SMGroup

SM Journal of Biometrics & Biostatistics

Research Article

Design and Analysis of Secure and Time-Space Efficient Models for Encoding of Four-Dimensional Multimedia Data in Biometric Security Systems

E J Garba*, P O Odion, A E Evwiekpaefe and F Ajakaiye

Department of Computer Science, Nigerian Defence Academy, Kaduna, Nigeria

Article Information

Received date: Aug 02, 2019
Accepted date: Aug 30, 2019
Published date: Sep 02, 2019

***Corresponding author**

E J Garba, Nigeria, Department of Computer Science, Nigerian Defence Academy, Kaduna, Nigeria, Tel: 234 (0) 803 694 3881; Email: ejgarba@nda.edu.ng

Keywords Biometric system, Enrollment, Four-dimensional multimedia data, Time-space complexity, Computational models.

Abstract

There are acceptable standards for designing biometric security systems that capture unique biological features and characteristics of individuals. Such standards guarantee global best practice especially with regards to the enrollment process. However, while most designers focus on the time-space efficiency of the biometric system, security of the system should be of utmost concern while the integrity of the data is not compromised. Note that a typical biometric system could be hacked at various points during the enrollment process. Therefore, this paper presents eight models for encoding of multimedia data in biometric security systems that have been proposed. These models were designed and simulated using Jet Brains IntelliJ IDEA as the development environment. These models are expected to make the enrollment process secure with reduced space consumption while keeping the time complexity optimally small. The results from the simulation were empirically analyzed with respect to algorithm steps, execution time, space, security level and quality. It was noticed that it is possible to achieve highest level of data integrity and security – though at the expense of increase in computation complexity. For instance, encryption is responsible for 100% rise in the execution time (that is from 0.69ms to 1.47, 1.48 and 1.32ms). While the models could be realized via any programming language of choice, the simulation source code written in Java programming language could be adopted as software framework by designers and developers of biometric security systems.

Introduction

The current trends in societal evolution of Smart City, Internet-of-Things (IoT), Cloud Computing, Big Data and Artificial Intelligence have provided the basis for need to use authentication systems based on biometrics. This implies that in the near future, biometrics systems will be everywhere in the society – in government, education, communities, banks, etc. However, biometrics systems are vulnerable to attacks that threaten the security and privacy of data captured. Worthy of note is the fact that the classic cryptographic algorithms are not sufficient to assure a strong level of security and privacy. The complexity of the algorithms and the time processing involved in achieving security and integrity of the data is quite challenging for designers and developers of biometric systems [1].

The twenty-first century spearheaded the emergence of biometric technologies. Increased concerns about national security and the tracking of individual's vis-à-vis migration and immigration necessitated the linking of personal identification details with biometric data. This is aimed at fighting crimes, insurgencies and terrorism. Therefore, biometrics is the process of identifying and recognizing people based on their unique behavioral characteristics (such as signature) and biological traits (such as fingerprint). Biometric systems strongly rely on the fact that every person possesses distinctive physical and behavioral features. The operational process of a typical biometric system is shown in (Figure 1) [2]. However, in this paper, we are concerned with the time and space efficiency of capturing and processing of the biometric data into reference format for optimal storage and retrieva.

Biometric systems are vulnerable to a variety of attacks that could jeopardize and compromise the enrolment, pre-processing, storage, retrieval, identification and authentication processes (Figure 2). Some of the threats identified are [3]:

a) **Fake biometric:** fake finger skin made from silicon, eye lens embedded with fake iris texture.

b) **Replay Attack**: an intercepted biometric data is submitted to the feature extractor.

c) **Sensor Bypassing:** it is possible to replace the feature extractor function with malicious codes that generate a false biometric template.

OPEN ACCESS

ISSN: 2573-5470

How to cite this article Garba EJ, Odion PO, Evwiekpaefe AE and Ajakaiye F. Design and Analysis of Secure and Time-Space Efficient Models for Encoding of Four-Dimensional Multimedia Data in Biometric Security Systems. SM J Biometrics Biostat. 2019; 4(1): 1038.

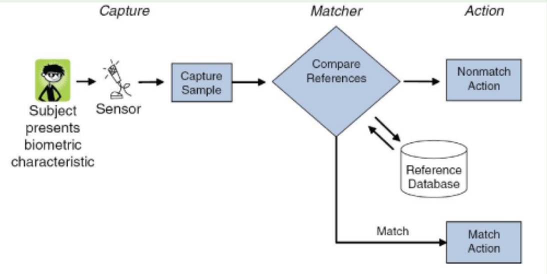


Figure 1: Operation of a biometric system (Source: [2]).

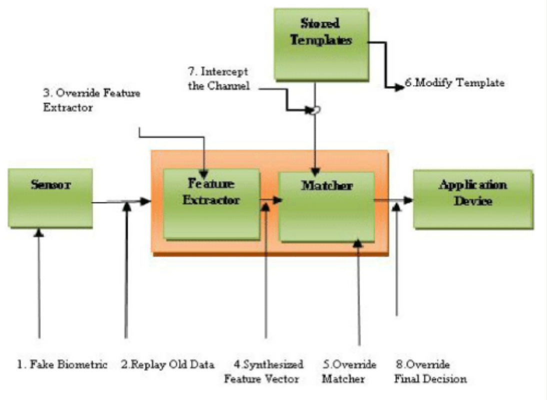


Figure 2: Possible attacks in biometric system (Source: [4]).

Figure 3: Fingerprint image in raw data format (bitmap)

d) **Biometric Data Replacement:** Genuine values of the biometric features in the reference template are replaced with synthetic values or real values (from a different individual).

Table 1: Four-dimensional multimedia data

Media Type	Media Expression		
	Elaboration	**Representation**	**Abstraction**
Text	Free text, Sentences, Paragraphs	Bold, italics, bullets, underlines, headlines, subheads	Shapes, icons
Graphics	Photographs, renderings, scanned images	Blueprints, Schematics	Icons
Sound	Speech, audio transcipts	Intensity, tone	Sound effects
Motion	Raw film footage	Animation, time-lapsed photography	Animated models, highly edited video

e) **Matcher Replacement**: the most critical function that carries out identification or authentication could be swapped with a malicious program code, such as Trojan horse.

f) **Database Attack**: the biometric database that stores the reference templates could be attacked with the sole aim of making modifications such as additions, deletion, replacements etc.

g) **Transmission Attack**: The reference templates could be tampered with when transmitted between the database and matcher.

h) **Matcher Result Overriding**: The result from the identification or authentication process could be overridden and then the status changed from accepted to rejected or vice versa.

Therefore, this research seeks to address the issues of security of biometric data with respect to data integrity (quality) and computational complexity associated with the enrollment process.

Related Work

Biometric Enrollment Process: According to Jaiswal et al [4] biometrics provides a more reliable system of authentication than identification cards, keys, passwords because; many biological traits (as well as physical and behavioral characteristics) are unique to an individual. When an individual's biometric data is captured and stored for the first time, this process is regarded as enrolment [5]. During the enrolment process, the capturing device (with an embedded

Table 2: Categorization of multimedia data and biometric features

Multimedia Data	Biometric Feature	Data Format	
		Raw	**Compressed**
Text	characters, words, sentences, paragraphs and passages	txt, doc, docx, rtf	pdf, ps, htm
Images	face, iris, fingerprint, palm, signature, handwriting	bmp	jpg, gif, png, tiff
Audio	speech, voice	Wav, aiff	mp3
Motion	videos	avi	mp4, mpeg

Citation: Garba EJ, Odion PO, Evwiekpaefe AE and Ajakaiye F. Design and Analysis of Secure and Time-Space Efficient Models for Encoding of Four-Dimensional Multimedia Data in Biometric Security Systems. SM J Biometrics Biostat. 2019; 4(1): 1038.

Table 3: Image size according to various standards including ISO [21]

Feature	Biometric Data Format
Fingerprint	8-bit 256 gray-level “bmp” format (size varies according to capturing sensors. See Table 4).
Face	24-bit bit depth “bmp” files and the size of image is 640×480 pixels
Iris	8-bit bit depth 256 gray-level “bmp” format with 768 × 576 pixels in size

Table 4: Image sizes captured by different biometric sensors

Sensor	Image Size
AES2501 Swipe fingerprint scanner	Not fixed
FPR620 optical fingerprint scanner	256*304
FT-2Bu capacitance fingerprint scanner	152*304
URU4000 optical fingerprint scanner	294*356
ZY202-B optical fingerprint scanner	400*400

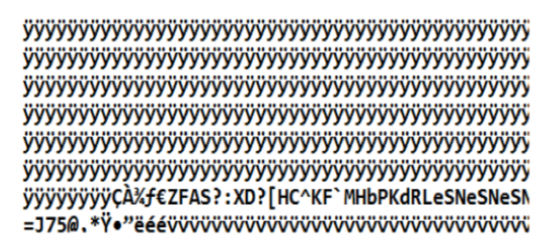


Figure 4: Binary (byte array) format of the raw fingerprint. This supports the philosophy of “hiding in plain sight”.

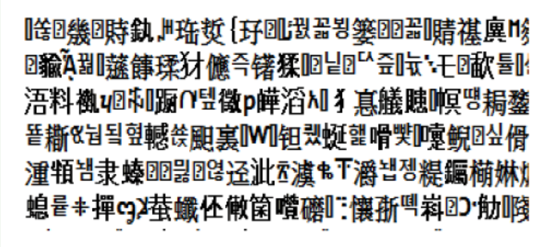


Figure 5: Compressed byte array of the fingerprint.

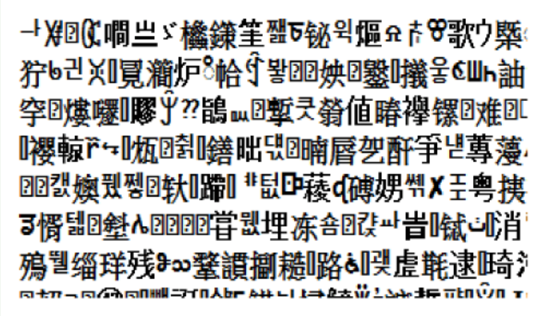


Figure 6: Encrypted compressed byte array

sensor) acquires the appropriate biometric data. The sensor serves as an interface between the real world and the biometric system. A good sensor is expected to eliminate background noise as much as possible before the biometric features are extracted. These features are formatted to create a template that consists of only relevant biometric traits and behavioral characteristics of the individual [6].

Once the raw biometric data is captured, the data is further processed into the reference format (vis-à-vis the data structure and file type) for storage into the database. However, due to different standards and varying system specifications, some of the reference formats are too large or complicated for efficient storage and high-throughput computer processing. As part of the strategies to overcome the aforementioned challenges, some biometric systems do not store the raw biometric data but rather extract key relevant features through techniques such as mathematical abstraction and modelling. Furthermore, another alternative is to compress the raw biometric data into a new format with rearranged data structure – at the expense of losing the original quality though! Once the biometric data is converted into the appropriate reference format, the enrolment process of the new person is completed by storing the format into the database [2].

Multimedia and Biometric Data: Multimedia data in biometric systems simply consist of text, images, video, clips, animations, graphics and audio (Table 1) [7]. For biometric security systems, this research considers the following categorization as shown in (Table 2). For the purpose of evaluation of the proposed models, fingerprint image is used herein. There are so many globally accepted standards vis-à-vis biometric data formats. (Table 3) summarizes some biometric

Table 5: Nigerian biometric enrolment standard [22]

Feature	Biometric Data Format	Remark
Face	24-bit bit depth colour images with a minimum of 90 pixels (recommended is least 120 pixels)	In order to preserve the quality of image, only uncompressed images are stored. Images with JPEG 2000/ WSQ lossless compression ratio of 10.
Iris	8-bit image with minimum of 140 pixels. 170 pixels is recommended for optimum quality	The iris images are to be stored according to ISO standard format using either JPEG 2000 or PNG lossless compression.
Fingerprint	24-bit colour images with a minimum optical resolution of 500 pixels is recommended	JPEG2000 compression is recommended up to 15 compression ratio. In order to preserve quality, uncompressed images are also accepted.

Table 6: Security levels during the encoding and decoding processes

Security Level	Description
0	Biometric data in either raw format or data structure compressed format.
1	Conversion of the raw or compressed formats to byte array. This is the lowest form of security that supports the philosophy of "hiding in plain sight".
2	Compression of the byte array in security level 1.
3	Encryption of the compressed byte array in security level 2.
4	Encryption of the filename of the encrypted compressed byte array in security level 3.
5	File compression of the biometric data in security level 4.

Table 7: Performance evaluation of the computational models

Model	Algorithm Steps	Execution Time (Sec)	Space (KB x 10^2)	Security Level	Quality
1	4	1.26	5.76	0	1
2	6	0.65	2.03	0	0.9
3	6	6.62	5.76	1	1
4	8	0.69	1.20	2	1
5	10	1.47	1.20	3	1
6	12	1.48	1.20	4	1
7	14	1.32	1.20	5	1
8	14	0.40	0.61	5	0.9

data formats as presented by [8]. The Nigerian biometric standard for enrolment is summarized as shown in (Table 5) [9]. Sang et al [10] have observed that face recognition is significantly influenced by the quality of the image. According to ISO/IEC standard 19794-5, the face

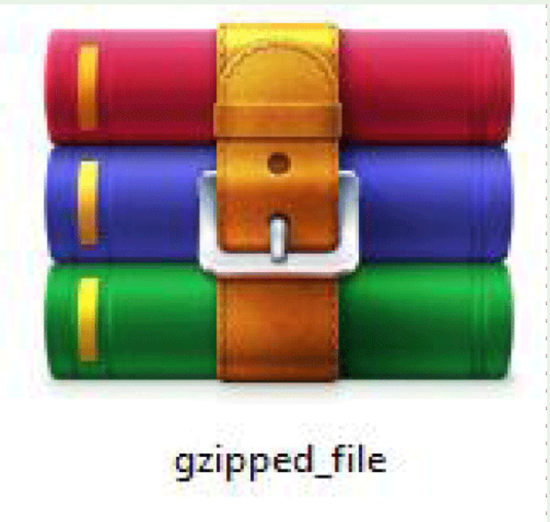


Figure 7: GZIP file compression of the encrypted compressed byte array

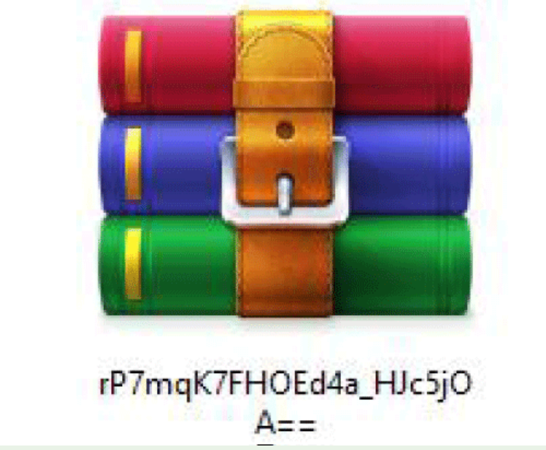
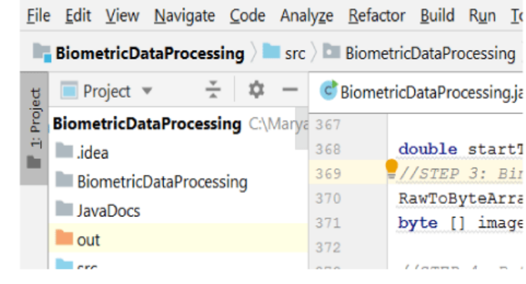
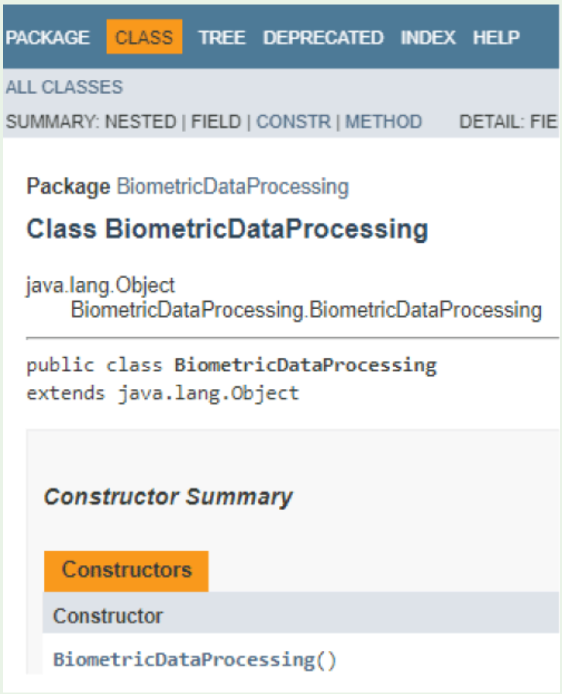


Figure 8: Encrypted filename of the biometric data


File Edit View Navigate Code Analyze Refactor Build Run To
BiometricDataProcessing
src
BiometricDataProcessing
1: Project
Project
BiometricDataProcessing.ja
BiometricDataProcessing C:\Marya
.idea
BiometricDataProcessing
JavaDocs
out
src

```
double startT
//STEP 3: Bir
RawToByteArra
byte [] image
```

Figure 9: Screenshot of simulation source code in Java using Jet Brains IntelliJ IDEA development environment


PACKAGE CLASS TREE DEPRECATED INDEX HELP
ALL CLASSES
SUMMARY: NESTED | FIELD | CONSTR | METHOD DETAIL: FIE

Package BiometricDataProcessing

Class BiometricDataProcessing

java.lang.Object
    BiometricDataProcessing.BiometricDataProcessing

```
public class BiometricDataProcessing
extends java.lang.Object
```

Constructor Summary

Constructors

Constructor
BiometricDataProcessing()

Figure 10: Screenshot of Java doc documentation of the software framework for the models.

Citation: Garba EJ, Odion PO, Evwiekpaefe AE and Ajakaiye F. Design and Analysis of Secure and Time-Space Efficient Models for Encoding of Four-Dimensional Multimedia Data in Biometric Security Systems. SM J Biometrics Biostat. 2019; 4(1): 1038.

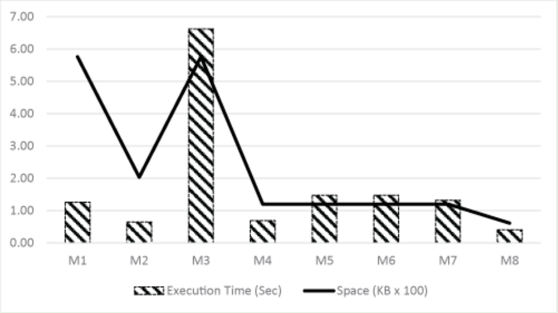


Figure 11: Graph showing the computational complexity of the models (in terms of Execution Time and Space)

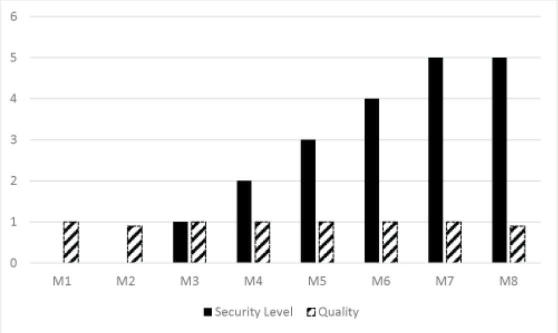


Figure 12: Graph showing relationship between Security Level and Quality of the models

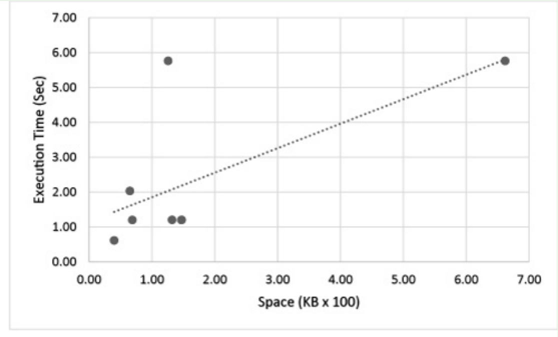


Figure 13: Graph showing relationship between Security Level and Quality of the models

image is expected to use 24-bit bit depth RGB color space of 300 dpi/pixels.

Bhavan and Marg [11] have also recommended uncompressed images for the enrolment process while the lossless JPEG 2000 color compression is also accepted with maximum compression ratio of 10. Accordingly, the face image standard accepts image specifications of 300dpi with bit-depth of 24-bit RGB color space. Specifications for that of the fingerprint image are 500dpi with bit depth of 8-bit using 200 grey levels.

The Indian e-Governance Standard has adopted the ISO-19794-4:2005(E) for fingerprint image data standard with the following of specifications: 500/100ppi with bit depth of 8-bit grey scale (200 grey levels) [12]. Face image data standard acceptable by the Indian government [13] is 300 ppi. According to the ISO 19794-6:2005(E) standard, only medium and higher quality images are acceptable for iris image data. The iris diameter is specified at 150 pixels with bit depth of 8-bit grey scale [14].

Security of Biometric Data: To shield biometric systems from attacks, a method of authentication that addresses privacy and template protection called Blind Authentication Protocol was developed [15]. Furthermore, other researchers also proposed the used encrypting algorithms such as RSA (Rivest-Shamir-Adleman) public-key cryptosystems for secure data transmission. The RSA encryption and digital signature algorithm is accepts key lengths up to 4096 bits [3,16]. The authors in [17] have developed an algorithm that demonstrates how the encryption over biometric data could be achieved at the binary level.

Security of sensitive information is of great concern during storage and transmission – especially, when such information are no longer under the control of the owner. Cryptography ensures secure communication between the different entities by transferring information in an unintelligible form. Afterwards it is expected that the authorized recipient is able to decode the information into a meaningful form [18].

However, the right choice of cryptographic algorithm is very imperative with regards performance, complexity, accuracy and efficiency [19]. In their experiment, Mushtaq et al.[18], Observed that the results of Blowfish, AES and Hi Sea provided more security. Blowfish was the best option when memory and encryption/decryption time are put into consideration. However, AES showed excellent performance generally. They concluded that the AES and Hi Sea could be employed in applications where data integrity and confidentiality is of highest priority.

Arunprakash et al.[20], developed a protocol for RSA encryption called BEBA (Biometric encoding and Biometric authentication). The AES algorithm was preferably adopted for template encryption during storage, key encryption and retrieval. In order to achieve both security and privacy of the biometric reference template, the template is converted into binary data for encryption, transmission and storage. Consequently, the stored data is retrieved and decrypted before identification and authentication processes take place. The protection of the biometric reference template is achievable using public key RSA algorithm and private key AES (Advanced Encryption Standard) algorithm [21].

