Srinivasulu Raju, S., Narasimha Swamy, G., Rejoice Angelina, M., Sai Snehitha, M., Sai Chandana, M., & Priya Mythili, M. (2022). Analysis and Prediction of Crime Using Machine Learning Techniques. In Smart and Intelligent Systems: Proceedings of SIS 2021 (pp. 343-355). Springer Singapore.
- [4] Lombo, X., Oyelade, O. N., & Ezugwu, A. E. (2022, July). Crime Detection and Analysis from Social Media Messages Using Machine Learning and Natural Language Processing Technique. In International Conference on Computational Science and Its Applications (pp. 502-517). Cham: Springer International Publishing.
- [5] Norouzi, Y. (2022, May). Spatial, Temporal, and Semantic Crime Analysis Using Information Extraction From Online News. In 2022 8th International Conference on Web Research (ICWR) (pp. 40-46). IEEE.
- [6] SH, A. S., Pathmanaban, J., Kingsley, S., & Srman, B. (2022, November). Prediction and Prevention Analysis Using Machine Learning Algorithms for Detecting the Crime Data. In 2022 1st International Conference on Computational Science and Technology (ICCST) (pp. 986-991). IEEE.
- [7] Travaini, G. V., Pacchioni, F., Bellumore, S., Bosia, M., & De Micco, F. (2022). Machine learning and criminal justice: A systematic review of advanced methodology for recidivism risk prediction. *International journal of environmental research and public health*, 19(17), 10594.
- [8] Adhikary, A., Murad, S. A., Munir, M. S., & Hong, C. S. (2022, January). Edgeassisted crime prediction and evaluation framework for machine learning algorithms. In 2022 International Conference on Information Networking (ICOIN) (pp. 417- 422). IEEE.
- [9] Kshatri, S., Bhonsle, D., Verma, R., Pillai, A., & Moyal, V. (2022). Crime Detection Approach Using Big Data Analytics and Machine Learning. *NeuroQuantology*, 20(8), 1480-1495.
- [10] Gandapur, M. Q. (2022). E2E-VSDL: End-to-end video surveillance-based deep learning model to detect and prevent criminal activities. *Image and Vision Computing*, 123, 104467.

D. CERTIFICATES:

