DOI: 10.24996/ijs.2023.64.1.40

ISSN: 0067-2904

Multibiometric System with Runs Bits Permutation for Creating Cryptographic key Generation Technique

Raghad Abdulaali Azeez[1*], Abeer Salim Jamil[2], Ayad Al-Adhami[3], Nidaa Flaih Hassan[4]

[1]*Computer Unit, Collage of Education for Human Science-ibn rushed, University of Baghdad, Iraq*
[2]*Department of Computer Technology Engineering, Al-Mansour University College, Iraq.*
[3]*Department of Computer Science, University of Technology, Baghdad, Iraq.*
[4]*Department of Computer Science, University of Technology, Baghdad, Iraq.*

Received: 24/12/2021 Accepted: 24/6/2022 Published: 30/1/2023

Abstract

The context of multibiometric plays a pivotal role in enhancing an identification system, since a biometric system is now the most physical way of identifying and verifying individuals. The feature of multibiometric could be merged to produce identification information. However, unimodal biometric systems suffer from different types of breaching. Thus, mixing biometrics with cryptography leads to overcome small variations existing between diverse acquisitions of the same biometric in order to produce the robust system. In this paper, a new robust multibiometric system is proposed to create a random key from person multibiometric, facial and fingerprint images which are used simultaneously to produce this key. Several manipulations are made on compactness information for these two images to get a unique key for each person. The generated random key can be used for electronic numbers, passport identification, civil identification card, and it could be used as seeds for pseudo-random number generators. The multi-biometric system operates on two images, faces and fingerprints, by partitioning each image into four parts and taking the highest density for each one, XOR these parts; diffusions process is applied on these parts including permutation and thresholding to produce a random key. The generated key cannot be revocable that passed through randomness tests to ensure whether the generated key is accepted as true. Thus, the results of the tests are passed and presented that all generated keys are accepted to be random and unpredictable binary sequences and hence they can be used efficiently.

Keywords: Key Generation, Multibiometric , Fingerprint, Face, Permutation, Randomness test

القياسات البايولوجية المتعددة مع تناوب تشغيل البتات لانشاء تقنية توليد مفاتيح التشفير

رغد عبد العالي عزيز[1] * ، عبير سالم جميل[2] ، اياد الاعظمي[3] ،نداء فليح حسن[4]

[1]*وحدة الحاسوب ، كلية التربية للعلوم الانسانية – ابن رشد ، جامعة بغداد ، العراق.
[2]قسم هندسة تقنيات الحاسوب ، كلية المنصور الجامعة ، العراق.
[3] قسم علوم الحاسب الآلي ، الجامعة التكنولوجية ، بغداد ، العراق.
[4] قسم علوم الحاسوب ، الجامعة التكنولوجية ، بغداد ، العراق.

*Email: abeer.salim@muc.edu.iq

الخلاصة

يلعب سياق القياسات الحيوية المتعددة دورًا محوريًا في تعزيز نظام تحديد الهوية ، لأن نظام القياسات الحيوية هو الطريقة المادية للتعرف والتحقق من هوية الأشخاص. يمكن دمج قياسات حيوية متعددة للحصول على معلومات تحدد هويته. ومع ذلك ، فإن أنظمة القياسات الحيوية أحادية الوسائط تعاني من أنواع مختلفة من الاختراق. وبالتالي ، فإن مزج القياسات الحيوية مع التشفير يؤدي إلى التغلب على الاختلافات الصغيرة الموجودة بين عمليات الاكتساب المتنوعة لنفس القياسات الحيوية من أجل إنتاج نظام قوي . في هذا البحث ، تم اقتراح نظام جديد قوي متعدد المقاييس لإنشاء مفتاح عشوائي لشخص بالاعتماد على المقاييس الحيوية المتعددة هي الوجه وبصمة الاصبع، والتي يتم استخدامها في نفس الوقت لإنتاج هذا المفتاح. يتم إجراء العديد من المعالجات على معلومات الضغط لهاتين الصورتين للحصول على مفتاح فريد لكل شخص. يمكن استعمال المفتاح العشوائي الذي تم إنشاؤه للأرقام الإلكترونية ، وتحديد جواز السفر ، وبطاقة الهوية المدنية ، ويمكن استعماله كبذور لمولدات الأرقام العشوائية الزائفة. يعمل النظام متعدد المقاييس الحيوية على صورتين هي الوجه و بصمة الأصبع ، عن طريق تقسيم كل صورة إلى أربعة أجزاء وأخذ أعلى كثافة لكل جزء، XOR هذه الأجزاء ؛ يتم تطبيق عملية الانتشار على هذه الأجزاء بما في ذلك التقليب والعتبة لإنتاج مفتاح عشوائي. لا يمكن إلغاء المفتاح الذي تم إنشاؤه والذي اجتاز اختبارات العشوائية للتأكد من قبول المفتاح الذي تم إنشاؤه على أنه صحيح. وبالتالي ، يتم تمرير نتائج الاختبارات وتقديم أن جميع المفاتيح التي تم إنشاؤها يتم قبولها لتكون تسلسلات ثنائية عشوائية وغير متوقعة وبالتالي يمكن استعمالها بكفاءة.

1. INTRODUCTION

Most applications of biometric systems in the real world are unimodal which is culminated in various security problems, such as non-universality, tricked attacks, less accurate results, and the sensed data suffering from noise [1]. Therefore, most strategies to overcome the security problems are to improve the performance of identification and authentication by using unimodal biometrics in a multibiometric system [2].

Multibiometric systems can accelerate the filtering or indexing of large-scale biometric databases. For example, in a bimodal system that consists of fingerprint and face, the fingerprint feature is used to extract a filtered list of potential identities from a large database of subjects, and face model determines the final identity from this limited filtered list[3]. The multibiometric systems are also degraded effectively due to the problem of noisy data. When the biometric signal is gained from a single feature, it is destroyed by noise; the safety of other features may assist in the efficient determination of identity. Some systems take into account the quality of the individual biometric signals during the authentication process, this is especially important when recognition takes place where certain biometric features cannot be reliably extracted. For example, the wearing of lenses in the eyes may lead to wrong computations in extracting eyes features, when an individual's iris characteristics cannot be correctly measured, the characteristics of the face may be used by the multi-biometric system to perform authentication[3].

A challenging problem is in quality estimation, therefore a multi-biometric system is represented as a fault tolerance system, thus it can operate on a large-scale authentication systems environment even when the software or sensor malfunction has happened, or intentional user impact, the system operates continually [3]. To make a strong Unique Identification Number (UIN), a system can be implemented to capture and store multiple biometrics, such as iris, fingerprints, and face from a person by making a template which is the final idea of the overall person identity and picking up this template may lead to an identity loss. The template is generated from a digital model of unique characteristics that have been concluded from a biometric sample of a person. Biometric templates are known to be the identity of an individual. These templates are processed during the biometric identification/authentication process. The matching unit compares the feature set that is picked up through authentication with the enrolled templates and get the results (match / unmatched). The decision unit processes these identical results to either confirm or ensure the identity of an individual. The biometrics characteristics are transformed into templates by using a different kinds of algorithms [4, 5, 6, 7].

The biometric approach for authentication is interesting because of its accessibility and possibility to offer security by increasing security lengths. For example, using a strong cryptographic approach such as AES256 cryptographic protocol ensures that sensitive private data are secured. Meanwhile, generating the secret key based on an individual's biometric signatures such as fingerprints and face is more secured if it is used as encryption/decryption key pairs and use the key to encrypt the sensitive personal information [8, 9, 10]. Many associated resource-constrained devices aren't equipped to perform costly traditional cryptographic computations, making it challenging to implement appropriate cryptographic functionality. Lightweight symmetric cryptography has recently been developed, allowing for a faster cryptographic process with fewer computer processes [11, 12].

In this paper, a multi-biometrics system based on mixing parts from facial and fingerprint images is used to extract the random key specified for each person. This key is confidentially secured for any identification application that is related to a person.

The rest of this paper is as follows: section 2 describes different key generation techniques from different literature sources. Section 3 represents various techniques that have been used in image segmentation. Section 4 and 5 discuss the concept of fingerprint images and multibiometric respectively. Section 6 introduces the proposed system that uses multi-biometric techniques. The experimental results of the proposed system can be seen in section 7. Section 8 presents the conclusions of the work.

2. RELATED WORKS

The cornerstone of the secret world is the secret key. For acquiring such a strong key, it's derived from biometric measurements, and these measurements are found in different characteristics in our bodies. Jagadeesan *et al.* in [8] constructed a multi-biometric template from iris and fingerprint. An image of a fingerprint is Region of Interest (ROI) and dividing to 16*16 block, while the iris is segmented and Hough transform is used to detect the edge and generated **256-bit** key is improved by the associated complexity of factoring the large number. Chandra S., et al in [13] generated a key that is extracted from the fingerprint image after preprocessing it according to ridges and furrows that exist in the image. The converting the image is converted to binary matrix according to the threshold (the mean intensity value of the block) after dividing the pixels of the matrix to block 16*16, and then removed the noise from the image by Region of Interest (ROI) using morphological operations. Abuguba S., et al in [14] generated key from multibiometric face and iris characteristics that are fused using Principle Component Analysis (PCA) and Gabor filter to construct the template containing 256 bits used to generate a strong key. The process of key generation are dependent on modulus 2 to the sum of one's and zeros for both face and iris binary images in an orderly procedure. Maček N. and et al in [15] presented an approach of building a table that contains two values for the same person; the first value is obtained from the hash calculation of an iris, and the second value is a cancelable template generated from a fingerprint. These two values are put in lookup table when the key is generated, hence, the key is released if two conditions are achieved, the hash of the key is found in the lookup table and there is a fault-tolerance rate matching between the corresponding templates (stored template and the generated template). Balamurugan, G. and et al in [16] presented a method for generating a key from multibiometric features by using face and iris images, their method consists of three steps. In the first step, features are extracted from the face imag**e** by using Local Mapping Binary Pattern (LMBP) due to the property of tolerance against illumination changes (histogram of the face image), and from iris images by using Gabor wavelets. In the second step, texture properties act as feature extraction, these features are fused to create a template by shuffling the feature vectors, combining and merging them. 256-bit key length, is generated from the template, in step three. Kanade S. et al in [17] proposed a key generation schema from information obtained from multi-biometric system based on uncorrelated left and right irises of a person. The iris image is decomposed using Gabor filters and quantitative stage information to build iris code with weighted error correction data. The codes of these irises are combined to compose the iris feature vector. The template is built from a fused iris feature vector with the key that is shuffled depending on the password. Fusing this information produced 147-bit key generation. With the enrolled templates and generated match scores, the decision module processes these match scores to determine or verify the identity of an individual. The biometrics traits are transformed into templates by using mixed and different algorithms.

Sun S W. et al in [18], described Key Mixed Template (K-M-T) method, their idea is to mix secret keys with the person's template, the function of blending C is to mix a vector of keys (K) to the person's template (P), as C(Pi; Ki) = Pi + Ki. The user has a secret key (K) in each database, When the attacker succeeds in accessing DB1, he cannot access DB2, as each database has a different secret key, so this method is safe for the user who has two different access to different databases, since (KMT) 1 ≠ (KMT) 2. The K-M-T method is devised to treat with available biometric systems to improve the Secrecy of template protection, thus it protects the user from any fraud attempt.

As seen in section 3 the proposed multibiometric system relies on two techniques which are image segmentation and fingerprint image techniques, tables and Figures are presented center, as shown in Table 1 and Figure 1, and cited in the manuscript before appearing.

3. MULTIBIOMETRIC

Personal identification or verification methods such as passwords, ID cards, etc. have proven ineffective and unable to satisfy the demands of modern society since they may be stolen or missing **[19, 20].**

For these reasons the biometrics identification systems becoming the focus of the research in recent years. These systems are mainly focused on using fingerprint images and image segmentation techniques.

3.1. FINGERPRINT IMAGES

A fingerprint image is typically represented as an unordered set of minutiae, which encodes the location {x, y} and orientation {è} of friction ridge discontinuities.