Abood and Guirguis [22] studied and analyzed the most prevalent cryptography algorithms up to date. They discovered that all encryption methods had their advantages and setbacks and were suitable for appropriate applications. The result of the studies revealed that symmetric algorithms were faster than their asymmetric counterparts. Also, they have established that the most reliable algorithm is AES vis-à-vis speed of encryption and decoding, low computational complexity, the length of the key, structure and flexibility. The AES algorithm is considered to be suitable for both hardware and software encryption which supports block length of 128 bits and key lengths of 128, 192, and 256 bits.

The security of biometric data is achievable when the following are considered [23]:

a) The biometric data should be encrypted such that cracking it will require a reasonably long time (as this deters the cracker from making subsequent attempts).

b) In order not to degrade system performance, the encryption and decryption processes should be have low time complexity.

c) The encryption and decryption algorithms should not result in generating large encrypted/decrypted data – thus achieving low space complexity.

Therefore, the authors in [23] have proposed the use of RSA and ECC (Elliptic-Curve Cryptography) algorithm for biometric reference template protection. They used K-means algorithm to generate the key from the biometric reference template itself. The discovered that the RSA algorithm had faster time response, the ECC algorithm performed better under noise analysis (this means ECC algorithm is useful for remote authentication applications).

Just like other authors, Rashmi and Shohreh [24] have proposed the use of a chaotic encrypting algorithm before storing the biometric template into the database. During identification and authentication, the cipher text of the template is extracted from the database and decrypted for matching. They had a project that implemented fingerprint encryption using chaos algorithm. They stressed the fact that the security of the biometric data is very important otherwise it could lead to identity theft if such details get into the wrong hands. Hence, they opined that encrypting sensitive data such that an unauthorized person will not be able to tamper with it – as the encrypted data will not make any sense to him. When the authorized user needs it, it will be decrypted. The only drawback with their technique the irreversibility of the biometric quality. Note that quality plays a very critical role in feature extraction for the creation of the biometric template.

A study revealed that AES is the best performed algorithm than other encryption algorithms [25]. Blowfish and XOR (exclusive OR) encryption algorithms had average rates of overall performance. The Blowfish algorithm ranked second best in encrypting multimedia contents such as texts and images. On the other hand, the XOR algorithm showed good performance in encrypting audio and video data. Further results from the study showed that asymmetric algorithms (such as RSA) took the most time during encryption and decryption. However, symmetric encryption algorithms (such as AES) took less time than the asymmetric algorithms. In conclusion, the authors suggested the use of symmetric encryption algorithms for multimedia content encryption.

The protection of the biometric template is achievable by adopting strong encryption algorithms. Therefore, in order to achieve both security and privacy of the biometric template, the raw format has to be converted into binary data and then it has to be encrypted when storing and transmitting the template. However, during matching, the encrypted data has to be decrypted and the converted to the initial raw format [26, 27].

Space and Time Complexity: The study of computational complexity dates as far back as 1930s. One of those early researchers is Turing [28]. He discovered that the computability of functions (sequences) varied according their complexity. Computational complexity determines the resources required during the calculation to solve a problem [29]. Kesavan et al.[30], have opined that computational complexity requires persistent data structures.

Filmus et al.[31], have observed that there have been lots of research in complexity and time-space trade-offs. The complexity of a computational model depends on how much memory (storage space) and processing time is required in solving tasks. Low complexity usually indicates an efficient computational model. However, achieving efficiency vis-à-vis time and space requires a trade-off such that while enhancing space complexity the time complexity is not adversely compromised and the other way around. The complexity of computational models determines how efficiently an algorithm is executed within a given time and space. Computational complexity measures the required processing time and space in order to determine the efficiency of the algorithms [32]. A computational model represented by algorithm is described by a program as finite sequence of instructions used to solve a task [33].

From the reviewed works, it has been discovered that the common standards for working with multimedia based biometric data revolve around raw or compressed data formats (Table 2). The raw data formats are used because they contain unaltered data structure that guarantees best quality (highest data integrity); however, raw data formats are unwantedly characterized by large sizes. In order to reduce the size of such data formats, many biometric standards accept the altering of the internal data structure of the raw data format by using compression techniques. Notably, compressed data guarantee lower memory and processing time – hence resulting in lower computational complexity. Contrastingly, compression of the internal data structure of the raw format results in the degradation of the quality of such biometric data. Therefore, this research is proposing computational models for the encoding and decoding of four-dimensional multimedia data during biometric enrollment. The encoding process takes place at the hardware (sensor) level; while the decoding process is at the software (application) level. These models would consider algorithm steps, computational complexity (in terms of processing time and memory space), security level and the integrity/quality of the biometric data.

Methods and Materials

Literature review of related works forms the basis for the conceptualization of this research. The gaps discovered had prompted the need to design, develop and evaluate models for encoding and decoding of four-dimensional multimedia data during biometric enrollment. This aim is achieved via the following objectives:

a) Formulation of theoretical framework expressed as mathematical models and equations.

b) Design of the computational models for encoding and decoding biometric data using Edraw Max Pro.

c) Simulation of the models using Java programming language. The integrated development environment is Jet Brains IntelliJ IDEA.

d) Development of software framework for encoding and decoding biometric data – to be presented as Java objects which are documented using Java doc.

e) Evaluation of the performance of the computational models using a given standard fingerprint image. The results of the evaluation are presented in a tabular form containing the performance metrics namely algorithm steps, computational complexity (of execution time and space), security level and quality.

The applications for text editing is Microsoft Word; audio editing is Sony Sound Forge; image editing is Adobe Photoshop and video editing is Sony Vegas Pro. The application for empirical data analysis, visualization and info graphics is Microsoft Excel/Statistical Package for the Social Sciences.

