Abstract

Owing to the various advantages offered by biometric authentication systems over knowledge and possession based systems like passwords and pins, the use of the former has increased exponentially in popularity. One of the most common ways of biometric authentication is by using fingerprints. Fingerprints are unique and tough to forge. However, more often than not, they are stored as plaintext strings in a device, and are hence vulnerable. This is why biometric authentication is used most prominently in smartphones, and rarely in client server applications. This project attempts to implement an authentication system in identification mode, that encrypts fingerprint templates before storing them, in order to prevent adversarial attacks. AES in CBC mode is employed in order to secure the fingerprint templates in the database.

Table of Contents

List of Figures

Acronyms

- **AES**:Advanced Encryption Standard
- **ECB**: Electronic Code Book
- **CBC**: Cipher Block Chaining
- **TP**: True positive
- **TN**: True Negative
- **FP**: False positive
- **FN**: False Negative
- **UART**: Universal Asynchronous Receiver/Transmitter

Chapter 1

Introduction

1.1 Background

A biometric is a measurement of an individual's unique physical or behavioural characteristics. Facial recognition, retinal scans, and fingerprint mapping are popular examples of biometric security systems. By using a Biometric Encryption (BE) method, one can personalize the biometric to encode a PIN, a password, or an alphanumeric string for a multitude of applications such as bank ATMs, building access, and computer terminal access. Basically, no PINs are required to be remembered in this case. Moreover, the database only needs to store the biometric's encryption, not the large biometric sample itself.

A biometric system may operate in one of two modes - the **Verification Mode** or the **Identification Mode**

Verification Mode: In the verification mode, the system verifies whether the identity claimed by an individual is true. This is accomplished by a one-to-one comparison. Verification asks "Is the person who they say they are?". A person identifies as a certain user, and must provide proof that they can confirm their identity compared to already-stored data. It aims to match a specific individual, rather than cast a wide net to find similarities in the identification process. Also, verification usually works more rapidly than identification, as it filters for highly-specific parameters in a smaller database. Authentication is another concept in biometric matching, closely aligned with verification.

Identification Mode: Identification is the task of answering "Who is this person?". It consists of receiving data about an unknown individual, such as a photo of their face, their voice biometrics or fingerprints, and comparing it to a larger database to uncover a potential match. Identification systems are described as a 1-to-N matching systems (sometime written '1:N'), where N is the total number of biometrics in the database. Identification tends to take longer than verification, as the algorithm must compare the reference data against a larger set of subjects to find a match. Forensics labs are one example of an identification operation, as they store large banks of biometric data ranging from fingerprints and DNA samples collected at a crime scene. That information is

then compared to newly-provided samples to prove the presence of the suspect.

1.2 Objective

The objective of this project is to develop a fingerprint authentication system that provides -

• **Confidentiality:** Even if an adversary is able to access a template, they must not be able to decrypt it.

• **Integrity:** The adversary should not be able to manipulate a fingerprint template in any way.

• **Revocability:** It must be possible to identify and revoke a compromised template, and reissue a new one that is representative of the user's true fingerprint.

• **Unforgeability:** The system must ensure ciphertexts cannot be existentially forged using fingerprint spoofs.

• **Efficiency:** The encryption scheme must not bear significantly on the speed of the matching algorithm.

1.3 Motivation

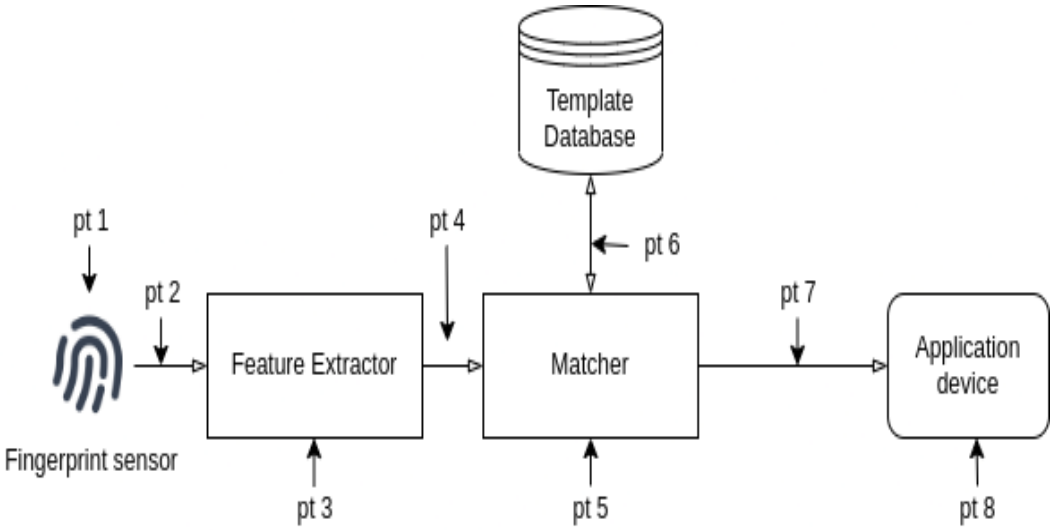


Fig1.1 Attacks on a generic Biometric system

Biometric authentication schemes are more efficient over traditional password or pin based authentication systems. Recent developments in sensing and computing technologies have made biometric systems more affordable and as a result they are easily embedded in a variety of smart consumer devices such as mobile phones and tablets. However, they are not immune to adversarial attacks. It is

quite easy for biometric templates stored as plaintext to be accessed illegally, and even tampered with and replayed to the authentication system, thereby risking the loss and forgery of confidential data by a malicious adversary.

Fig.1.1 demonstrates the most common attacks on biometric security systems.

- Point 1: Spoofing: Fake biometric at the sensor
- Point 2: Replay attack: re-submission of a digitally stored old fingerprint
- Point 3: Trojan horse attack at the feature extractor
- Point 4: Tampering with the feature representation
- Point 5: Masquerading: Override matcher
- Point 6: Substitution: Tampering with stored templates
- Point 7: Channel attack between template and matcher
- Point 8: Decision (yes/no) override

In our project, we focus on developing a system that counters possible attacks at points 5, and 6. Our proposed design works in identification mode.

1.4 Scope

As we know, Biometric fingerprint authentication is widely used in the public sector, and one such sector is banking. In bank branches, fingerprint authentication is used to verify the identity of customers of the branch, as well as authenticate certain transactions they make. It is also used to permit bank employees access to confidential information.
At ATMs, a user's fingerprint may be used to retrieve their account to perform transactions. Fingerprint Based ATM is a desktop application where fingerprint of the user is used as a authentication. The finger print minutiae features are different for each human being so the user can be identified uniquely. Instead of using an ATM card, fingerprint based ATM is considered to be safer and secure. Given the value of transactions performed, it is absolutely essential that these systems are secure. However, in most banks, these systems are vulnerable to channel and database attacks. For example, a malicious listening device may be planted at the wire connecting the fingerprint module and the bank's computer system. Similarly, an adversary may gain access to the fingerprint template database. Side-channel attacks are also probable.
The attacks which we describe are seen in Fig1.1

1.5 Organisation of report

In **Chapter 2**,we start of with the literature survey on topics related to our project.

In **Chapter 3** we do an in-depth analysis of the problem statement, breaking it down into smaller components and dealing with each separately. We elaborate on the need for our project the major problems in existing technology.

In **Chapter 4** we discuss the various hardware and software requirements needed to meet the constraints applied, to give us an optimal and efficient model to achieve our described goal and standard. A brief summary is provided about the choice of hardware and software.

In **Chapter 5** comes the heart of the report. The system architecture, which provides a simple visualization of the various components of the system and their interaction with each other is described. The logic design, two simple flowcharts which demonstrate the logic and data flow of the system are included. Next, we describe the basic algorithm designed to successfully write efficient code to accomplish the task at hand.

In **Chapter 6** we have described the implementation of our proposed model. The interconnections and pin connection are provided along with the expected output and the results obtained.

Finally, a brief conclusion is provided to end the report, in **Chapter 7**.

Chapter 2

Literature Survey

Biometrics are a popular paradigm in the world of information security. In order to make the use of biometrics more robust and insurmountable to adversaries, several methods have been developed and worked on. Some of these methods that are relevant to this project have been studied, and briefly elucidated on, in the literature survey that follows. Of the methods available, fuzzy extractors and cancelabe biometrics are most used in the industry. However, it has to be noted that they take completely different approaches. Our approach, while orthogonal to these techniques, is similar to the design proposed by J. Mwema's research.