The popularity of smart cards as a storage media for the template, created for biometric purposes is increased, but there are some problems associated with mistakes that originate with using templates, such as integrity and security from the confrontation [21]. The fingerprint image is commonly used in the recognition field, it is deployed extensively and studied deeply. Fingerprint contains ridges and valleys, **and** their location has been determined since the birth of humans. The identical twins also have a dissimilar pattern of a fingerprint. The following are some points that distinguish fingerprint from the rest of the biometrics (for example iris, voice, and face):

- Fingerprint reader machines are deployed at an increasing pace currently, such as mobile devices, door locks, automotive, and surveillance systems.
- A new Master-Card which contains nestled fingerprint reader tries to insert a biometric authentication layer for card payment.
- Fingerprint biometric devices are deployed over a large market share and have been integrated into various applications [22].

3.2. IMAGE SEGMENTATION TECHNIQUES

Image segmentation divides the image into several parts or segments having similar features or attributes to get image characteristics, the segmentation is the way to analyze these characteristics to be meaningful, analyzable, and easy to extract features [23, 24].

There are several existing techniques used for image segmentation, and they are threshold method, edge based method, clustering based method, ANN Based method, PDE based method, region-based method, and watershed based method. The threshold method is an effective way to segment an image. This method is based on statistical analysis and it depends on peak spatial details which are found in the face **[10, 25, 26]~~.~~**

A digital image is classified according to the Intensity Level (INT), this operation converts the image into (X×Y) pixel**s**, each pixel represents an INT value that belongs to the level which is classified if the image is a grey level image. The method of threshold segments the area into two classes, A1 and A2, if the pixel has an INT greater than a threshold, it can be classified as the first class (A1), else it is classified as the second class (A2). The following equation describes this process:

$$A_1 = R \; if \; (0 < R < T) \tag{1}$$

$$A_2 = R \; if \; (T < R < INT - 1) \tag{2}$$

Where R is the value of the pixel, and T is the thresholding value. For multilayer thresholding, the image is divided into more than two classes, to find the threshold values which are determined according to the mean intensity of the image to divide pixels into various groups:

$$\mu = x / n \tag{3}$$

Where µ is the mean intensity value of x [27]. Standard Deviation, Decimal Scaling, and Min-Max methods are used in normalization when the data is preprocessed, all the above methods treat with **a** data range (0-1), and each method handles the data differently. For example, Decimal scaling method, the range is tight in subinterval; to solve this problem, Standard Deviation or Min-Max normalization can be used [28].

In this paper, the standard deviation (*SD*) is used as described in the following equation[27]:

$$SD = \sqrt{\frac{\sum_{i=1}^{n} (\bar{x} - x_i)^2}{n-1}} \tag{4}$$

Where $\bar{x}$ is the mean sample, x_1, x_2, x_3… x_n are the data samples, and n= is the sample size

4. THE PROPOSED MULTIBIOMETRIC SYSTEM

The proposed multibiometric system relies on associating cryptography techniques with biometric to achieve singularity and randomness and to accomplish the robustness **of** the proposed system.

In the proposed system, two multibiometric images (i.e., facial ~~face~~ and fingerprint images) are mixed simultaneously to generate **a** random key , key is characterized by its robustness and singularity. These two biometric images are passed through many processes to generate a unique key. There are three reasons to select these images which are:

1) Human face recognition plays an important role in many user authentication applications in the modern world, it is a convenient, non-intrusive authentication method

2) Fingerprints have used for more than a century since it denotes the most broadly used technology for biometric identification, and the fact that fingerprints are formed in the final stage and remain structurally unchanged throughout an individual's lifetime.

3) Most techniques can only generate short keys that are of insufficient length~~.~~ Using biometric images overcomes on this limitation, since we could deal with huge data as inherent with images.

As mentioned previously, the basis of this proposed key generation depends on mixing two types of images, the first type is facial images that are taken from Essex Faces 95 database, some of its samples are shown in Figure 1, and the second type is fingerprint images that are derived from SDUMLA-HMT Database~~.~~ [29], some of its samples are shown in Figure 2. Each biometric process is exploited statically variance from two biometric images for the same person.

Figure 1: Three samples from Essex Faces 95 Dataset.

Figure 2: Three samples from SDUMLA-HMT Database.

The general architecture of the proposed system is shown in Figure 3. After receiving the faces images, the system starts with resizing facial images by dividing them into four parts and then compute the standard deviation for each part. Consequently, the maximum standard deviation is detected to convert parts to one-dimensional vector. The same procedure is used for fingerprint images. Overall, 2-dimensional vectors are being selected from the facial and fingerprint images to be XORed as binary vectors. The number of cells with the number of zero and one calculation is applied to a threshold function to get a master key generation.

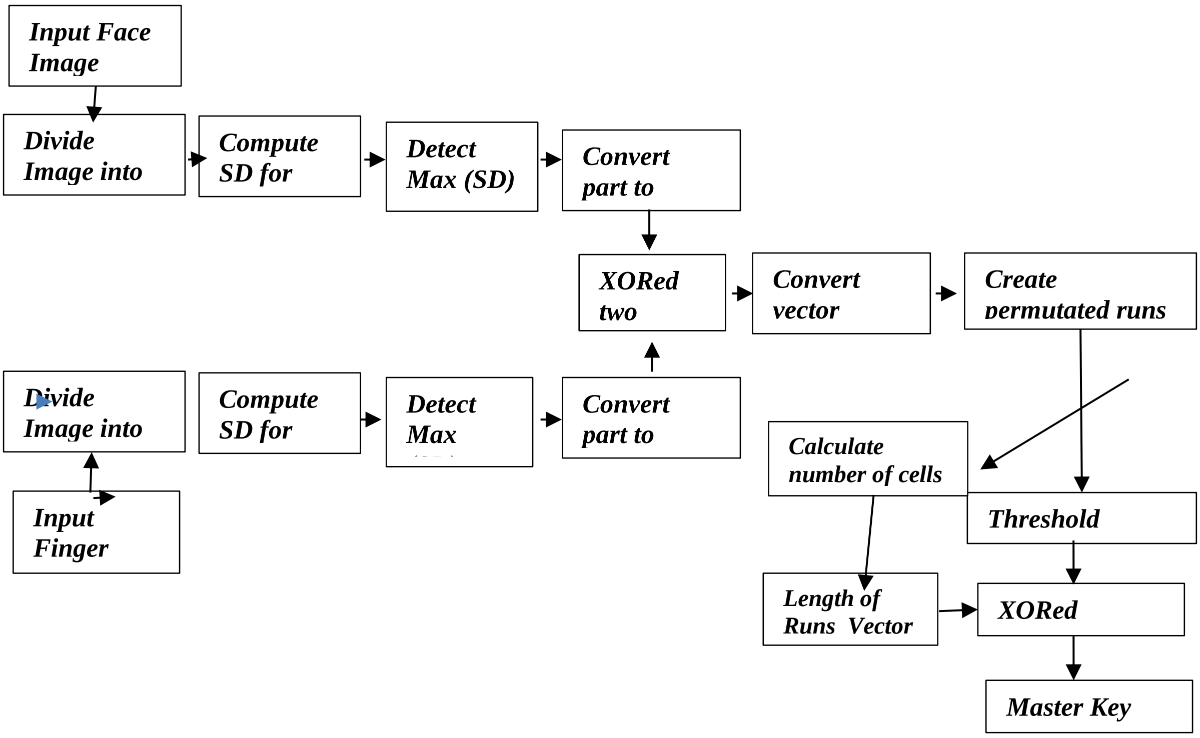

Figure 3: General architecture of the proposed system.

The whole procedure of the Multibiometric system uses different phases to generate multibiometric keys which are Biometric image partition, High-density block detection, Mixed high-density parts, Binary conversion parts, Permutation process, Apply threshold function, and key generation phase.

*4.1 **BIOMETRIC IMAGE PARTITION***

This phase is applied to manage image content easier by partitioning the image contents since it's easier to process a small part of the image rather than the entire image, thus a quadratic division is applied to obtain four parts from a facial image and four parts from the fingerprint image**. E**ach part can be handled alone for further processing~~.~~

4.2 *HIGH-DENSITY BLOCKS DETECTION*

As the quadratic division is applied to each biometric image from the previous phase to obtain four parts, the mean value of the image provides the contribution of a separate pixel intensity for the entire image. The deviation value estimates diversity or variability used in statistics, and in terms of image processing, it demonstrates how much variation or dispersion exists from the mean. A low standard deviation indicates that the data points tend to be very close to the mean, whereas a high standard deviation indicates that the data points are spread out over a large range of values. Image part with highest standard deviation means the high density is needed to be as a seed~~s~~ value for generating a random key. Mean and standard deviation measures are applied to each part, then the high-density part is selected to be processed. Algorithm 1 illustrates the steps of the first and second phases ~~:~~

Algorithm 1 High-density blocks detection

```
Input: Face Image(Face_Img), Fingerprint(Fing_Img )
Output: Max_SD_Face      // The block with the highest SD of face image
        Max_SD_Fing      // The block with the highest SD of finger image
Start
 1: Face_part_Len = Length (Face_Img ) / 4
 2: For all X, Y Do in Face_part do the following :
        Compute Mean for each part using Eq. (3).
        Compute SD for each part using Eq. (4).
 3 : Find the maximum value of SD ( Max_SD_Face)
 4: Finger_part_Len = Length (Fing_Img ) / 4
 5: For all X,Y Do in Finger_part do the following:
        Compute Mean for each part using Eq.(3).
        Compute SD for each part using Eq. (4).
5      : Find the maximum value of SD ( Max_SD_Fing)
End
```

Thus, two high-density parts are detected, one from the facial image, and the second from the fingerprint image. Table 1 shows six samples of the facial and fingerprint images with their sizes in bytes, their parts sizes in bytes, and their standard deviation values.

The output from this phase detects two high-density parts (i.e., the maximum value of SD) from two biometric images. There is one important issue that must be considered, the detected parts must have the same size to be processed for key generation, if they differ, part with small size is considered, although the data is lost from the other part.

Table 1: Shows standard deviation (SD) for each part of facial and fingerprint images

Sample Name	Image Size in Byte	Part Size in Byte	Part_1	Part_2	Part_3	Part_4
			SD	SD	SD	SD
Face_Img_1	115302	28825	46.992	43.263	46.145	38.370
Finger_Img_1	240578	60144	107.635	112.832	105.517	98.330
Face_Img_2	115302	28825	58.436	54.863	61.076	46.153
Finger_Img_2	240578	60144	103.828	105.726	101.617	95.923
Face_Img_3	115302	28825	45.953	47.511	58.472	18.386
Finger_Img_3	240578	60144	110.430	104.528	105.022	96.428

4.3 MIXED HIGH-DENSITY BLOCKS DETECTION

After determining the highest SD value for each image in the previous phase, the two highest SD values for each image are mixed using the XOR operation in this phase. This operation results in diffusion in the two selected parts, i.e., redundancy property of two images values are dissipated. It's important to consider parts sizes as mentioned before, if the two parts are equal in size, the XOR operation proceeds for whole bytes in parts, but if differ, XOR operation is limited by the small part size, the output produced by XORed operation is called Initial_Vector.

4.4 BINARY CONVERSION OF MIXED HIGH-DENSITY BLOCKS

5 The necessity for random binary numbers increased as recent communication structures progressively employed electronic connections and digital signature requests for

validity, thus it has become essential for safe secrecy through these actions. This is the cause for developing true random binary number generation, which should designate high randomness for usage in authentication for digital communications. A binary string achieves good results in the formation of random string if no observable relationship exists between the individual bits of the sequence, thus the established XORed block from the previous phase (Initial_Vector) is converted to binary vector, each byte is converted to binary form to produce a sequence of unsystematic bits (Binary_Vector). Algorithm 2 illustrates the steps of the third and fourth phases**.**

Algorithm 2 Creation initial vector and binary vector

```
Input: Max_SD_Face , Max_SD_Fing
Output: Face_Vector     //  Face vector with maximum SD
Finger_Vector  // Finger vector with maximum SD
Initial _Vector     //  Binary vector
Start
1: Len_1= length of  Max_SD_Face
2: Len_2= length of  Max_SD_Fing
3: Min_Len= Minimum of ( Len 1, len_2)  // Find  minimum length of two parts
4 : For I = 1 to Min_Len do the following
Initial_Vector ( I) = Face_Vector  XOR Finger_Vector
End For
5: Binary_Vector = Convert ( Initial_Vector)   //  Convert
byte in  Initial_Vector into Binary form
End
```

