Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2016, Article ID 2670494, 15 pages
http://dx.doi.org/10.1155/2016/2670494

Research Article

A Fingerprint Image Encryption Scheme Based on Hyperchaotic Rössler Map

F. Abundiz-Pérez,[1] C. Cruz-Hernández,[2] M. A. Murillo-Escobar,[2] R. M. López-Gutiérrez,[1] and A. Arellano-Delgado[2]

[1]*Engineering, Architecture and Design Faculty, Autonomous University of Baja California (UABC), Ensenada, BC, Mexico*
[2]*Electronics and Telecommunications Department, Scientific Research and Advanced Studies of Ensenada (CICESE), Ensenada, BC, Mexico*

Correspondence should be addressed to C. Cruz-Hernández; ccruz@cicese.mx

Received 8 February 2016; Revised 16 June 2016; Accepted 22 June 2016

Academic Editor: Miguel A. F. Sanjuan

Currently, biometric identifiers have been used to identify or authenticate users in a biometric system to increase the security in access control systems. Nevertheless, there are several attacks on the biometric system to steal and recover the user's biometric trait. One of the most powerful attacks is extracting the fingerprint pattern when it is transmitted over communication lines between modules. In this paper, we present a novel fingerprint image encryption scheme based on hyperchaotic Rössler map to provide high security and secrecy in user's biometric trait, avoid identity theft, and increase the robustness of the biometric system. A complete security analysis is presented to justify the secrecy of the biometric trait by using our proposed scheme at statistical level with 100% of NPCR, low correlation, and uniform histograms. Therefore, it can be used in secure biometric access control systems.

1. Introduction

Nowadays, the biometric systems are widely used to authenticate and identify an individual, in order to recognize the user identity in a secure way. Nevertheless, these sophisticated recognition systems are prone to be attacked and the biometric identifier could be compromised. Identity fraud is a security problem in secure access systems controls. Therefore, there is an increasing interest in designing high and effective secure access systems based on biometric identifiers.

Techniques such as SHA-1, MD5, 3DES, RC5, AES, and IDEA are conventional encryption methods to protect information such as images and documents, when it is transmitted over on insecure communication channel. Nevertheless, they are not suitable for bulk and highly correlated data encryption such as images. On the other hand, there is an increasing research to design nonconventional encryption techniques such as chaotic and hyperchaotic encryption, since chaotic systems are related to cryptographic properties in confusion and diffusion process. In [1–10], chaotic encryption techniques are proposed to protect information such as text, grey images, color images, and fingerprint template. In the next paragraphs, recent chaotic encryption schemes related to biometric data protection by using chaos are discussed.

In [11], the authors proposed an encryption scheme by using the affine transform, the fractional wavelet packet transform (FrWPT), chaotic map, and the Hessenberg decomposition. They present a security analysis such as space keys, histogram, and another with good results. In [12], a fingerprint image encryption process is performed by using a chaotic Frequency Amplitude Phase Model (FAPM). Their scheme has high resistance against exhaustive attack with 10,238 combinations and the chaotic sequence presents a uniform distribution. In 2012, Liu presents a fingerprint image protection scheme by using two chaotic logistic maps [13]. Although the key space is enough to resist an exhaustive attack, the histogram distribution is far away from a good chaotic source for encryption process.

In [14], the authors proposed a multiple chaos-based biometric image cryptosystem for fingerprint security. The cryptosystem is constructed whit two 1D and two 3D chaotic

systems. Although their scheme presents a secret key space of 10^{154}, it encrypts low dimensional grey image in 0.5 seconds. Therefore, high dimensional color images could require a lot of processing time and the scheme could not be practical in real-time applications.

In [15], we present our first encryption approach, where the encryption process for biometric image is presented. In contrast, in this paper we have included an enhanced encryption process to provide higher security in the biometric trait and avoid identity theft. In addition, we have included a complete security analysis to verify the robustness of our scheme.

In [16], the authors presented an encryption scheme, based on a parallel full shuffling and encryption algorithm. In this paper, the logistic map, the Lorenz system, and the Chen system were used in the encryption algorithm. The encryption procedure is more confusing and complex, when the plain image is first divided into 4 subimages, and then the position of each subimage is changed pseudorandomly by using the logistic map. Furthermore, a full shuffling matrix is used to shuffle the position of pixels in the whole image and the subimages are encrypted simultaneously in a parallel manner. The results presented by the authors show good results in the correlation analysis and great sensitivity at key, but it has a slow speed encryption.

In [17], the authors presented an image encryption algorithm based on genetic recombination and hyperchaotic system. First, the plain image is expanded into two compound images, which are composed of selected four bit-plane and diffuse bit-plane levels. Then, the compound bit-planes and key streams are reconstructed based on the principles of genetic recombination. Finally, they perform traditional diffusion to produce the cipher images. Experimental results demonstrate the effectiveness of encryption with good results in the histograms, large space of keys, and high sensitivity at secret key, which are similar to the results obtained by us in this work.

In [18], the authors presented an encryption scheme based on image operations with DNA sequences and a chaotic system. First, they perform bitwise operations XOR by using the pseudorandom sequences produced by the chaotic system CML. After that, a DNA matrix is generated by encoding the confused image using a DNA coding rule. Subsequently, they generate new initial conditions of CML according to the DNA array and initial conditions. Then, the rows and columns of DNA matrix are permuted and the DNA matrix is confused again. Finally, after decoding the confused DNA matrix using a kind of DNA decoding rule, the encryption image is obtained. Experimental results demonstrate the effectiveness of encryption, and the authors show good results in the histogram analysis, a large space of keys, and a good sensitivity at key, which are similar to the results obtained by us in this work.

Nowadays, in literature there are several approaches of implementation of chaotic systems in cryptography, because chaotic systems present cryptographic properties related to confusion and diffusion such as sensitivity to initial conditions and control parameters, mixing, determinism, and ergodicity [1, 2, 12, 13, 15]. In recent years, researchers have introduced the hyperchaotic systems in the cryptography field [3, 4, 9, 19]. The hyperchaotic systems present higher sensitivity to initial conditions and control parameter, higher space of keys, and more complex dynamic characteristics and have two or more positive Lyapunov exponents, which indicates that their dynamic sequence diverges to a greater extent than a chaotic system. All these advantages are very useful to produce a cryptogram with better statistical properties.

In this paper, we present an image encryption scheme by using hyperchaotic Rössler map. We use the high pseudorandom sequences to generate excellent encryption effects and produce a highly secure encryption scheme, which present better results in some aspects compared with [16–18] such as uniform distribution histograms, the large space of key, low correlation of pixels, and speed encryption.

The organization of this work is as follows. In Section 2, some vulnerabilities of a biometric system are presented. In Section 3, we describe briefly the hyperchaotic system used in the encryption process. The encryption and decryption process details are presented in Section 4. In Section 5, we present a complete security analysis at statistical level to verify the capabilities of our scheme in security terms. In Section 6, we present a comparison of our proposed scheme with recent schemes reported in recent years. Finally, the conclusions are mentioned in Section 7.

2. Vulnerabilities in a Biometric System

There are several biometric identification systems, which are composed of two important stages, enrollment and identification. In enrollment stage, the user is registered in the database, while in the identification stage the user's identity is determined by using a biometric identifier. The biometric identification systems can be divided into five subsystems: data collection, transmission, signal processing, decision, and data storage. In Figure 1, these five subsystems and their iteration are shown (the dashed lines showing the enrollment stage) [20–23].

Currently, the biometric systems are a topic of high interest in the scientific community, because they provide a practical way for secure access control systems. Nevertheless, these systems have some vulnerable points, which are classified in two categories. One of these vulnerabilities is the attack on the communication lines. A snooper can spy the communication to steal confidential information of the biometric identifier, which can be used to extract the user identity. The second attack is on the modules (sensor, feature extraction, matching, database, etc.). The attacks use malicious programs such as a Trojan horse and it emulates the function of some modules of the biometric system and could reject an authorized user [20].

Due to the existence of attacks in the vulnerability points of the biometric systems, the scientific community and engineers have implemented some actions to protect the biometric system against these powerful attacks. Some of the proposed schemes are based on random data, withheld data, on-life detection, biometric multiple, cryptography, digital

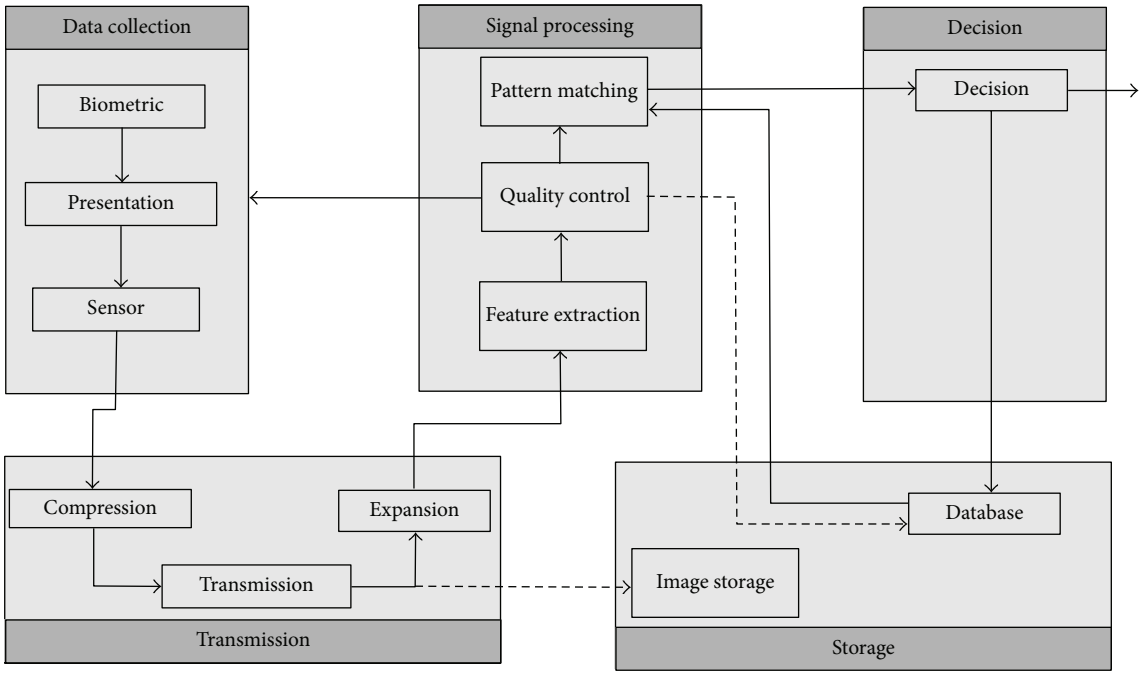


Figure 1: General biometrics system [20].

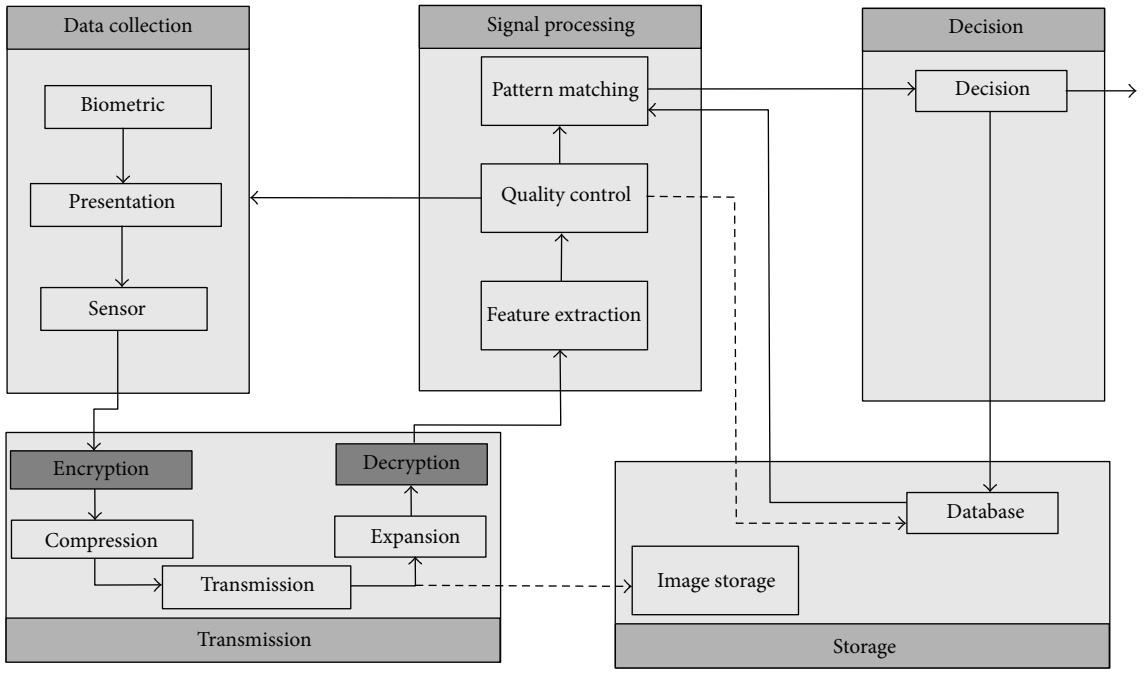


Figure 2: Possible location of the encryption and decryption process on general biometric system.

signature, network clean-up, and physical security [20–23, 27–31].

In this paper, we propose an encryption scheme that provides security in the transmission subsystem, with the aim of protecting the communication line, where the image is sent to the storage subsystem and the signal processing subsystem; see Figure 2. The proposed encryption scheme is based on hyperchaotic Rössler map.

3. Hyperchaotic Rössler Map

Analogous or digital communication schemes need new cryptographic schemes to protect confidential information. Motivated by this fact, in recent years several researchers have reported great variety of advances related to chaotic encryption. These schemes exploit the pseudorandom properties of the states in a chaotic system; see, for example, [4–10].

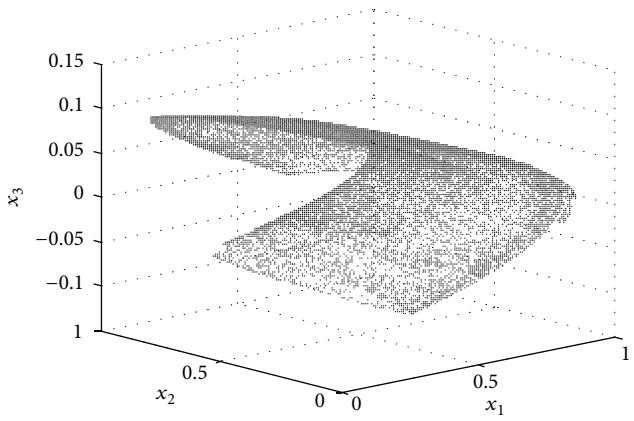


Figure 3: Hyperchaotic attractor, generated by the Rössler map.

In addition, some chaotic characteristics that benefit the encryption are as follows:

(i) Simple operation can generate complex dynamics, which provides a pseudorandom sequence where the confidential information can be hidden.

(ii) Small variation in initial conditions in chaotic system provides great changes in the output dynamic, which benefits the number of keys that could be used for encryption.

(iii) Encryption statistics preserve the uniform distribution for any chaotic sequence, which benefit the encryption against statistical attacks.

In this paper, we use the Rössler map for encryption purposes. This map generates hyperchaotic dynamics; that is, it presents greater complex behavior than a chaotic system. One distinctive characteristic of these systems is the existence of more than one positive Lyapunov exponent [4–6]. The following are the Rössler map equations [5]:

$$\begin{aligned} x_1(k+1) &= \alpha x_1(k)\left(1 - x_1(k)\right) \\ &\quad - \beta\left(x_3(k) + \gamma\right)\left(1 - 2x_2(k)\right), \\ x_2(k+1) &= \delta x_2(k)\left(1 - x_2(k)\right) + \varsigma x_3(k), \\ x_3(k+1) &= \eta\left(\left(x_3(k) + \gamma\right)\left(1 - 2x_2(k)\right) - 1\right)\left(1 - \theta x_1(k)\right). \end{aligned} \tag{1}$$

The hyperchaotic attractor generated by Rössler map is shown in Figure 3 considering the initial conditions $x(0) = (0.3, 0, 0.05)$, parameter values $\alpha = 3.8$, $\beta = 0.05$, $\gamma = 0.35$, $\delta = 3.78$, $\varsigma = 0.2$, $\eta = 0.1$, and $\theta = 1.9$, and 28,000 iterations by using MATLAB software simulation.

4. Encryption and Decryption Process

The encryption process is based on two important stages, the diffusing and permutation stage. On the other hand, the decryption process is constituted by the inverse permutation and inverse diffusing process (Figure 4).

4.1. Encryption Process. Hyperchaotic Rössler map is used to generate a sequence of pseudorandom numbers, which are used in permutation and diffusion process.

4.1.1. Diffusion Process

Step 1. Read the fingerprint plain image to generate $I_o(i, j)$, where $i = 1, 2, \ldots, n$ and $j = 1, 2, \ldots, m$. It represents an array of $n \times m$, where n is the number of rows and m is the number of columns.

Step 2. Generate an array of $n \times m$, which contains the output data from the hyperchaotic state $x_1(k)$ from the Rössler map, where $k = 1, \ldots, N$ and $N = n \times m$. This array will be named as hyperchaotic state matrix and it will be represented by $M_c(i, j)$.

Step 3. By using $M_c(i, j)$, we generate a new array in the set of $\{a \in \mathbb{N} : 0 \le a < 256\}$. This array is used to confuse the image. To perform this step, we transform $M_c(i, j) \in \mathbb{R}$ in the range 0 to 1 to the set $\{a \in \mathbb{N} : 0 \le a < 256\}$ by using the next expression:

$$\begin{aligned} &M_{cc} \\ &= \left(\left\{M_c(i, j) : \left(\left|M_c(i, j)\right| \times 100000\right) \in \mathbb{N}\right\} \bmod 256\right). \end{aligned} \tag{2}$$

Step 4. Add elements of the array $M_{cc}(i, j)$ to elements of plain image $I_o(i, j)$ with module 256. With this process, we will get a new array that will be named as dithered image matrix and will be represented by $I_c(i, j)$.

4.1.2. Permutation Process

Step 1. Generate a vector for the positions of the rows and a vector for the positions of the columns, which will be represented by the variables $V_{\text{row}} = \begin{bmatrix} 1 & 2 & 3 & 4 & \cdots & n \end{bmatrix}$ and $V_{\text{col}} = \begin{bmatrix} 1 & 2 & 3 & 4 & \cdots & m \end{bmatrix}$.