[1] Ratha, N.K., Connell, J.H., Bolle, R.M. (2001). "An Analysis of Minutiae Matching Strength."

In recent years there has been exponential growth in the use of biometrics for user authentication applications because biometrics based authentication offers several advantages over knowledge and possession-based methods such as password/PIN-based systems. However, it is important that biometrics-based authentication systems be designed to withstand different sources of attacks on the system when employed in security-critical applications. This is even more important for unattended remote applications such as e-commerce. In this paper, the authors outline the potential security holes in a biometrics-based authentication scheme, quantify the numerical strength of one method of fingerprint matching, then discuss how to combat some of the remaining weaknesses.

[2] Mwema, J. M. (2015). "Encryption of Biometric Fingerprint Templates using encryption keys obtained from other biometric fingerprint templates"

Biometric data extracted from physiological features of a person including but not limited to fingerprints, palm prints, face or retina for purpose of verification identification is saved as biometric templates. The inception of biometrics in access control systems has not been without its own hitches like other systems it has had its fair share of security challenges. Biometric fingerprints are

the most mature of all biometric spheres. Biometric systems are further subdivided into multimodal biometric systems and unimodal biometric systems. Effectiveness of biometric systems lies on how secure they are at averting inadvertent disclosure of biometric templates in an information system's archive. This however has not been the case as biometric templates have been fraudulently accessed to gain unauthorized access to information systems. In order to achieve strong and secure biometric systems, systems designers and developers need to build biometric systems that properly secure biometric templates. Several approaches and biometric template protection schemes have been used to safeguard stored biometric templates. Even though there are various biometric template protection schemes and approaches in existence, few of them have been concretely tailored for unimodal biometric systems. This research's intent was to establish an approach for securing biometric fingerprint templates in a relational database. To come up with this approach, precedent biometric template protection schemes and approaches were studied to determine their shortcomings after which an encryption scheme for securing biometric templates stored in a database by encrypting fingerprint templates with encryption keys derived from other fingerprints was designed, developed and tested to ascertain its efficacy. Evaluation of the results showed that a combination of security measures and not just one particular technique aids in optimizing security of archived biometric fingerprint templates.

[3] Topcu, Berkay Erdogan, Hakan Karabat, C. Yanikoglu, Berrin. (2013). "BioHashing with Fingerprint Spectral Minutiae"

In recent years, the interest in human authentication has been increasing. Biometrics are one of the easy authentication schemes, however, security and privacy problems limit their widespread usage. Following the interest in privacy protecting biometric authentication, template protection schemes for biometric modalities has increased significantly in order to cope with security and privacy issues. BioHashing, which is based on transforming the biometric template using pseudo-random projections that are generated using a user-specified key or token, has received much attention as it improves verification accuracies over using only the biometric data, allows template revocation and preserves privacy. In this work, the authors develop a new BioHashing scheme for fingerprints. A fixed-length feature vector is required in order to design a BioHashing scheme. In the literature, most of the studies on fingerprint BioHashing uses features extracted from fingerprint texture. On the other hand, our new BioHashing

scheme is based on minutia based feature vectors. The authors use the spectral minutiae representation for obtaining a fixed-length feature vector for a fingerprint sample. Then, they use a random projection matrix, which is generated from user's key/token, in order to generate a BioHash vector. They propose to randomly project each column of the spectral minutiae feature matrix via a single matrix which allows fast bit string extraction and adaptive quantization. Experiments on FVC2002 databases show the promise of the proposed system for fast and secure verification.

[4] V. M. Patel, N. K. Ratha and R. Chellappa. (2015). "Cancelable Biometrics: A review"

Recent years have seen an exponential growth in the use of various biometric technologies for trusted automatic recognition of humans. With the rapid adaptation of biometric systems, there is a growing concern that biometric technologies may compromise the privacy and anonymity of individuals. Unlike credit cards and passwords, which can be revoked and reissued when compromised, biometrics are permanently associated with a user and cannot be replaced. To prevent the theft of biometric patterns, it is desirable to modify them through revocable and noninvertible transformations to produce cancelable biometric templates. In this article, the authors provide an overview of various cancelable biometric schemes for biometric template protection. They discuss the merits and drawbacks of available cancelable biometric systems and identify promising avenues of research in this rapidly evolving field.

[5] G. I. Davida, Y. Frankel, and B. J. Matt. (1998). "On enabling secure applications through off-line biometric identification"

In developing secure applications and systems, designers must often incorporate secure user identification in the design specification. In this paper, the authors study secure off-line authenticated user identification schemes based on a biometric system that can measure a user's biometrics accurately (up to some Hamming distance). The presented schemes enhance identification and authorization in secure applications by binding a biometric template with authorization information on a token such as a magnetic strip. Also developed are schemes specifically designed to minimize the compromising of a user's private biometrics data, encapsulated in the authorization information, without requiring secure hardware tokens. They also study the feasibility of biometrics performing as an enabling

technology for secure systems and applications design. They investigate a new technology which allows a user's biometrics to facilitate cryptographic mechanisms.

[6] Jayapal, Ranjith Pramod,. (2016). "Biometric encryption system for increased security"

Security is very important in present day life. In this computer-networked world, most of the activities are computer based, and the data transactions arc protected by passwords. These passwords identify various entities such as bank accounts, mobile phones, etc. People might reuse the same password, or passwords might be related to an individual that can lead to dictionary attacks. Indeed, remembering several passwords become a tedious task. Biometrics is a science that measures individual physical characteristics in a unique way. Thus, biometrics serves as a method to replace the cumbersome use of complex passwords. Our research uses the features of biometrics to efficiently implement a biometric encryption system with high level of security.

[7] Rahul Gaikwad, Sangita K. Chaudhari, Sairaj Jadhav, (2021), "A Comparative Study on Fingerprint Matching Algorithms"

Abstract fingerprints are studied and analyzed from a long duration of time and it has been identified that it has a vital role to play in the upcoming and future applications. However matching two fingerprints is quiet a complex process and can go wrong due to different reasons or problems in the method used for matching. In this project, the authors compare the various fingerprint matching algorithms - direct matching, minutiae matching and matching based on Ratios of distance. Further, they test various datasets and identify which is the best out of the three algorithms, based on various parameters such as cost, time complexity and accuracy.

Chapter 3

Requirement and analysis

In this project, we attempt to deal with two basic attacks on the biometric system - **Masquerading and Substitution attacks**

1. Masquerading: Masquerade attack on biometric hashing, which reconstructs the original biometric image from the given hashcode, has been given much attention recently. It is mainly used to validate the security of biometric recognition system or expand existing biometric databases like face or iris scans.

2. Substituting: Substituting means tampering with the stored temlates in the database. A template represents a set of salient features that summarizes the biometric data (signal) of an individual. The templates can be modified to obtain a high verification score, no matter which image is presented to the system. The templates which are stored in the database can be replaced, stolen or even can be altered. Thus, bringing the system down by making the score low for legitimate users. The template-generating algorithms have been viewed as one-way algorithms.

There are few more attacks that can happen on a biometric system that are mentioned below:

3. Channel attack: In a so-called match-on-card system [26], a smart card (without sensor) receives the fingerprint image (or the extracted features) from an external reader and performs the comparison internally. In this case, the adversary can easily send chosen inputs to the card and measure classical side channels like power consumption.

4. Spoofing the Feature set: The replacing of the feature set with fake or altered features are called spoofing of data. These types of spoofing attacks are typically used to attack various networks, spread malware and to gain confidential information.

5. Replay Attack: In this attack, the data stream which is contained in the biometric system is injected between the sensor and the processing system. A replay attack can be of two to three stage process. It first intercepts or copies the sensor transmission, then it modifies or alters the information, thus finally replaying the data.

6. Trojan horse attack: In Trojan horse attack the feature extractor is itself replaced to produce the desired features and to add on those features in the existing database. The spoof detection technology has become a crucial part of a biometric system as with an increasing concern for security, the biometric attacks are to be identified, controlled and minimized. Researchers are developing various new approaches for a secure biometric system.