4.3 CREATING PERMUTATED RUNS BITS VECTOR

A sequence of bits represented in Binary_Vector is manipulated in this phase to convert unsystematically obtained bits to be permutated and chaotically diffused. In this phase, a new configuration from a binary vector is achieved by writing each distinct binary bit followed by its runs (occurrences), the following equation is used to obtain the runs vector.

$$\text{₱} = \{clear\ Y, IF\ Binary - Vector\ (r) \neq\ Binary -\ Vector\ (r+1\}$$

and

$$\text{₱} = \{increment\ Y\ by\ 1,\ \ if\ Binary - Vector\ (r)\ =\ Binary - Vector\ (r+1)\} \qquad (5)$$

Where ₱ is the permutation process, **r** represents the position value in Binary-Vector array, the result of permutation process is Runs-Vector array and its values are integers explained the runs that occurred for ones and zeroes. Figure 4 shows a simple sample of creating Runs_Vector, the first row contains a binary sequence obtained from merging two high-density parts, and the second row contains each cell value followed by its runs (Runs_Vector). Algorithm 3 illustrates this process:

Binary Vector	**0**	**1**	**1**	**0**	**1**	**1**	**1**	**0**	**0**		**……**
Runs Vector	0	1	1	2	0	1	1	3	0	2	....

Figure 4: A sample of creating runs vector

Algorithm 3 Creating permutated runs bits vector

```
Input:  Binary_Vector
        Bin_Len                  // Length of Binary_Vector
Output: Runs_Vector .
Start
1: I=0  : J=0                    // Initialize indices
2: Bit_Occur = 1                 // counter for 0s and 1s runs
3: Runs_Vector (J) =  Binary_Vector (I)
4: While EOF ( Bin_Len)
     While  Binary_Vector (I) = Binary_Vector (I+1) And   I < Bin_Len
       Increment Bit_Occur by 1
        I = I+ 1         // Increment index to read next number
       End While
        Runs_Vector (J+1)= Bit_Occur
        J=J+1
       Runs _Vector (J) =  Binary_Vector (I)
       Bit_Occur = 1                 // Initialize counter of runs
        I = I+ 1
     End While
5 : Number of Runs_Cell  = Length (Runs_Vector)
                                 // number of run cell
6 : Bin_Run_Vector = Convert (Number of Runs_Cell  )  //  Convert number of
run cell into binary  form
End
```

4.6 APPLY THRESHOLD FUNCTION

The content of Runs_Vector produced from the previous phase results in a binary number (zero or one) with numbers representing runs value. Another step to increase unpredictability and randomness is applied by using the threshold function, this function makes the runs vector containing only zeroes and ones, the following equation describes the threshold function:

$$T(X) = \begin{cases} 0 & I(X) < \mathrm{TH} \\ 1 & I(X) \geq TH\} \end{cases} \qquad (6)$$

Where T(X) represents value in runs vector, Threshold function represents threshold value which is equal to 1, the output of this equation is either 0 or 1, and the output vector is called Threshold_Vector. Figure 5 explains this process when threshold =1:

Runs Vector	**0**	**1**	**1**	**2**	**0**	**1**	**1**	**3**	**0**	**2**
Zero_One Vector	0	1	1	1	0	1	1	1	0	1

Figure 5: Shows creating the Threshold_Vector

4.7 KEY GENERATION

The strength of the random number generator used by the security systems often determines how safe the systems are, this strongly depends on the actual randomness of the generated bits. Randomness is increased by applying operations that increase separation, unsystematically, chaotically, unguessed, permutated, and diffused. In this phase, a new master vector (key) is generated by mixing two vectors, the first one is the Binary Vector, which is obtained from creating a master vector. Algorithm 4 illustrates steps of applying threshold function and master vector generation. In addition, Table 2 shows the sizes of selected parts with the highest SD, binary vectors and runs vectors and size value binary of runs cells in bit run bits vector step four then its value will be XORed with Threshold_Vector. This operation can increase randomness, unpredictability and arbitrariness.

Algorithm 4: Key Generation

```
Input:  Runs_Vector
        Runs_Len                  // Length of Runs_Vector
        TH                        // Threshold value
        Key_Size
Output: Master _Vector
Start
1: For  I =  0  to  Runs_Len
       If    Runs_Vector(I) < TH  then
                  Threshold _Vector(I)= 0
           Else
                  Threshold _Vector(I)=1
       End If
    End For
2 : Num_Cell = Length (Threshold _Vector)
3:  For I=  0  to  Key_Size      // Mixing  Threshold  and  Binary vectors
          Master _Vector(I) = Threshold _Vector(I) XOR  Binary_Vector (I)
    End For
4 : For I= 0 To Key_Size     // shuffle Master _Vector
        Sh_Rnd = Int(Rnd * (Unbound (Master _Vector)-  Bound(Master _Vector) + 1) + Bound(Master  _Vector))
        Temp = Master _Vector (i)
         Master _Vector (i) = Master _Vector (Sh_Rnd)
         Master _Vector (Sh_Rnd) = Temp
     End For
End
```

Table 2: Selected parts with high SD, binary vectors , runs vectors and size value binary of runs cells in bit

Sample Name	Size of Selected part with high SD	Size of Binary vector	Size of Runs vector
Face_Img_1 **Finger_Img_1**	28825	230600	215404
Face_Img_2 **Finger_Img_2**	28825	230600	224954
Face_Img_3 **Finger_Img_3**	28825	230600	230596

From Table 2, it is observed that initial sizes of the binary runs vectors are different from person to person. This difference in vector sizes supports strength in the key generation process since the size of the vector is not easy to detect through key generation phases.

As mentioned previously, the basis of this proposed key generation depends on mixing two types of images, the first is the facial image that is derived from Essex Faces 95 database and some of its samples are shown in Figure 1, and the second is fingerprint images, derived from SDUMLA-HMT Database [29], some of its samples are shown in Figure 2.