Step 2. Generate two vectors that contain the data resulting from chaotic states x_2 and x_3 from the Rössler map; these will be appointed as chaotic vector of rows and chaotic vector of columns, which will be represented by the variables $X_c(k_1) = \begin{bmatrix} x_2(0) & x_2(1) & \cdots & x_2(N) \end{bmatrix}$ and $X_r(k_1) = \begin{bmatrix} x_3(0) & x_3(1) & \cdots & x_3(N) \end{bmatrix}$, where $k_1 = 1, 2, 3, \ldots, N$ and $N > (21m)$.

Step 3. Generate two vectors of pseudorandom sequences for column and row permutation by using the following expression:

$$\begin{aligned} S_{\text{ren}} &= \left(\left\{X_r : \left(\left|X_r\right| \times 100000\right) \in N\right\} \bmod n\right) + 1, \\ S_{\text{col}} &= \left(\left\{X_c : \left(\left|X_c\right| \times 100000\right) \in N\right\} \bmod m\right) + 1. \end{aligned} \tag{3}$$

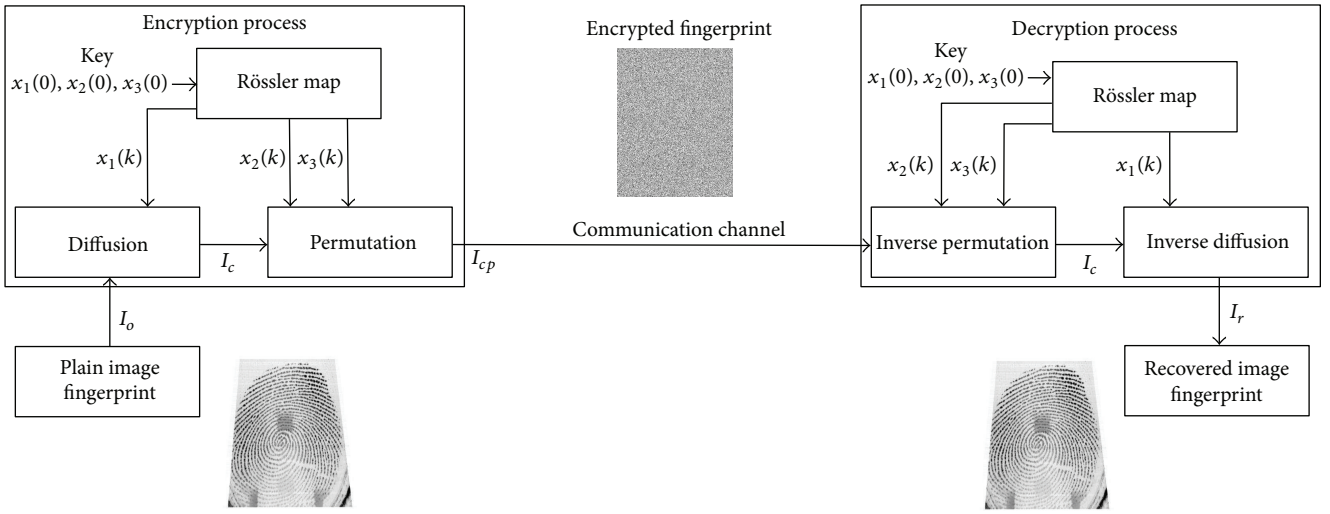


Figure 4: Block diagram of the proposed encryption algorithm.

Figure 5: Program implemented in MATLAB 2008a version.

After that, $V_{\text{row}}, V_{\text{col}}, S_{\text{ren}}$, and S_{col} are used to calculate two pseudorandom vectors to permute the diffused image. They will be used in Algorithm 1.

Step 4. The pixels from the diffused image will be permuted as follows:

$$I_{cp}(i,j) = I_{in}\left(V_{\text{sap}_r}(i), V_{\text{sap}_c}(j)\right). \tag{4}$$

4.2. Decryption Process. The decryption process is based on the inverse steps of the encryption process. The hyperchaotic Rössler map is used with the same initial conditions and control parameters. Basically, the decryption steps are described as follows.

4.2.1. Inverse Permutation Process. In this process, $I_c(i,j) = I_{cp}(i,j)$ to perform the reverse permutation. The next expression is calculated:

$$I_c(i,j) = I_{in}\left(V_{\text{sap}_r}(i), V_{\text{sap}_c}(j)\right). \tag{5}$$

4.2.2. Inverse Diffusion Process. The steps to perform the inverse blurring process are similar to those presented in the diffusion stage. Performing the reverse blurring equally will use the matrices $M_{cc}(i,j)$ and $I_c(i,j)$, but, in this case, the operation is performed as follows:

$$I_r(i,j) = \left(I_c(i,j) - M_{cc}(i,j)\right) \bmod 256. \tag{6}$$

Finally, I_r is the recovered fingerprint plain image.

5. Experimental Results

This section presents the experimental results of the proposed fingerprint encryption algorithm implemented in MATLAB simulation software 2008a (Figure 5). The results are divided into two subsections. The first part presents the cipher, decipher, permutation, diffusion, inverse permutation, and inverse diffusion stages at image level. The second subsection presents the security analysis of the proposed encryption algorithm.