7. Overriding Yes/No response: An inherent error prevailing in your biometric systems is that the result of the system is always a binary response, Yes/No (i.e., either match/no match). In other words, there is still a fundamental disconnecting between the biometric and applications, which make the system, open to potential attacks.

3.1 Requirement of a secure biometric system

Biometric system is subjected to many malicious attacks which can be performed by various forms of threats. Malicious attacks on a biometric machine are a security concern and degrade the system's performances.Of particular concern when we talk about biometrics is the concept of informational privacy, referring generally to an individual's personal control over the collection, use and disclosure of recorded information about them, as well as to an organization's responsibility for data protection and the safeguarding of personally identifiable information (PII), in its custody or control. A lack of informational privacy can have profound negative impacts on user confidence, trust, and the usage of a given information technology, specific application or deployment, or even an entire industry.

3.2 Summary of the techniques studied to secure biometric authentication systems

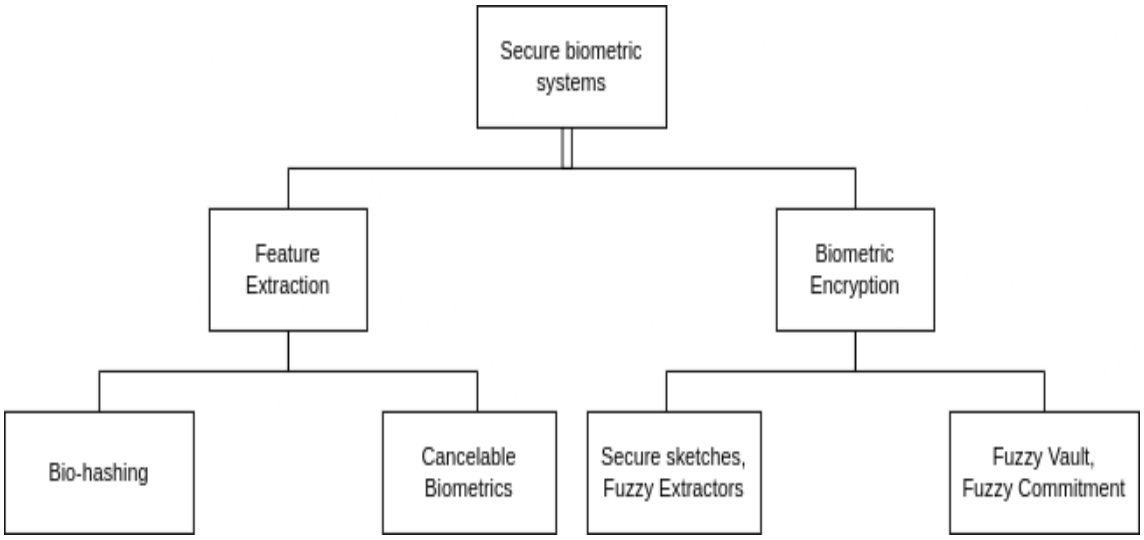


Fig3.1. Classification of methods used to secure biometric authentication systems

Through the years, there have been several proposals for authentication schemes that keep biometrics secure. Multimodal biometric authentication systems, that use more than one biometric (for example, fingerprint + retina scan) are considered more secure than unimodal biometric systems [3]. However, they complex and expensive. Fig.3.1 summaries the techniques studied to secure biometric authentication systems.

A. Feature Transformation Feature transformation involves applying an invertible or non-invertible transform to the biometric template.

1) Biohashing:

BioHashing is an invertible method which is based on transforming the biometric template using pseudo-random projections that are generated using a user-specified key or token[4]. However, biohashing techniques operate under the assumption that the TRN would never be lost, stolen, shared or duplicated, which is not generally the case. If it was possible to guarantee a secure TRN, there would be little use to combine biometrics as it would serve as a perfectly secure password by itself [5].

2) Cancelable Biometrics:

Another popular idea in feature transformation, which is non-invertible is Cancelable Biometrics. In this method, instead of storing the original biometric, it is transformed using a one way function. The transformation can be applied either in the original domain or in the feature domain. It provides privacy since it is computationally difficult to recover the original biometric from a transformed

one. [6]. The vulnerability of cancelable systems using compressive sampling, and against blind deconvolution systems are not known yet. Additionally, it is difficult to ensure that the transforms are repeatable [7].

B. Biometric encryption

A method that is gaining popularity over the last decade is the generation of a secure key from a biometric template. The secret key is derived from the fingerprint itself, and this key is used to encrypt the data.

1) Fuzzy Extractors and Secure Sketches:

Biometric data is noisy and often tough to process. This is made simpler using the help of some extra information called a sketch. [8]. Fuzzy extractors enable the generation of a secure cryptographic key from any noisy data [9]. Sahai and Waters propose a unique fuzzy identity based encryption scheme with substantial performance improvement [10]. However, fuzzy key generation does not address the need for the construction for efficient biometric identification schemes. Therefore, operating an authentication system in the identification mode using fuzzy sketches and extractors is computationally expensive [10].

2) Fuzzy vault and Fuzzy commitment

Key binding techniques monolithically bind a secret key and the biometric template in such a way that ensures it requires extreme computational ability to access the key or the template without the user's biometric input. Fuzzy commitment schemes combine both the techniques of cryptography and error correcting codes, so that the scheme is both binding and concealing. In such schemes, it is impossible to decommit a value in more than one way [11] Fuzzy vaults are suitable for applications that combine biometric authentication and cryptography. They provide security as well as user convenience [12]. The drawback with key binding methods is their high-time complexity.

In this project we are using the process similar to the method suggested by Joseph Mwema. In order to prevent attacks on point 6, J.Mwema proposes the encryption of the biometric template stored at the database using a key generated using another fingerprint template [2].

Chapter 4

Hardware and Software requirement

4.1 Hardware requirement

1. Arduino uno
2. R307 Fingerprint module

4.2 Hardware description

1. Arduino uno

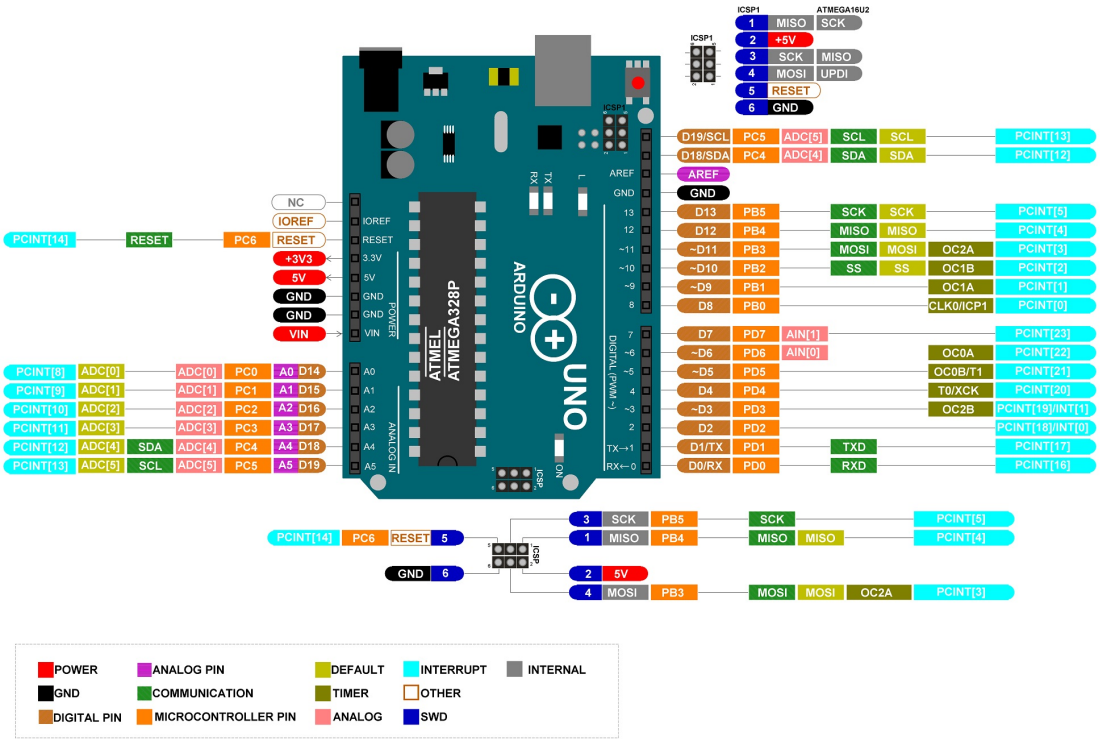


Fig4.1.Arduino uno pin configuration

The Arduino UNO is a standard board of Arduino. Here UNO means 'one' in Italian. It was named as UNO to label the first release of Arduino Software. It

was also the first USB board released by Arduino. It is considered as the powerful board used in various projects. Arduino.cc developed the Arduino UNO board. Arduino UNO is a microcontroller board based on the ATmega328P. It has 14 digital input/output pins (of which 6 can be used as PWM outputs), 6 analog inputs, a 16 MHz ceramic resonator, a USB connection, a power jack, an ICSP header and a reset button. The configuration of pins is shown in Fig4.1. It contains everything needed to support the microcontroller; simply connect it to a computer with a USB cable or power it with a AC-to-DC adapter or battery to get started. You can tinker with your UNO without worrying too much about doing something wrong, worst case scenario you can replace the chip for a few dollars and start over again.