5. EXPERIMENTAL RESULTS

The proposed key generator produces random secured keys. thus, they should be tested by specific tests to check whether they are like the truly random sequence or not [30]. in this experiment, 500 samples of facial images obtained from 50 subjects, and 500 samples of fingerprint images obtained from 50 subjects are tested. each facial image is normalized into an 8-bit 196*196-pixel image, and each fingerprint image is normalized into 383*209-pixel image. in the experimental examination, a range of {1-5} thresholds are verified using False Acceptance Rate (FAR) and False Rejection Rate (FRR) metrics. Since the key length is started with 64 bits.

TABLE 3: FAR and FRR for different key lengths and different threshold values

Threshold	Key Length	FAR %	FRR%
1	64	5.3%	22.4 %
	128	3.7%	28.4%
	256	2.4%	32.3%
	512	1.3%	36.6%
2	64	5.4%	24.4%
	128	4.3%	32.3%
	256	3.1%	37.4%
	512	3.1%	40.1%
3	64	4.6%	20.9%
	128	4.0%	15.4%
	256	3.2%	12.7%
	512	2.9%	12.1%
4	64	0.9%	10.8%
	128	0.7%	10.4%
	256	0.0%	9.7%
	512	0.0%	9.0%
5	64	0.9%	10.8%
	128	0.7%	10.4%
	256	0.0%	9.7%
	512	0.0%	9.0%

From Table 3, the results show that when the threshold is equal to 5 with key's length equal to 512 bits, then FAR value is exceeded to be 0% and FRR value is exceeded to be 9.0%. Therefore, the key is suitable to be used for a critical application that needs a randomness and unpredictability key~~;~~ since because any fraud key cannot enter the system due to the FAR value being 0.0%. In addition, for every 100 generated keys , 9 authorized keys are treated as fraud keys. Thus, certainly, the keys endeavor over the system.

The second step of the examination is to test a set of randomness tests to evaluate the multibiometric system. The randomness is evaluated by using the NIST Test Suite. The NIST Suite can evaluate the confusion and diffusion properties of an encryption scheme. The test judges, if the construction of considerable algorithms under test conditions shows features that the outputs are randomly generated.

To evaluate the multibiometric system, five basic tests are used to examine the performance of randomness characteristic of the generated key and also to evaluate the output sequences of key generator [31].

The five randomness tests are applied on different key sizes: 64, 28, 256, and 512 bits, as an outcome, keys that are extracted by the proposed key generator produce good robustness generate a key by effectively surpassing the five benchmark tests as illustrated in Table 3, Table 4 and Table 5.

It is noticed that the block frequency test is determined by 128 bits block length and the serial test is determined by 16 block lengths. Tables 4, Table 5, and Table 6 show the results of applying five randomness tests on these different keys produced from applying the proposal on facial and fingerprint images that are shown in Figure 1 and Figure 2.

Table 4- Randomness tests and their passing values for key with 64,128, 256 and 512 bits for face_img_1 and finger_img_1

Test		Freedom Degree	Key Size = 64 bits	Key Size = 128 bits	Key Size = 256 bits	Key Size = 512 bits
Frequency Test		Should be <=3.84	0.000	0.281	0.250	0.781
Runs Test	T0	Should be <=19.391	9.813	8.375	5.563	12.156
	T1		10.250	6.563	3.531	3.406
Poker Test		Should be <=11.1	5.300	4.850	4.700	6.994
Serial Test		Should be <=7.81	1.000	1.875	1.750	1.344
Auto Correlation Test	Shift No. 1	Should be <= 3.84	0.016	0.071	0.192	0.018
	Shift No. 2		4.129	0.794	1.906	2.831
	Shift No. 3		0.410	1.800	0.889	0.018
	Shift No. 4		2.400	0.516	0.254	0.638
	Shift No. 5		0.831	0.984	0.036	0.018
	Shift No. 6		0.621	0.820	0.016	0.008
	Shift No. 7		0.439	0.008	0.325	0.446
	Shift No. 8		1.786	0.033	0.581	0.286
	Shift No. 9		0.018	0.210	0.101	0.097
	Shift No. 10		1.185	0.305	0.000	0.590

Table 5: Randomness tests and their passing values for key with 64,128,256 and 512 bits for face_img_2 and finger_img_2.

Test		Freedom Degree	Key Size = 64 bits	Key Size = 128 bits	Key Size = 256 bits	Key Size = 512 bits
Frequency Test		Should be <=3.84	0.063	0.500	0.391	0.125
Runs Test	T0	Should be <=19.391	3.125	11.109	8.109	0.188
	T1		3.250	5.094	5.094	8.672
Poker Test		Should be <=11.1	3.125	4.225	3.450	7.650
Serial Test		Should be <=7.81	0.250	0.625	0.438	6.156
Auto Correlation Test	Shift No. 1	Should be 3.84	1.921	1.729	6.156	0.863
	Shift No. 2		2.323	0.142	0.863	2.008

	Shift No. 3		0.148	6.012	2.008	1.228
	Shift No. 4		0.600	2.286	1.228	0.504
	Shift No. 5		0.153	0.323	0.504	0.444
	Shift No. 6		0.276	0.256	0.444	0.387
	Shift No. 7		0.860	0.679	0.387	2.156
	Shift No. 8		0.643	0.258	2.156	0.071
	Shift No. 9		0.455	0.684	0.071	0.447
	Shift No. 10		2.667	0.065	0.447	7.657

Table 6: Randomness tests and their passing values for key with 64,128, 256 and 512 bits for face_img_3 and finger_img_3.