Require: Order variables $N = 21m$, $S_{cao}(k_1) = S_{ren}(k_1)$, $V_p(l) = V_{row}(i)$ and $N_2 = n$ for generating pseudorandom sequence from the rows positions or $S_{cao}(k_1) = S_{col}(k_1)$, $V_p(l) = V_{col}(j)$ and $N_2 = m$ for generating pseudorandom sequence the columns positions.
Ensure: V_{sap} Vector pseudorandom sequence of positions.
(1) $p = 1$; $a = 1$; $b = N/21$;
(2) **while** $a \neq 0$ **do**
(3) **if** $S_{cao}(b)$ is equal to $V_p(S_{cao}(b))$ **then**
(4) $V_{sap}(p) = S_{cao}(b)$
(5) $V_p(S_{cao}(b)) = 0$; $p = p + 1$
(6) **end if**
(7) $b = b + 1$
(8) **if** sum of all elements of V_p is equal to 0 **then**
(9) $a = 0$
(10) **end if**
(11) **if** b is equal to N **then**
(12) $b = 1$
(13) $S_{cao} = ((S_{cao} + 1) \bmod N_2) + 1$
(14) **end if**
(15) **end while**
(16) **return** V_{sap}

Algorithm 1: Generation of pseudorandom sequences of positions.

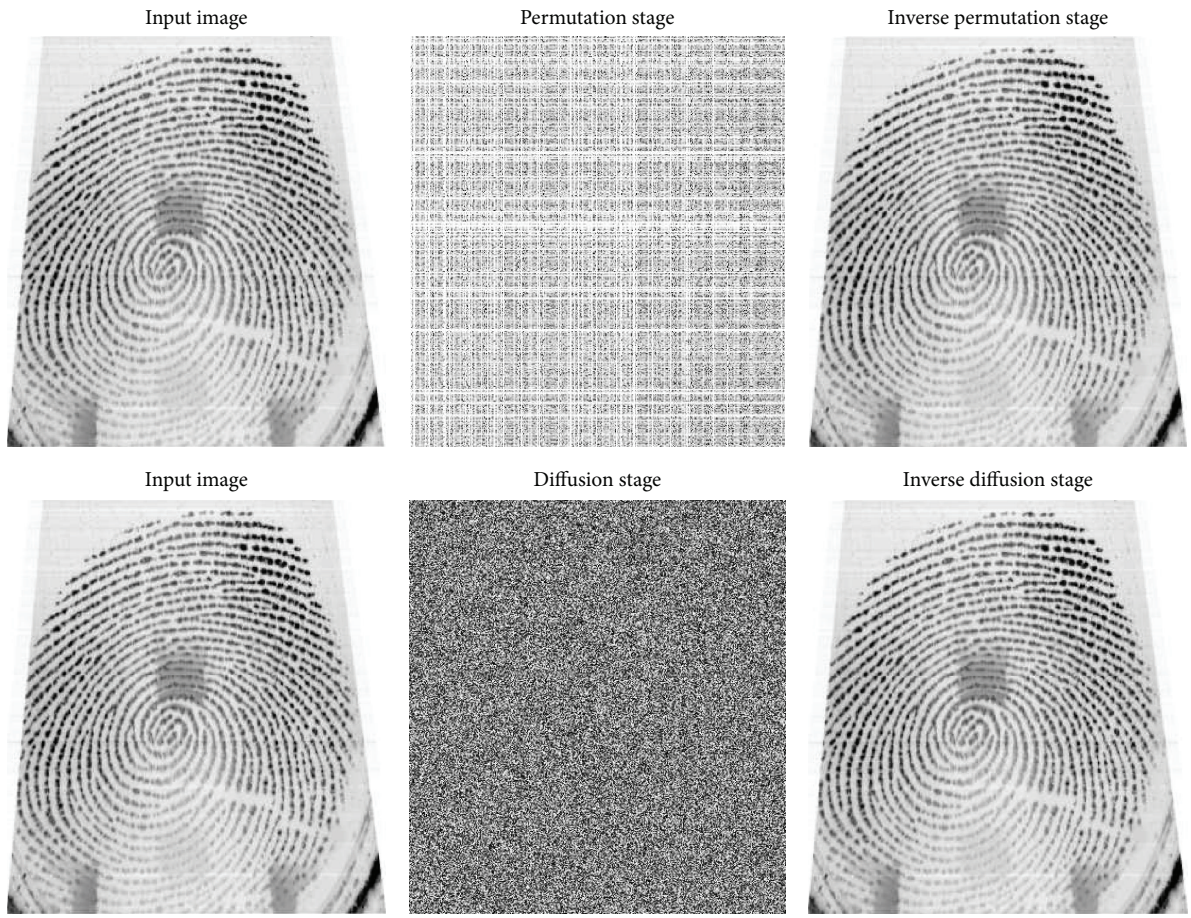


Figure 6: Encryption and decryption process in permutation and diffusion process.