Results

Theoretical Framework: The theoretical framework for this research is made up of five mathematical models. These mathematical models form the basis for formulating subsequent computational models for simulation. The mathematical models consist of functions for the abstraction of computational complexity, encoding/decoding of biometric data and measurement of performance of the models.

$$C = f\,(\text{et, sp}) \qquad (1)$$

Where: C is the computational complexity of the model.

et is the execution time (time required for processing to take place).

sp is the space required during processing.

$$M = f\,(E, D) \qquad (2)$$

Where: M is the model for encoding and decoding biometric data during the enrollment process.

$$r = \frac{n(\sum xy) - (\sum x)(\sum y)}{\sqrt{[n\sum x^2 - (\sum x)^2][n\sum y^2 - (\sum y)^2]}} \qquad (6)$$

E stands for the encoding process.

D stands for the decoding process.

$$E = f(\text{cp,rf,ba,dc,eb,ef,fc,oa}) \qquad (3)$$

Where: E stands for the encoding process.

cp represents the capturing of the biometric feature via the sensor.

rf represents the raw format creation process.

ba represents the binarization process of converting the raw format to byte array.

dc represents data structure compression of the raw format.

eb represents encryption of the byte array.

ef represents encryption of the filename.

fc represents file compression process.

oa represents output activity (writing the encoded biometric data to the storage).

$$D = f(\text{ia, fd, df, db, dd, rf}) \qquad (4)$$

Where: D stands for the decoding process.

ia represents input activity (reading the encoded biometric data from the storage).

fd represents file decompression.

df represents decryption of filename.

db represents decryption of byte array.

dd represents data structure decompression back to the raw format.

rf represents the raw format creation process..

$$P = f(\text{as, et, sp, sl, ql}) \qquad (5)$$

Where: P stands for the performance measurement function of the model.

as represents algorithm steps (as $\in \mathbf{Z}^{\neq}, \mathbf{Z}^{+}|$ as ≥ 1).

et represents the execution time (time required for processing). It is measured in milliseconds (ms).

sp represents the space required during processing. It is measured in kilobytes (KB).

sl represents the security level during the encoding and decoding processes. Note that sl $\in \mathbf{Z}^{+} \mid 0 \leq$ sl ≤ 5. See (Table 6) for the breakdown of the security levels.

ql represents the quality (integrity) of the biometric data before and after the encoding and decoding processes. Note that ql T$\mathbf{Q}^{+}| \; 0 \leq$ ql ≤ 1.

Simulation of the Models: Based on the theoretical framework, eight computational models for encoding and decoding biometric data were designed and simulated using Java programming language. The source code was developed in the Jet Brains IntelliJ IDEA development environment.

Model 1: This model consists of four algorithm steps:

Step1: Capturing and reading of raw data from the biometric sensor.

Step2: Creating raw format of the biometric data. (Figure 3) for sample of raw data format of fingerprint

Step3: Output activity (writing the raw data to the storage).

Step4: Input activity (reading the raw data from the storage).

In this model, the only step for the encoding process is the creation of the raw format (Step 2).

Model 2: This model consists of six algorithm steps:

Step1: Capturing and reading of raw data from the biometric sensor.

Step2: Creating raw format of the biometric data.

Step3: Data structure compression and conversion of the raw format.

Step4: Output activity (writing the compressed data to the storage).

Step5: Input activity (reading the compressed data from the storage).

Step6: Conversion of the compressed data format to the original raw data format.

The encoding process involves the creation of the raw data and its subsequent conversion to a data-structure compressed format (Steps 2 and 3). During the decoding process (Step 6), the compressed data is reconverted to its original raw form. Note that even though the raw format is recreated, the original quality can't be recovered because it has been lost during the data structure compression process.

Model 3: This model consists of six algorithm steps:

Step 1: Capturing and reading of raw data from the biometric sensor.

Step 2: Creating raw format of the biometric data.

Step 3: Binarization (Conversion of the raw data into binary format presented as byte array). (Figure 4)

Step 4: Output activity (writing the binary data to the storage).

Step 5: Input activity (reading the binary data from the storage).

Step 6: Conversion of the byte array to the original raw format.

The encoding process involves steps 2 and 3 – where the raw data is converted to its binary form (that is byte array). Step 6 indicates the decoding process of reconverting the byte array to its original raw format. This process retains the quality of the original raw data.

Model 4: This model consists of eight algorithm steps:

Step 1: Capturing and reading of raw data from the biometric sensor.

Step 2: Creating raw format of the biometric data.

Step 3: Binarization (Conversion of the raw data into binary format presented as byte array).

Step 4: Compression of the binary format – primarily to reduce the size of the byte array (see Figure 5).

Step 5: Output activity (writing the compressed binary data to the storage).

Step 6: Input activity (reading the compressed binary data from the storage).

Step 7: Decompression of byte array to its uncompressed format.

Step 8: Conversion of the byte array to the original raw format.

The encoding process (Steps 2-4) involves the conversion of the raw format into byte array which is subsequently compressed for reduced size. The decoding process involves steps 7 and 8. Both the decompression and reconversion of the byte array to the original raw format do not affect the original quality of the biometric data.

Model 5: This model consists of ten algorithm steps:

Step 1: Capturing and reading of raw data from the biometric sensor.

Step 2: Creating raw format of the biometric data.

Step 3: Binarization (Conversion of the raw data into binary format presented as byte array).

Step 4: Compression of the binary format – primarily to reduce the size of the byte array.

Step 5: Encryption of the compressed byte array (Figure 6).

Step 6: Output activity (writing the encrypted compressed binary data to the storage).

Step 7: Input activity (reading the encrypted compressed binary data from the storage).

Step 8: Decryption of the encrypted compressed byte array.

Step 9: Decompression of byte array to its uncompressed format.

Step 10: Conversion of the byte array to the original raw format.

The encoding of the biometric data starts from steps 2-5. The encryption of the compressed byte array provides additional level of security to the biometric data. The decoding process (steps 8-10) decrypts the compressed byte array and consequently reconverts the binary data into its original raw format.