Components of arduino uno:

ATmega328 Microcontroller - It is a single chip Microcontroller of the ATmel family. The processor code inside it is of 8-bit. It combines Memory (SRAM, EEPROM, and Flash), Analog to Digital Converter, SPI serial ports, I/O lines, registers, timer, external and internal interrupts, and oscillator.

ICSP pin - The In-Circuit Serial Programming pin allows the user to program using the firmware of the Arduino board.

Power LED Indicator - The ON status of LED shows the power is activated. When the power is OFF, the LED will not light up.

Digital I/O pins - The digital pins have the value HIGH or LOW. The pins numbered from D0 to D13 are digital pins.

TX and RX LED's - The successful flow of data is represented by the lighting of these LED's.

AREF- The Analog Reference (AREF) pin is used to feed a reference voltage to the Arduino UNO board from the external power supply.

Reset button - It is used to add a Reset button to the connection.

USB- It allows the board to connect to the computer. It is essential for the programming of the Arduino UNO board.

Crystal Oscillator - The Crystal oscillator has a frequency of 16MHz, which makes the Arduino UNO a powerful board.

Voltage Regulator - The voltage regulator converts the input voltage to 5V.

GND- Ground pins. The ground pin acts as a pin with zero voltage.

Vin- It is the input voltage.

Analog Pins - The pins numbered from A0 to A5 are analog pins. The function of Analog pins is to read the analog sensor used in the connection. It can also act as GPIO (General Purpose Input Output) pins.

Why is Arduino recommended over other boards?
The USB port in the Arduino board is used to connect the board to the computer using the USB cable. The cable acts as a serial port and as the power supply to interface the board. Such dual functioning makes it unique to recommend and easy to use for beginners.

Technical Specifications of Arduino UNO

There are 20 Input/Output pins present on the Arduino UNO board. These 20 pis include 6 PWM pins, 6 analog pins, and 8 digital I/O pins.

The PWM pins are Pulse Width Modulation capable pins.

The crystal oscillator present in Arduino UNO comes with a frequency of 16MHz.

It also has a Arduino integrated WiFi module. Such Arduino UNO board is based on the Integrated WiFi ESP8266 Module and ATmega328P microcontroller.

The input voltage of the UNO board varies from 7V to 20V.

Arduino UNO automatically draws power from the external power supply. It can also draw power from the USB.

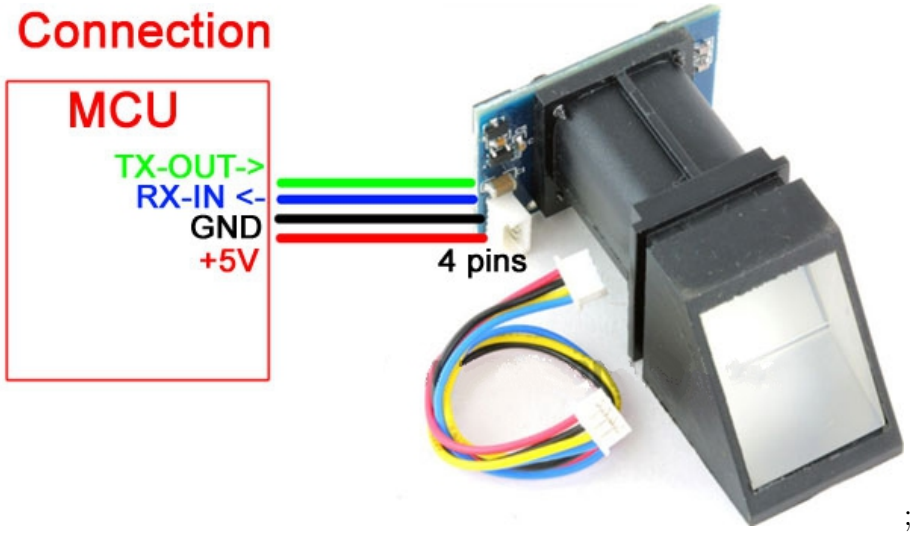
How to get started with Arduino UNO?

We can program the Arduino UNO using the Arduino IDE. The Arduino IDE is the Integral Development program, which is common to all the boards. We can also use Arduino Web Editor, which allows us to upload sketches and write the code from our web browser (Google Chrome recommended) to any Arduino Board. It is an online platform. The USB connection is essential to connect the computer with the board. After the connection, the PWR pins will light in green. It is a green power LED.

Memory of Arduino uno

The Arduino UNO has only 32K bytes of Flash memory and 2K bytes of SRAM. That is more than 100,000 times LESS physical memory than a low-end PC! And that's not even counting the disk drive! Working in this minimalist environment, you must use your resources wisely.

2.R307 Fingerprint module



;

Fig4.2 R307 Fingerprint module

Fig4.2 shows the R307 fingerprint sensor module. The R307 is used in biometrics for security in fingerprint detection as well as verification. These devices are mainly used in safes where there is a high-powered DSP chip used in the rendering of image, feature-finding, searching and calculation by connecting it to any microcontroller with the help of TTL serial, send data packets to get photos, notice prints, search and hash. The enrollment of new fingers can be stored directly within the flash memory of on board.

Features of Fingerprint Sensor

1. It includes image collection as well as chip algorithm.
2. The fingerprint reader can perform lesser growth and can be fixed into a range of end products.
3. Power use is low, excellent performance, small in size, and less cost.
4. Optical technology which is used is professional, and exact module developed techniques.

5. The capabilities of image processing are good, and can effectively capture pictures up to 500 dpi resolution.

Fingerprint Sensor Working Principle

The working principle of the fingerprint sensor mainly depends on the processing. The fingerprint processing mainly includes two elements namely enrollment and matching. In fingerprint enrolling, every user requires to place the finger twice.

So that the system will check the finger images to process as well as to generate a pattern of the finger and it will be stored. When matching, a user places the finger using an optical sensor then the system will produce a pattern of the finger compares it with the finger library templates.

For 1:1 fingerprint matching, the system will evaluate the exits finger with a precise pattern which is selected within the module. Similarly, for 1: N matching, the scanning system will look for the complete finger records for the finger matching. In both situations, the scanning system will go back to the corresponding result, success otherwise crash.

Specifications of fingerprint sensor

1. The fingerprint sensor is an optical type.
2. The interface is USB1.1/ TTL logical level (UART).
3. The speed of scanning is 0.5 sec.
4. The speed of verification is 0.3 sec.
5. The capacity storage is 1000.
6. The security level is 5
7. The baud rate is 4800BPS 115200BPS variable.
8. Current is typical 50 mA, and peak 80mA.
9. Fixed indicators-15KV bright green backlight.
10. The life of the sensor is 100 million times.
11. The corresponding technique is 1: N.
12. The dimension is 44.1 X 20 X 23.5mm.
13. The size of the character file is 256 bytes.
14. The FRR (False Rejection Rate) is ¡1.0
15. The FAR (False Acceptance Rate) is 0.001
16. Voltage is 4.2 to 6.0 VDC.
17. Operating surroundings temperature is -20° C to 40° C

4.3 Software requirement

1. Arduino IDE

4.4 Software description

1. Arduino IDE

;

Fig4.3. Arduino logo

The Arduino Integrated Development Environment (Fig4.3) - or Arduino Software (IDE) - contains a text editor for writing code, a message area, a text console, a toolbar with buttons for common functions and a series of menus. It connects to the Arduino hardware to upload programs and communicate with them. Programs written using Arduino Software (IDE) are called sketches. These sketches are written in the text editor and are saved with the file extension .ino. The editor has features for cutting/pasting and for searching/replacing text. The message area gives feedback while saving and exporting and also displays errors. The console displays text output by the Arduino Software (IDE), including complete error messages and other information. The bottom right-hand corner of the window displays the configured board and serial port. The toolbar buttons allow you to verify and upload programs, create, open, and save sketches, and open the serial monitor.