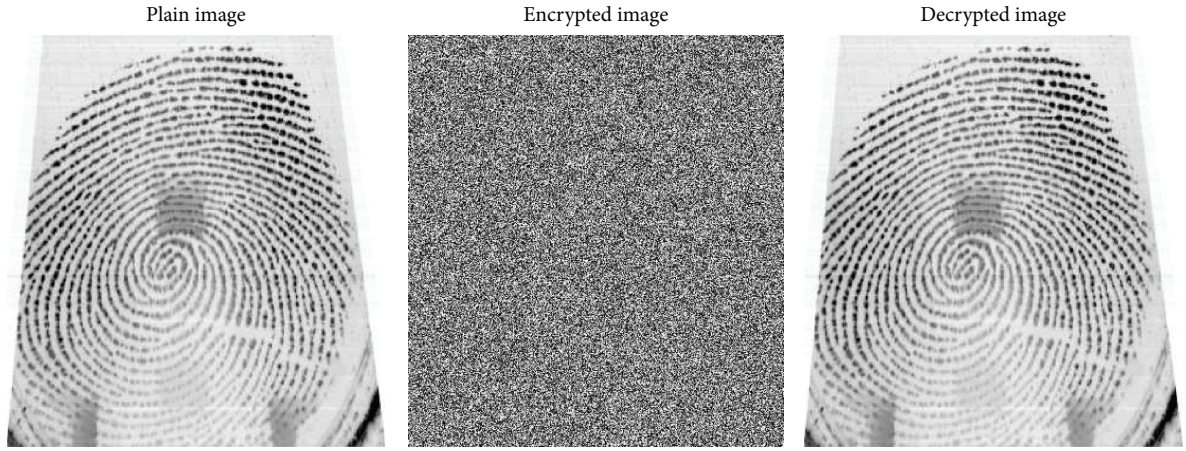


Figure 7: Decryption process with correct secret key.