Model 6: This model consists of twelve algorithm steps:

Step 1: Capturing and reading of raw data from the biometric sensor.

Step 2: Creating raw format of the biometric data.

Step 3: Binarization (Conversion of the raw data into binary format presented as byte array).

Step 4: Compression of the binary format – primarily to reduce the size of the byte array.

Step 5: Encryption of the compressed byte array.

Citation: Garba EJ, Odion PO, Evwiekpaefe AE and Ajakaiye F. Design and Analysis of Secure and Time-Space Efficient Models for Encoding of Four-Dimensional Multimedia Data in Biometric Security Systems. SM J Biometrics Biostat. 2019; 4(1): 1038.

Step 6: File compression (GZIP) of the encrypted compressed byte array (see Figure 7).

Step 7: Output activity (writing the encrypted compressed binary data to the storage).

Step 8: Input activity (reading the encrypted compressed binary data from the storage).

Step 9: Decompression (unzipping the file) of the encrypted compressed byte array.

Step 10: Decryption of the encrypted compressed byte array.

Step 11: Decompression of byte array to its uncompressed format.

Step 12: Conversion of the byte array to the original raw format.

Steps 2-6 encompass the encoding process for this model. Uniquely, step 6 provides further compression of the biometric data before formulating the output file. This step ensures more compression in order to reduce the size of the file. The decoding of the biometric data start from steps 9-12

Model 7: This model consists of fourteen algorithm steps:

Step 1: Capturing and reading of raw data from the biometric sensor.

Step 2: Creating raw format of the biometric data.

Step 3: Binarization (Conversion of the raw data into binary format presented as byte array).

Step 4: Compression of the binary format – primarily to reduce the size of the byte array.

Step 5: Encryption of the compressed byte array.

Step 6: Encryption of the filename of the biometric data (see Figure 8). The filename "**gzipped_file**" is now replaced with "**rP7mqK7FHOEd4a_HJc5jOA=**" after encrytption.

Step 7: File compression (GZIP) of the encrypted compressed byte array.

Step 8: Output activity (writing the encrypted compressed binary data to the storage).

Step 9: Input activity (reading the encrypted compressed binary data from the storage).

Step 10: Decryption of the filename of the biometric data.

Step 11: Decompression (unzipping the file) of the encrypted compressed byte array.

Step 12: Decryption of the encrypted compressed byte array.

Step 13: Decompression of byte array to its uncompressed format.

Step 14: Conversion of the byte array to the original raw format.

Model 7 is quite similar to Model 6. The only difference here is that the filename for the biometric file is encrypted before it is stored externally. The encoding process for this model encompasses steps 2-7. The decoding process starts from step 10-14.

Model 8: This model consists of fourteen algorithm steps:

Step 1: Capturing and reading of raw data from the biometric sensor.

Step 2: Creating data-structure compressed format of the biometric data.

Step 3: Binarization (Conversion of the raw data into binary format presented as byte array).

Step 4: Compression of the binary format – primarily to reduce the size of the byte array.

Step 5: Encryption of the compressed byte array.

Step 6: Encryption of the filename of the biometric data.

Step 7: File compression (GZIP) of the encrypted compressed byte array.

Step 8: Output activity (writing the encrypted compressed binary data to the storage).

Step 9: Input activity (reading the encrypted compressed binary data from the storage).

Step 10: Decryption of the filename of the biometric data.

Step 11: Decompression (unzipping the file) of the encrypted compressed byte array.

Step 12: Decryption of the encrypted compressed byte array.

Step 13: Decompression of byte array to its uncompressed format.

Step 14: Conversion of the byte array to the original raw format.

Models 7 and 8 are practically the same. Here, the data-structure compressed format of the biometric data is rather converted into the byte array instead of the raw data. The encoding process for this model encompasses steps 2-7. The decoding process starts from step 10-14. Worthy of note is the fact that data-structure compressed formats never guarantee 100% quality after reconversion to original raw data formats.

Documentation of Software Framework: The software framework presented as Java source code is standardly documented using the proprietary Java doc program. (Figure 10) shows a sample the homepage of the documentation.

Evaluation of the Computational Models: After simulation, the performances of the computational models were evaluated using a given standard fingerprint image. The results of the evaluation are presented in (Table 7). The table captured the performance metrics namely algorithm steps, computational complexity (of execution time and space), security level and quality.

Discussion

The result of the performance evaluation of the computational models (Table 7) shows that each model has its strength and weakness. The following sections shall concisely discuss some of the relationships between the computational models.

Computational Complexity: The computational complexity of the models is based on the relationship between the Execution Time (in Seconds) and Space (in Kilobytes). (Figure 11) shows that even though the space required for models 1 and 3 are the same, there is a sharp variation between them in terms of the execution time. This is owed to the fact that the raw data format was decoupled to its binary format which increased the entropy of the data. Compression is solely responsible for the drop in the execution time.

Data Integrity: (Figure 12) shows that it is possible to achieve highest level of security without distorting the quality of the biometric data .Model 7 stands out in this case. However, in situations where quality is not a prerequisite, then models 2 and 8 will suffice. Note that model 8 has the lowest computational complexity.

Correlation: Correlation coefficient (r) measures the strength and direction of a linear relationship between two variables. The value is always between +1 and –1. The interpretation of the values depicts uphill positive linear relationship (in case of positive correlation value), no correlation (when value is zero) and downhill negative linear relationship (in case of negative correlation value). Pearson's correlation coefficient formula is used here.

Where r is correlation coefficient; n is the number of sets of pairs of x and y values; x and y represent the columns of values.

From Table 7, the correlation coefficients for the following were calculated:

i) Execution Time and Space: r = 0.665738741 (strong uphill/ positive linear relationship). This supports time-space relativity in computational complexity.

j) Execution Time and Security Level: r = -0.266819004 (weak downhill/negative linear relationship).

k) Security Level and Quality: r = 0 (No linear relationship).