Chapter 5

Encryption of Biometric Templates for Secure Authentication

5.1 Proposed Methodology

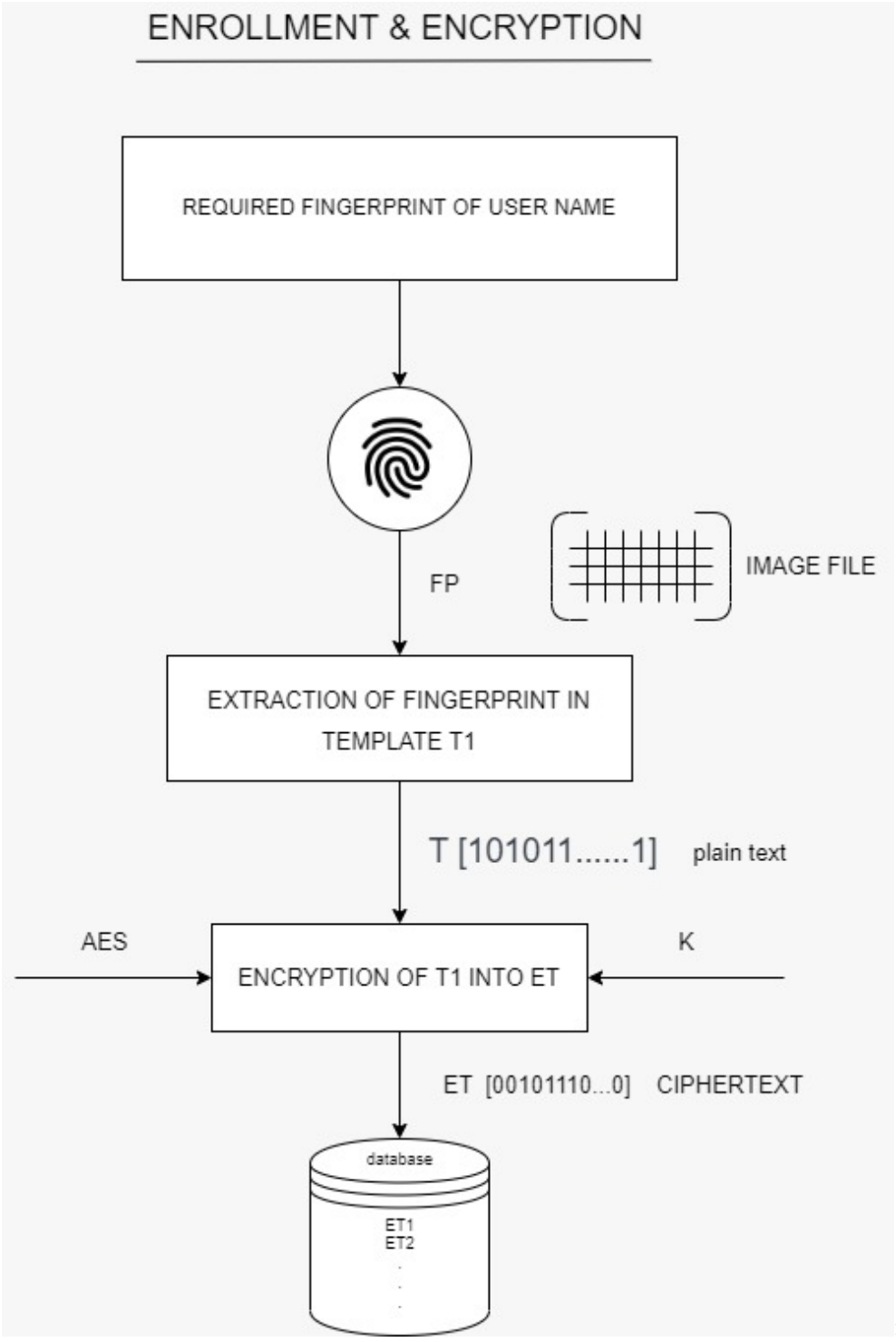
In this project, we propose a design for a secure authentication system that operates in the identification mode. The functioning of this system may be divided broadly into two phases - the enrollment and encryption phase, then the decryption and identification phase. Both phases work one after the other to achieve the goal of a secure authentication system. A generic authentication systems consists normally of enrollment and verification/identification. Our system is unique in that it utilizes the benefits of cryptography, bu using an encryption scheme, to ensure additional security.

Our proposal is different from other authentication systems by virtue of the encryption and decryption steps introduced. This additional layer of security becomes all the more important and appreciable in a commercial setting, like in banks. The purpose of encryption is to prevent any tampering with the fingerprint template, which is otherwise quite vulnerable to attacks.

Below, the system flow diagram of the proposal is elaborated on in detail. The phases are **Enrollment and encryption**, and then **Decryption and matching**.

These phases work sequentially, but in tandem, to ensure enhanced security.

5.2 System flow diagram



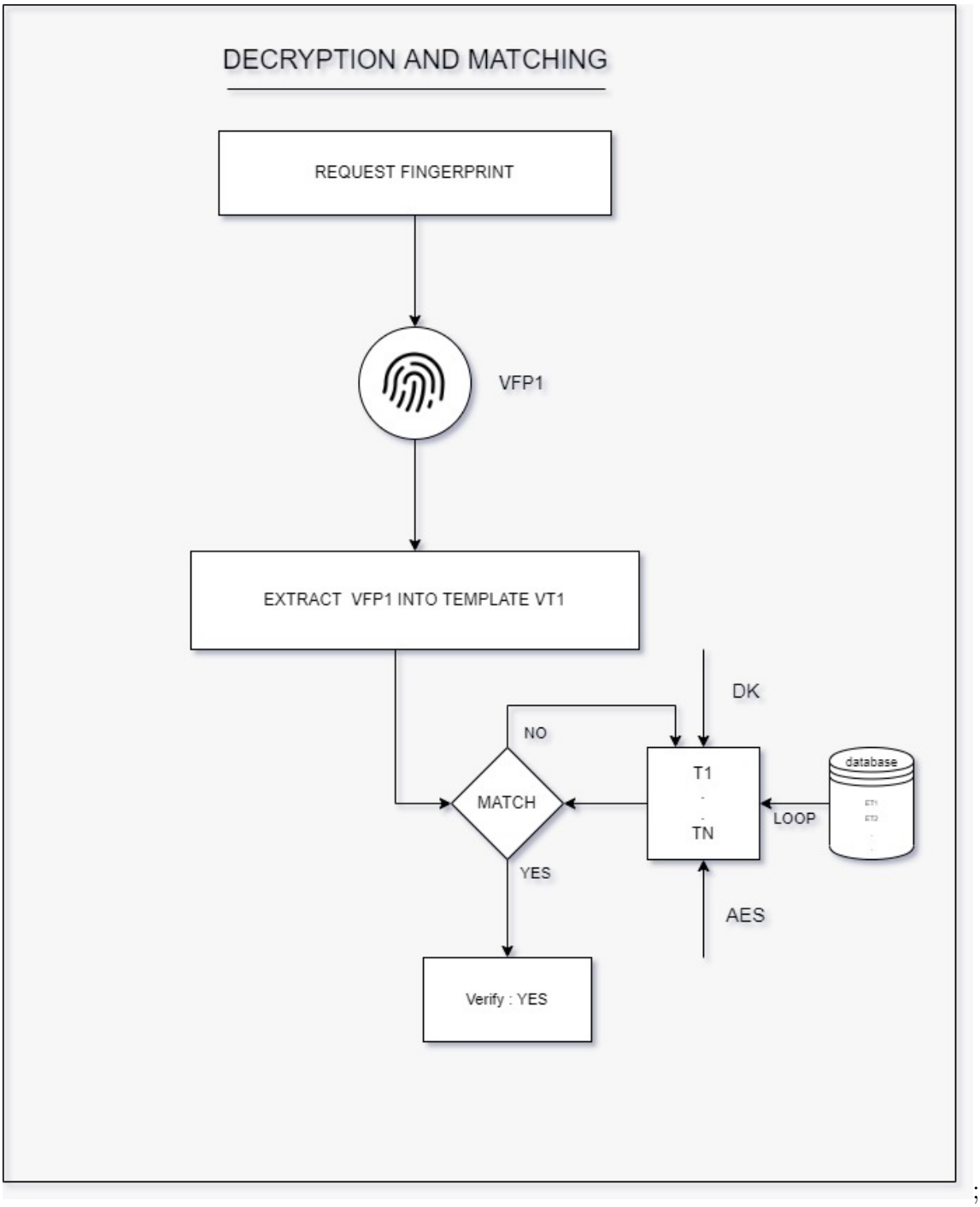
;