Test		**Freedom Degree**	**Key Size = 64 bits**	**Key Size = 128 bits**	**Key Size = 256 bits**	**Key Size = 512 bits**
Frequency Test		Should be <=3.84	0.250	0.031	0.250	0.281
Runs Test	T0	Should be <=19.391	11.313	12.375	6.438	8.391
	T1		1.000	1.938	2.156	13.109
Poker Test		Should be <=11.1	4.300	0.725	1.263	4.588
Serial Test		Should be <=7.81	4.250	0.125	1.250	0.375
Auto Correlation Test	Shift No. 1	Should be <= 3.84	0.016	0.386	2.075	8.785
	Shift No. 2		1.032	3.841	0.142	0.196
	Shift No. 3		0.803	0.008	0.004	0.002
	Shift No. 4		0.000	0.032	0.778	2.016
	Shift No. 5		0.017	0.073	0.195	0.333
	Shift No. 6		0.000	0.131	0.256	0.387
	Shift No. 7		0.018	0.074	0.197	0.018
	Shift No. 8		0.643	2.700	1.032	0.008
	Shift No. 9		0.018	0.076	1.785	1.449
	Shift No. 10		1.185	0.034	2.341	0.127

From the results listed in the above tables, the observation shows that the different key sizes produce good results of randomness tests, while key size 64 bits produce low acceptable results of randomness tests, thus a key with a large size is recommended for generating random keys.

Table 7 shows the comparison between the proposed system and two related works, any work using these metrics in a different scope, in the proposed system check the range of the threshold and key length, while in Abuguba [14], it checks the normalized range of Gabor filter, and in Maček [15] these metrics are used to check the hash function and brute force attack.

Table 7: The comparison between the proposed work and the related works which are listed in this paper.

	Multimodal Biometrics	Database	Key length	FAR%	FRR%
The proposed work	Face + Fingerprint	Essex Faces 95+SDUMLA-HMT	128	0.7%	10.4%
			256	0.0%	9.7%
Abuguba [14]	Face + Iris	ORL+ CASIA	128	/	9.59%
			256	/	12.51 %
Maček [15]	Iris + Fingerprint	CASIA-IrisV4+ CASIA-FingerprintV5	128	0%	2.62%
			256	0%	15.97%

6. DISCUSSION

In a biometric system, the biometric process for authentication is acceptable because of its suitability and possibility of giving security with non-repudiation, but there are some problems in this subject [32]:

1) It is possible to create a biometric template used in different applications as user authentication when a referenced template is exposed by an intruder; all the applications belonging to this template will be vulnerable to the intruder.

2) A matching process is required; the referenced biometric template is matched with the user biometric template and computes the difference~~s~~ measurements to determine whether the result is in the acceptable range or not. In case that reference template is stealing, then this biometric is lost forever.

There are solutions for these problems as mentioned in [33,34] may be addressed in the following two issues:

a) Different biometric keys can be created for the same person, these different biometric keys can be used to achieve privacy.

b) When an application has a referenced biometric template, and this template is exposed, then the original media of the biometric must never be lost, and it can re-create a new referenced biometric template from the original biometric media as mentioned.

In this proposal, new suggestion can be presented to solve two problems mentioned in [32]:

– As a simple and easy suggestion, if the biometric template is compromised, it is possible to generate a key by partitioning images to more than four parts or even partitioning images according to any geometrical shape, then calculate the highest density for each part, XORed these parts, diffusions process is applied on these parts including permutation and thresholding to produce a random key

- A more complicated and robust solution, this one is similar to the solution presented in [33], noise can be added to the original images: face and fingerprint, noise could be salt or pepper noise, noise also could be modeled as the first letter from person's name and added to images. As a result, three noised images are presented, and they are: facial image, fingerprint image, and image salting are XORed to create Initial Vector to be further processed to obtain a key. This solution will overcome the compromising of the key since the model or type of image noise is changed whenever the key is compromised.

7. CONCLUSIONS AND FUTURE WORKS

The biometric template offers a trustworthy approach to the trouble of a user civil authentication in electronic identification. This paper proposes a multibiometric system that uses the high-density area of facial and fingerprint images which play an important role in determining random and unique identification numbers. The facial and fingerprint images are passed through many operations to create permuted and confused random keys. The range of threshold is used to verify the best one by using FAR and FRR metrices, the result has proved that the best to use Threshold 5 with a 512-bit key. The random key~~,~~ is evaluated by using randomness tests. All tests have shown that the randomness of secured keys is efficient in the case of generating 64, 128, 256 and 512 bits length. In comparison with various previously developed biometric technologies , the experimental results showed that 128, 256, 512 bits secure keys are generated from the mixing of two biometric templates confirm the efficiency of the proposed approach to produce user-specific random identification keys than using 64-bit secure key since whenever the size of the key is large the randomization is best. In future work, many biometric templates could be used, such as palm vein, fingering vein, iris, voice, signature and retina for generated random key; keys are generated based on the extracted features from these templates.

References

[1] D. T. Meva, “Study of Various Multibiometric Techniques~~.”~~, "Proceedings of the 11th INDIACom; INDIACom-2017 IEEE Conference ID: 40353; *4th International Conference on “Computing for Sustainable Global Developm*ent”, 01st - 03rd March, Bharati Vidyapeeth's Institute of Computer Applications and Management (BVICAM), New Delhi (INDIA), pp. 6612-6614, 2017.

[2] W. Dahea and H. S. Fadewar, “Multimodal biometric system: A review,” *Int. J. Res. Adv. Eng. Technol.*, vol. 4, no. 1, pp. 25–31, 2018.

[3] H. Zhang, V. M. Patel, and R. Chellappa, “Robust multimodal recognition via multitask multivariate low-rank representations,” in *2015 11th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG)*, vol. 1, pp. 1–8, 2015.

[4] M. M. Ashish, G. R. Sinha, and R. P. Petal, “Biometric: fingerprints protection,” *Biom Biostat Int J*, vol. 7, no. 3, pp. 156–161, 2018.

[5] D. D. Salman and R. A.-A. Azeez, “BUILD CRYPTOGRAPHIC SYSTEM FROM MULTI-BIOMETRICS USING MEERKAT ALGORITHM,” *Iraqi J. Comput. Informatics ijci*, vol. 45, no. 2, pp. 1–8, 2019.

[6] D. D. Salman, R. A. Azeez, and A. M. J. Hossen, “Key generation from multibiometric system using meerkat algorithm,” *Eng. Technol. J.*, vol. 38, no. 3, pp. 115–127, 2020.

[7] G. N. S. A.M.A. Alrahawe, H.T. Vikas, “MULTI-BIOMETRIC TRAITS FUSION: A REVIEW,” *www.IJRAR.org*, vol. 7, no. 2, 2020.