Conclusions

In all, eight computational models for encoding and decoding biometric data were designed and simulated. If low complexity is the target while aiming at maximum security of the biometric data, then model 8 is the best choice. However, this model has a quality value of 0.9. Also, increase in the security levels (vis-à-vis encryption) is responsible for the about 100% rise in the execution time (that is from 0.69ms to 1.47, 1.48 and 1.32ms). For average security level with relatively low execution time, model 4 is most preferable. If highest data integrity (quality) and best security is of concern (regardless of the computational complexity), the model 7 is candidate of choice. Finally, model 2 is very well suitable for most popular purposes including ISO (International Standards Organizations) standards. While these models are realizable using any programming language of choice, the developed source code in Java is useful as software framework to be used by biometric security systems designers and developers.

References

1. S. L. Nita, M. I. Mihailescu , V. C. Pau. Security and Cryptographic Challenges for AuthenticatioBased on Biometrics Data. Cryptography. 2018.
2. J. N. Pato, L. I. Millett. Biometric Recognition: Challenges and Opportunities. Whither Biometrics Committee, National Research Council. 2010.
3. M. Manoria, A. K. Shrivastava, S. S. Thakur, D. Sinha. Secure Biometric Cryptosystem for Distributed System. IJCNS. 2011.
4. S. Jaiswal, S. S. Bhadauria, R. S. Jadon. Biometric: Case Study. JGRCS. 2011.
5. H. Saini, K. Garg. Comparative Analysis of Various Biometric Techniques for Database Security. IJSR. 2013.
6. M. R. Loured, D. Khosla. Fingerprint Identification in Biometric Security System. IJCEE. 2010.
7. H. H. Kim, S. S. Park, W. Kim. A Framework for the Integration of Multimedia Data. JOT. 2005.
8. Y. Yin, L. Liu, X. Sun. SDUMLA-HMT: A Multimodal Biometric Database. CCBR. 2011.
9. Sun Z., Lai J., Chen X., Tan T. Biometric Recognition. CCBR 2011. Nigerian Biometric Standards Regulation. NIMC.
10. J. Sang, Z. Lei and S. Z. Li. Face Image Quality Evaluation for ISO/IEC Standards 19794-5 and 29794-5. Advances in Biometrics. ICB 2009.
11. Y. Bhavan, S. Marg. Biometrics Design Standards for UID Applications. UIDAI Committee on Biometrics. 2009.
12. Department of Information Technology. Fingerprint Image and Minutiae Data Standard for e-Governance Applications in India. Ministry of Communications and Information Technology. 2010.
13. Department of Information Technology. Face Image Data Standard for e-Governance Applications in India. Ministry of Communications and Information Technology. 2010.
14. M. Upmanyu, A. M. Namboodiri, K. Srinathan, C. V. Jawahar. Blind Authentication: A Secure Crypto-Biometric Verification Protocol. IEEE. 2010.
15. L. Szollosi, T. Marosits, G. Feher. Accelerating RSA Encryption Using Random Precalculations. IJNS. 2010.
16. S. L. Nita, M. I. Mihailescu , V. C. Pau. Security and Cryptographic Challenges for Authentication Based on Biometrics Data. Cryptography.2018.
17. M. F. Mushtaq, S. Jamel, A. H. Disina, Z. A. Pindar, N. S. A. Shakir, M. M. Deris. A Survey on the Cryptographic Encryption Algorithms. IJACSA .2017.
18. S. H. Jamel, M. M. Deris. Diffusive primitives in the design of modern cryptographic algorithms. ICCCE. 2008.
19. R. Arunprakash, T. Jayasankar , K. Vinothkumar. Biometric Encoding and Biometric Authentication (BEBA): Protocol for Secure Cloud in M-Commerce Environment. Appl.Math.Inf.Sci. 2018.
20. S. Kavinhariharasudhan, S.Saravanakumar. A Review on Approaches to Shield against Ddos Attack in Cloud Computing. International Journal of Emerging Technology and Research. 2014.
21. O. G. Abood , S. K. Guirguis. A Survey on Cryptography Algorithms. IJSRP. 2018.
22. M. ManiRoja, S. Sawarkar. Biometric Database Protection using Public Key Cryptography. IJCSNS. 2013.
23. J. C. Rashmi, K. Shohreh. Biometric Encryption. IARJSET. 2017.
24. Md. M. Ahamad, Md. I. Abdullah. Comparison of Encryption Algorithms for Multimedia. RUJSE. 2016.

25. T. Ristenpart, E. Tromer, H. Shacham, S. Savage. Hey, You, Get off of My Cloud: Exploring Information Leakage in Third-Party Compute Clouds. ACMDL. 2009.

26. B. Lampson, M. Abadi, M. Burrows, E. Wobber. Authentication in Distributed Systems: Theory and Practice. ACMDL.1991.

27. Department of Information Technology. Iris Image Data Standard for e-Governance Applications in India. Ministry of Communications and Information Technology. 2011.

28. M. Turing. On computable numbers, with applications to the Entscheidungs problem. PLMS. 1937.

29. R. Kesavan, R. Singh, T. Grusecki , Y. Patel. Algorithms and Data Structures for Efficient Free Space Reclamation in WAFL. ACMDL. 2017.

30. Y. Filmus, M. Lauria, J. Nordstrom, N. Thapen, N. Ron-zewi. (CCC) IEEE. 2012.

31. J. A. Ruiz-Vanoye, O. Díaz-Parra. An Overview of the Theory of Instances Computational Complexity. Redaly. 2011.

32. V. D. Blondel , J. N. Tsitsiklis. A survey of Computational Complexity Results in Systems and Control. IFAC .2000.

33. J. Hartmanis , J.E. Hopcroft. An Overview of the Theory of Computational Complexity. JACM. 1971.