Fig5.1. The flow of enrollment and encryption

a) Enrollment of fingerprint and encryption of the template

Fig5.1 demonstrates the first phase of the system's operation: Enrollment of a new fingerprint, followed by its encryption and storing in the database. The user

inputs their fingerprint fp1 via the fingerprint sensor module. The fingerprint is received as a raw image file, which is then extracted to a template t1. Template t1 is a plaintext string which acts as the input to the encryption scheme. In this implementation, we use AES. The secret key K is generated using the hardware random number generator. The output of the encryption is et1, which is stored in the database along with other encrypted fingerprint templates.



;

Fig5.2. The flow of decryption and matching

b) Decryption of template and matching of fingerprint

This phase begins with a request for the user's fingerprint. The input fingerprint vfp1 is extracted onto a template vt1. A loop is implemented, which fetches an encrypted fingerprint etn from the database, decrypts it using secret key K with AES, and performs a comparison to establish identification. This loop executes until the decrypted template t1 is found as a match for vt1. If no match is found, the decision NO is output and the user is denied access. Fig5.2. demonstrates the decryption and matching phase.

5.2.1 Description of the components of the authentication system

A conventional fingerprint authentication system has 3 main components - sensing, feature extraction, and matching. Our proposed system has the extra steps of encryption and decryption.

5.2.1.1 Extraction of fingerprint on to template

a) Sensing

The sensing element is the most important part of the fingerprint module. In general, sensors can be divided into three categories - optical sensors, solid state sensors, and ultrasound sensors. There are several parameters that characterise digital raw fingerprint images, such as[1]:

- Resolution
- Area
- Number of pixels
- Depth
- Geometric Accuracy
- Image quality

It is essential to ensure the image is satisfactory in all these areas.

The fingerprint module used in this implementation is a livescan, optical fingerprint sensor with TTL UART interface. It is a 500 dpi sensor with Fake rate (FAR) less than 0.001 percent Refusal rate (FRR) less than 1.0 percent.

b) Feature Extraction

• Ridges and valleys: Ridges are the dark lines while valleys are the light ones. They

may run parallel to each other, bifurcate, or terminate.

• Ridgelines give rise to singular regions like loops, deltas, and whorls.

• Core: Also known as the registration point, it is situated at the centre of the north most loop singularity. It is not always straightforward to locate it due to the variability of fingerprints.

• Minutiae: While ridges, valleys, cores, and singularities are analysed as global markers, locally, we analyse minutiae which refer to various ways that the ridges bifurcate or terminate.

c) Pre and post-processing

To ensure a greater success rate, noise caused by cuts, scars, dryness, and wetness needs to removed, and ridges must simultaneously be enhanced There are three steps involved - Pre-processing, Minutia extraction, and Post-processing [13]

1) **Pre-processing**

Pre-processing is done in two steps, Image Enhancement and Image binarisation/segmentation. Image enhancement is done using the methods of histogram equalization and Image filtering. This is followed by noise removal.

Image segmentation is the process of isolating the fingerprint from the background.

2) **Minutia Extraction**

Extraction of minutia points is done following ridge thinning. There are three methods available to accomplish this:

• Minutiae extraction technique.

• Pattern matching or ridge based technique.

• Correlation method.

3) **Post-processing**

During minutia detection, there occurs the accumulation of errors in the form of false minutiae being detected. Hence, false Minutia Removal is performed as a last step in feature extraction

5.2.1.2 Encryption and Decryption

Encryption and decryption are performed using AES 256 in CBC mode. The Advanced Encryption Cipher is a substitution permutation network with key lengths 128, 192, or 256. There are 4 main steps to the encryption process - byte substitution, shifting rows, mixing columns, and addition of the round key. Decryption includes the inverse of the first three transformations followed by the addition of a round key. Fig5.3 illustrates the encryption and decryption process using AES.

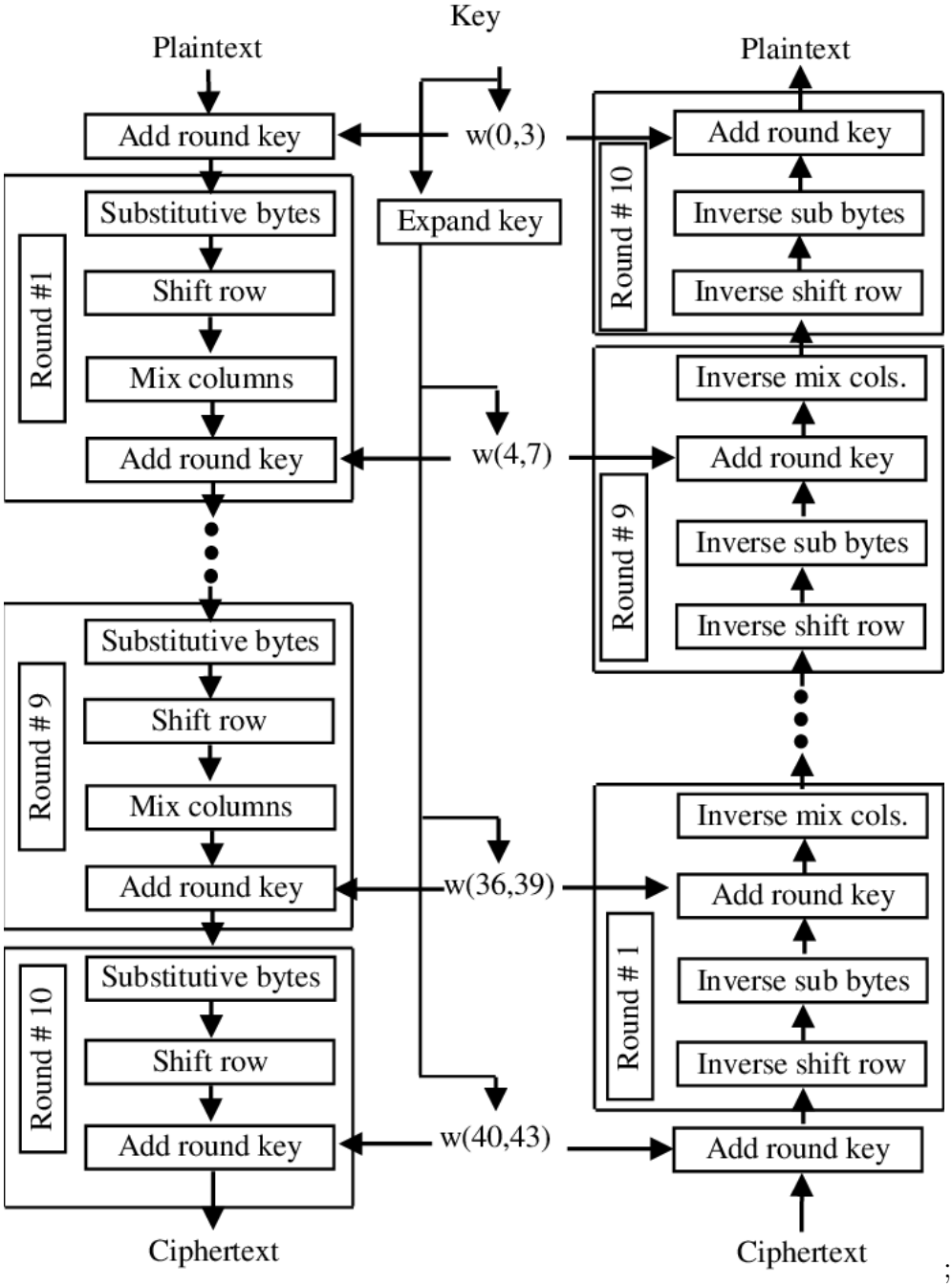


Fig5.3. The encryption and decryption process for AES [15]

Key feature of AES:

Security: Competing algorithms were to be judged on their ability to resist attack as compared to other submitted ciphers. Security strength was to be considered the most important factor in the competition.