[8] A. Jagadeesan, T. Thillaikkarasi, and K. Duraiswamy, “Cryptographic key generation from multiple biometric modalities: Fusing minutiae with iris feature,” *Int. J. Comput. Appl.*, vol. 2, no. 6, pp. 16–26, 2010.

[9] K. G. Salim, S. M. K. Al-alak, and M. J. Jawad, “Improved Image Security in Internet of Thing (IOT) Using Multiple Key AES,” *Baghdad Sci. J.*, vol. 18, no. 2, p. 417, 2021.

[10] N. H. M. Ali, A. M. S. Rahma, and A. S. Jamil, “Text Hiding in Color Images Using the Secret Key Transformation Function in GF (2 n),” *Iraqi J. Sci.*, vol. 56, no. 4B, pp. 3240–3245, 2015.

[11] M. S. Mahdi, R. A. Azeez, and N. F. Hassan, “A proposed lightweight image encryption using ChaCha with hyperchaotic maps,” *Period. Eng. Nat. Sci.*, vol. 8, no. 4, pp. 2138–2145, 2020.

[12] M. S. Mahdi, N. F. Hassan, and G. H. Abdul-Majeed, “An improved chacha algorithm for securing data on IoT devices,” *SN Appl. Sci.*, vol. 3, no. 4, pp. 1–9, 2021.

[13] S. Chandra, S. Paul, B. Saha, and S. Mitra, “Generate an Encryption Key by using Biometric Cryptosystems to secure transferring of Data over a Network,” *IOSR J. Comput. Eng.*, vol. 12, no. 1, pp. 16–22, 2013.

[14] S. Abuguba, M. M. Milosavljevic, and N. Macek, “An efficient approach to generating cryptographic keys from face and iris biometrics fused at the feature level,” *Int. J. Comput. Sci. Netw. Secur.*, vol. 15, no. 6, p. 6, 2015.

[15] N. Maček, B. Đorđević, J. Gavrilović, and K. Lalović, “An approach to robust biometric key generation system design,” *Acta Polytech. Hungarica*, vol. 12, no. 8, pp. 43–60, 2015.

[16] G. Balamurugan, K. B. Jayarraman, V. Arulalan, and V. Lokesh, “Multimodal biometric key generation for-cryptographic security using face and iris,” *Adv. Nat. Appl. Sci.*, vol. 9, no. 6 SE, pp. 525–531, 2015.

[17] S. Kanade, D. Petrovska-Delacrétaz, and B. Dorizzi, "Multi-biometrics based cryptographic key regeneration scheme," in *2009 IEEE 3rd International Conference on Biometrics: Theory, Applications, and Systems*, pp. 1–7, 2009.

[18] S.-W. Sun, C.-S. Lu, and P.-C. Chang, "Biometric template protection: A key-mixed template approach," in *2007 Digest of Technical Papers International Conference on Consumer Electronics*, 2007, pp. 1–2.

[19] N. F. Hassan and H. I. Abdulrazzaq, "Pose invariant palm vein identification system using convolutional neural network," *Baghdad Sci. J.*, vol. 15, no. 4, **pp. 502-509,** 2018.

[20] H. I. Abdulrazzaq and N. F. Hassan, "Modified Siamese Convolutional Neural Network for Fusion Multimodal Biometrics at Feature Level," in *2019 2nd Scientific Conference of Computer Sciences (SCCS)*, 2019, pp. 12–17.

[21] M. M. Ashish and G. R. Sinha, "Biometric template protection," *J. Biostat. Biometric Appl.*, vol. 1, no. 2, pp. 1–7, 2016.

[22] W. Yang, S. Wang, J. Hu, G. Zheng, and C. Valli, "Security and accuracy of fingerprint-based biometrics: A review," *Symmetry (Basel).*, vol. 11, no. 2, p. 141, 2019.

[23] H. Ayad, M. H. Abdulameer, L. E. George, and N. F. Hassan, "Descriptor trends in texture classification for material recognition," *Res. J. Appl. Sci. Eng. Technol.*, vol. 10, no. 10, pp. 1089–1101, 2015.

[24] A. S. Jamil, A. M. S. Rahma, and N. H. M. Ali, "Using Shape Representation to Design Panorama Video System," *AL-MANSOUR J.*, no. 25, 2016.

[25] S. Yuheng and Y. Hao, "Image segmentation algorithms overview," *arXiv Prepr. arXiv1707.02051*, 2017.

[26] S. N. Hasan, M. Gezer, R. A. Azeez, and S. Gülseçen, "Skin lesion segmentation by using deep learning techniques," in *2019 Medical Technologies Congress (TIPTEKNO)*, pp. 1–4, 2019.

[27] M. Abd El Aziz, A. A. Ewees, and A. E. Hassanien, "Whale optimization algorithm and moth-flame optimization for multilevel thresholding image segmentation," *Expert Syst. Appl.*, vol. 83, pp. 242–256, 2017.

[28] S. E. Umbaugh, *Digital image processing and analysis: human and computer vision applications with CVIPtools*. CRC press, 2010.

[29] Y. Yin, L. Liu, and X. Sun, "SDUMLA-HMT: a multimodal biometric database," in *Chinese Conference on Biometric Recognition*, 2011, pp. 260–268.

[30] N. F. Hassan, A. E. Ali, and T. W. Aldeen, "Generate Random Image-Key using Hash Technique," *Eng. Tech. J.*, vol. 28, no. 2, pp. 382–397, 2010.

[31] A. Rukhin, J. Soto, J. Nechvatal, M. Smid, and E. Barker, "A statistical test suite for random and pseudorandom number generators for cryptographic applications," Booz-allen and hamilton inc mclean va, 2001.

[32] U. Uludag, S. Pankanti, S. Prabhakar, and A. K. Jain, "Biometric cryptosystems: issues and challenges," *Proc. IEEE*, vol. 92, no. 6, pp. 948–960, 2004.

[33] N. K. Ratha, S. Chikkerur, J. H. Connell, and R. M. Bolle, "Generating cancelable fingerprint templates," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 561–572, 2007.

[34] A. R. Abbas and A. R. Kareem, "Age Estimation Using Support Vector Machine," *Iraqi J. Sci.*, vol. 59, no. 3C, pp. 1746–1756, 2018.