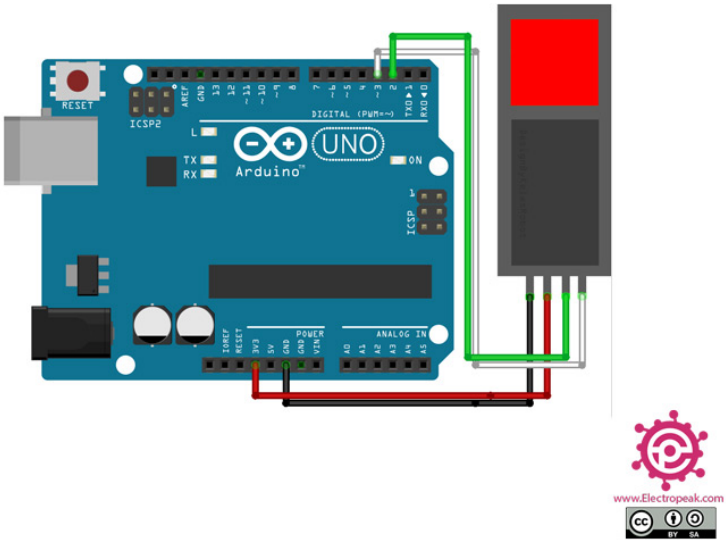
Cost: Intended to be released on a global, nonexclusive and royalty-free basis, the candidate algorithms were to be evaluated on computational and memory efficiency.

Implementation: Factors to be considered included the algorithm's flexibility, suitability for hardware or software implementation, and overall simplicity.

5.2.1.3 Matching

In order to perform verification or identification, an algorithm that compares the similarity between two fingerprint templates is required. Fingerprint matching algorithms may either return a binary decision (yes or no), or might output a degree of similarity between 0 and 1. The process of matching is made difficult by certain factors, like a high degree of displacement or rotation that causes the fingerprint area fingerprint area to fall outside the sensor's "field of view," different pressure and skin condition, nonlinear distortion, noise from the fingerprint sensor, and errors made during the feature extraction. There's a selection of fingerprint matching algorithms used, the most popular of which are correlation based, or minutia based. Given the superior performance of minutia based matching [14], it is selected for this project.

5.3 Implementation



;

Fig5.4. Interfacing of Arduino Uno with R307

Fig5.4 demonstrates the interfacing between Arduino Uno and R307 fingerprint template. Fingerprint sensor includes pins like DNC, VCC, TX, RX, and GND. These pins are connected through different colored connecting wires. Each color wire is used to indicate each pin of the sensor.

DNC pin is connected by a white wire.

VCC pin is connected by a red wire.

TX pin is connected by a blue wire.

RX pin is connected by a green wire.

GND pin is connected by a black wire.

The connection of the fingerprint sensor module to an Arduino board can be done like the following.

1.The black wire is connected to the GND pin of the Arduino.

2.The red wire is connected to the 5V of the Arduino.

3.Green wire is connected to the digital pin-2 of the Arduino.

4.The white wire is connected to the digital pin-3 of the Arduino.

First of all the project code of this project requires different libraries namely the

Adafruit Fingerprint, the Adafruit GFX the Sumotoy's for the display.

Take an enroll example code and upload it into an Arduino board. Go to File Examples Adafruit Fingerprint Sensor Library Enroll.

By using this code, the fingerprints can be stored within the FLASH memory of the device. Once the serial monitor opens then it asks to enter the credentials to register.

5.4 Demonstration:

Here are the steps carried out by a user in order to use our biometric authentication system:

1. Place fingerprint on sensor
2. Fingerprint is accepted
3. Fingerprint template is generated
4. Fingerprint template is extracted
5. Fingerprint template is encrypted and stored
6. Place finger on sensor for identification
7. Stored fingerprint templates are successively decrypted
8. Fingerprint templates are matched
9. Decision MATCHED or NO MATCH FOUND is output

Chapter 6

Results and discussion

6.1 Output

Enrollment of a fingerprint

```
Fingerprint template extractor
Found fingerprint sensor!
----------------------------------
Attempting to load #1
Template 1 loaded
Attempting to get #1
Template 1 transferring:
495 bytes read.
Decoding packet...
030153197C00FFFEFFFEFFFEE1FE00FE00FE00FE00FE00FE00FE00FE00FE00FE003E000E000600060000000000000
000000000000000055342F82DB382DD9383843D9392D7EEF01FFFFFFFF0200820F4159D73837DB4E15405A94349354
523B96DF4C3518DAD22DB6047200000000000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
0000000000000000000000000000000000000B9AEF01FFFFFFFF0200820301531D8400003E003E001E001E001
E001E001E001E001E001E801E801E801E801E801E800E8000000000000480801DE1C0E05DE299E05DE4E1FC05E41
2703FE4CBA02BE5EC2195E46C31A3E5043599E4F8D8137400DC2FF081086F72C16053F1B22C61F43A301FF1EC2405
B3843D95B4EB5C2192201EF01FFFFFFFF08008229B7D9D732C3D7572BC357941CB35C1037BA43B014BA5B4A274016
CE2FB51BEF3EB7174824B3C38A243D974B000000000000000000000000000000000000000000000000000000000
000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000000
00000000000000139CFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF
F
done.
```

Fig6.1. Extraction of fingerprint template

Fig.6.2 and Fig.6.3 demonstrate the enrollment and encryption of the fingerprint. The process begins by selecting an ID for the fingerprint the user is about to enroll. The sensor obtains an image of the finger placed on it. For further use, this image is encoded as a plaintext template. We first extract this plaintext template. This extracted plaintext is shown in Fig6.1.
In order to ensure that the template is captured correctly, we run the enrollment function twice. There are two different cases that may follow.

```
Ready to enroll a fingerprint!
Please type in the ID # (from 1 to 6) you want to save this finger as...
Enrolling ID #1
Waiting for valid finger to enroll as #1
.
.
.
.
.
.
.
Image taken
Image converted
Remove finger
ID 1
Place same finger again
.............Image taken
Image converted
Creating model for #1
Fingerprints did not match
```

Fig6.2. Mismatch of fingerprints meant to be enrolled

Fig6.2 illustrates case 1, in which the fingerprints submitted for enrollment are not the same. In the event of a mismatch, both templates are immediately discarded.

Fig6.3 illustrates case 2, in which both instances of the fingerprints are the same. Then, the second template is retained, encrypted using AES, and successfully stored under the selected ID. This marks the completion of enrollment; the fingerprint template for that particular ID now exists in the template database.

```
Ready to enroll a fingerprint!
Please type in the ID # (from 1 to 6) you want to save this finger as...
Enrolling ID #1
Waiting for valid finger to enroll as #1
.
.
.
.
Image taken
Image converted
Remove finger
ID 1
Place same finger again
.....Image taken
Image converted
Creating model for #1
Prints matched!
ID 1
Encrypting
Image encrypted
Stored!
```

Fig6.3. Fingerprint successfully enrolled under ID 1

6.1.0.1 Decryption and identification

We begin this phase having stored our fingerprint templates in the database. Fig6.4 and Fig6.5 describe the decisions for match and non-match.

```
Waiting for valid finger...
Sensor contains 11 templates
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
Image taken
Image converted
Decrypting ID 1
Decrypting ID 2
Decrypting ID 3
Decrypting ID 4
Decrypting ID 5
Found a print match!
Found ID #66 with confidence of 130
```

Fig6.4. Fingerprint successfully identified

When the finger is placed on the sensor, its template is extracted. Then, one by one, the templates in the database are fetched, decrypted, and matched against the new fingerprint template. If there is a positive match, the ID number is output. We also see the confidence with which the match has taken place. This is demonstrated in Fig.6.4, where the fingerprint input is identified as a previously enrolled template.

```
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
No finger detected
Image taken
Image converted
Decrypting ID 1
Decrypting ID 2
Decrypting ID 3
Decrypting ID 4
Decrypting ID 5
Did not find a match
```

Fig6.5. Fingerprint not matched

If the fingerprint input is not previously enrolled, the algorithm outputs a message informing us that there is no match. This can be seen in Fig.6.5.

6.2 Evaluation metric

The performance of our design has been evaluated on the following criteria:

- Accuracy = (TP+TN)/(TP+FP+FN+TN)...(1)
- Precision = TP/(TP+FP).....(2)
- Sensitivity = TP/(TP+FN).....(3)
- Specificity = TN/(TN+FP).....(4)
- F-Score = 2*Recall*Precision/Recall+Precision.....(5)

6.3 Observations

Experimental procedure:
Before implementing our design, we used the in-built matching algorithm provided by the fingerprint sensor and performed enrolment and verification. This process was repeated 50 times. The observations were recorded in a confusion

matrix.
Then, we implemented our encryption scheme and performed enrollment and verification. This too was repeated 50 times. These observations too were recorded in a confusion matrix. The confusion matrices are shown in Fig6.3.

	Output match	Output non-match
Real Match	TP 24	FN 2
Real Non-match	FP 1	TN 23

	Output match	Output non-match
Real Match	TP 22	FN 5
Real Non-match	FP 3	TN 20

;

Fig6.3. Confusion matrices of R307's authentication system (left) vs our proposed authentication system (right)

6.4 Accuracy

Our system is has an accuracy of **84 percent**.
Accuracy = (TP+TN)/(TP+FP+FN+TN)
Fingerprint verification was performed 50 times out of which 42 matches were true positives and true negatives.
Accuracy of R307: 94 percent
Accuracy of our design: 84 percent

6.5 Precision

Our system is has a precision of **88 percent**.
Precision = TP/(TP+FP)
Fingerprint verification was performed 50 times out of which 22 matches were true positives.
Precision of R307: 96 percent
Precision of our design: 88 percent

6.6 Sensitivity

Our system is has a sensitivity of **81.4 percent**.
Sensitivity = TP/(TP+FN)
Fingerprint verification was performed 50 times out of which 22 matches were true positives.
Sensitivity of R307: 92.3 percent
Sensitivity of our design: 81.4 percent

6.7 Specificity

Our system is has a Specificity of **86.9 percent.**
Specificity = TN/(TN+FP)
Fingerprint verification was performed 50 times out of which 20 matches were true positives.
Specificity of R307: 95.8 percent
Specificity of our design: 86.9 percent

6.8 F-score

Our system is has an **F-Score of 0.84**.
F-score = 2*Recall*Precision/Recall+Precision
F-Score of R307: 0.94
F-Score of our design: 0.84

Chapter 7

Conclusion

The main purpose of this project was to develop a fingerprint authentication system that was more secure. We proposed a system that encrypts biometric templates, which would have otherwise been stored in a database as plaintext strings. Encryption of the fingerprints reduces the possibility of successful attacks on the channel or the database, while not affecting the speed or accuracy of the system.

7.1 Limitations of the design:

While we have achieved what we set out to accomplish, there are still certain limitations in our designs that are worth addressing in the future. Some of these limitations are listed below:

7.1.1 Encryption scheme:

In this design we use AES in ECB (Electronic Code Book) mode. In ECB mode, the plaintext is encrypted in blocks, whereby the blocks are independent of each other. As a result, identical plaintext blocks result in identical ciphertext blocks. So you can identify by the ciphertext alone, where the same blocks exist. This weakness is especially easy to see in the encryption of image templates. Hence, the scheme used is not completely secure.

7.1.2 Errors:

Our design is not error free and is prone to false positives and fasle negatives in the matching phase. This is reflected in the accuracy, precision, and related scores.

7.1.3 Time efficiency:

A single enroll and verify cycle takes a non-trivial amount of time, which is a disadvantage when it comes to commercial applications. Authentication and access control in commercial cases prioritize speed of the authentication system. While our system has takes time comparable to other industry standard authentication systems, it can be made quicker.

7.1.4 Security against different kinds of attacks:

Our system is secure against tampering and masquerading, but falls short when it comes to channel attacks, spoofing, trojan horses, substitution attacks and replay attacks. Our approach does not cater satisfactorily to the prevention of these types of attacks.

7.2 Further Scope

1. We propose a system for the identification mode of authentication systems. A proposal for verification mode using a similar approach of encryption of fingerprint templates may be developed.
2. More secure encryption schemes like RSA, ECC, or AES in CBC mode may be used to ensure confidentiality and integrity.
3. The key used in the encryption scheme may be generated using fuzzy extractors or other methods that allow for more secure keys.
4. Speed improvements may be made by a better choice of encryption and matching algorithm.
5. The system may be extended to be made multimodal, by integrating another biometric like retinal scan or voice recognition.

Bibliography

[1] Maltoni, D. (2005). "A Tutorial on Fingerprint Recognition". In: Tistarelli, M., Bigun, J., Grosso, E. (eds) Advanced Studies in Biometrics. Lecture Notes in Computer Science, vol 3161. Springer, Berlin, Heidelberg.

[2] Ratha, N.K., Connell, J.H., Bolle, R.M. (2001). "An Analysis of Minutiae Matching Strength". In: Bigun, J., Smeraldi, F. (eds) Audio- and Video-Based Biometric Person Authentication. AVBPA 2001. Lecture Notes in Computer Science, vol 2091. Springer, Berlin, Heidelberg.

[3] Mwema, J. M. (2015). "Encryption of Biometric Fingerprint Templates Using Encryption Keys Obtained from other Biometric Fingerprint Templates"

[4] B. Topcu, H. Erdogan, C. Karabat and B. Yanikoglu (2013). "Biohashing with fingerprint spectral minutiae," 2013 International Conference of the BIOSIG Special Interest Group (BIOSIG), pp. 1-12.

[5] Kong, Adams Cheung, King-Hong Zhang, David Kamel, Mohamed S. You, Jane. (2006). "An analysis of BioHashing and its variants". Pattern Recognition. 39. 1359-1368. 10.1016/j.patcog.2005.10.025.

[6] V. M. Patel, N. K. Ratha and R. Chellappa (2015), "Cancelable Biometrics: A review," in IEEE Signal Processing Magazine, vol. 32, no. 5, pp. 54-65,

[7] G. I. Davida, Y. Frankel, and B. J. Matt (1998) , "On enabling secure applications through off-line biometric identification," IEEE Symposium on Security and Privacy

[8] Li, Qiming Sutcu, Yagiz Memon, Nasir. (2006). "Secure Sketch for Biometric Templates". Adv. Cryptology Asiacrypt

[9] Dodis, Yevgeniy Reyzin, Leonid Smith, Adam. (2004). "Fuzzy Extractors: How to Generate Strong Keys from Biometrics and Other Noisy Data". Computing Research Repository - CORR. 38. 523-540. 10.1137/060651380.

[10] LI, N., Guo, F., Mu, Y., Susilo, W. Nepal, S. (2017). "Fuzzy Extractors for Biometric Identification". 37th IEEE Internaitonal Conference on Distributed Computing Systems (ICDCS 2017) (pp. 667-677). United States: IEEE.

[11] A. A. Al-Saggaf, (2021) "A Post-Quantum Fuzzy Commitment Scheme for Biometric Template Protection: An Experimental Study," in IEEE Access, vol. 9, pp. 110952-110961

[12] A. Juels and M. Sudan (2006) “A fuzzy vault scheme,”Des. CodesCryptography, vol. 38, no. 2, pp. 237–257

[13] Jayapal, Ranjith Pramod,. (2016). ”Biometric encryption system for increased security”. 1-3. 10.1109/CCST.2016.7815700.

[14] Rahul Gaikwad, Sangita K. Chaudhari, Sairaj Jadhav (2021) ”A Comparative Study on Fingerprint Matching Algorithms”, (IJERT) NTASU – 2020 (Volume 09 – Issue 03),

[15] Stallings, Cryptography and Network Security: Principles and Practice, 7th edition, 2017