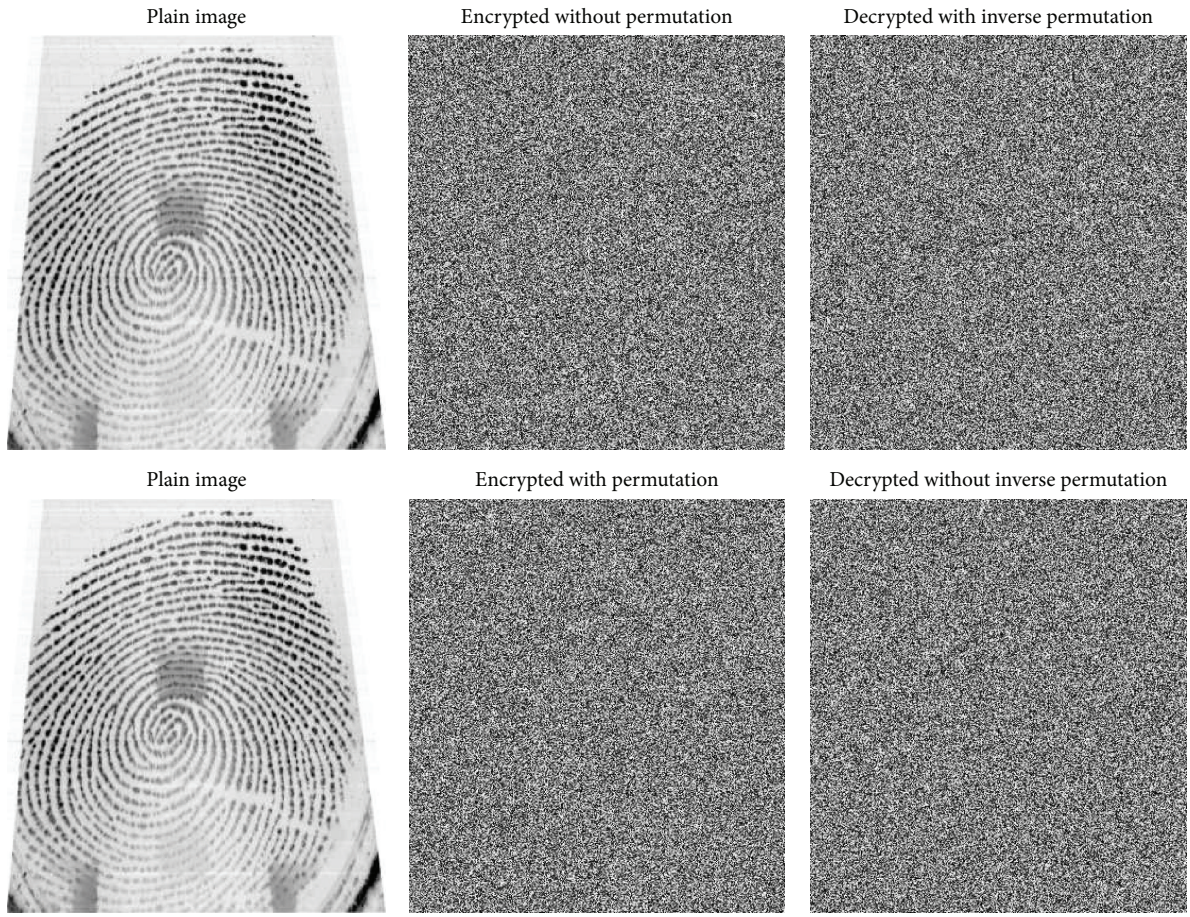


Figure 8: Decryption process with an omitted step.

5.1. Encryption and Decryption Processes. In Figure 6, we present the permutation and diffusion encryption processes separately. This provides an idea of what performs these steps.

Figure 7 illustrates the results of the hyperchaotic encryption and decryption processes, without omitting any steps that are performed in each process. The encrypted image does not show any information at human eye level. In addition, the recovery image is the same as the plain image, if the same key is used in the hyperchaotic map.

If any of the stages is not implemented, that is, diffusion or permutation that is performed in the encryption or decryption process, the image will not retrieve. Figure 8 shows this idea, where the plain image is not recovered.

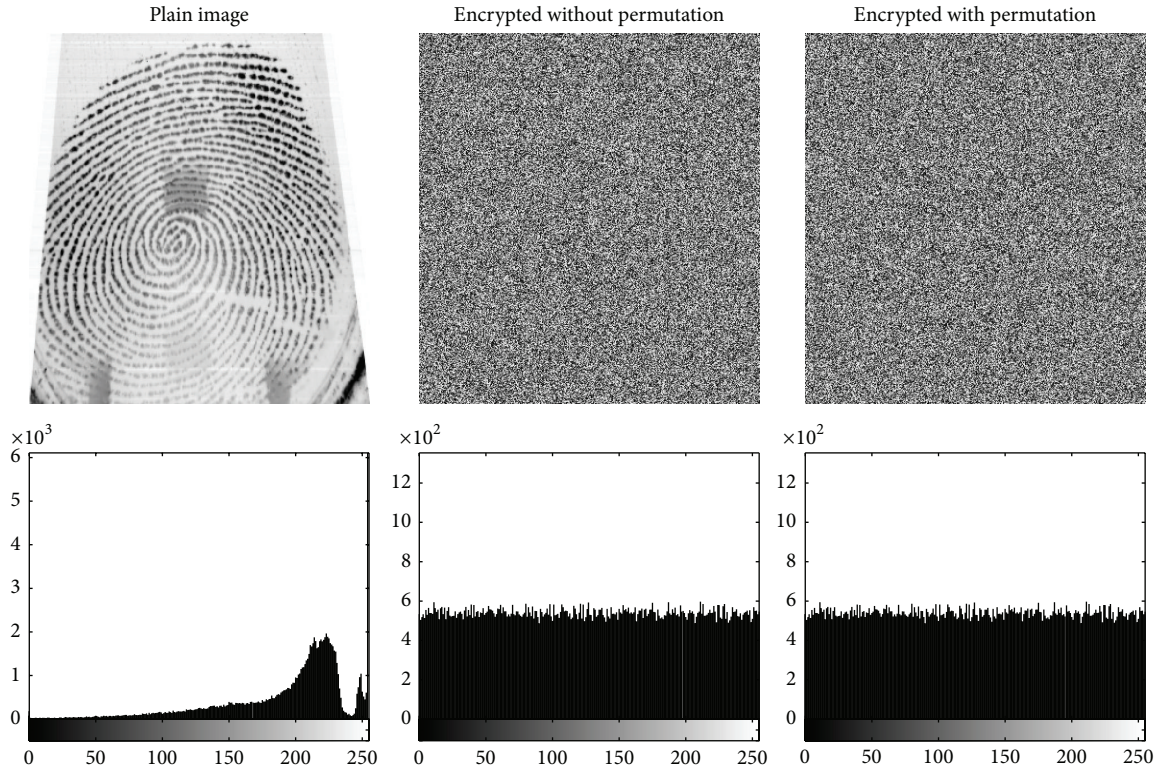


Figure 9: Histogram analysis of encryption process with and without permutation process, by using the key (0.1, 0.1, 0.1).

5.2. Security Analysis. To evaluate the security at statistical level of the proposed fingerprint hyperchaotic encryption algorithm, we considered different types of attacks. Attacks such as exhaustive attack, statistical attack, and differential attack are analyzed. In some cases, we show the results for both encryptions with and without permutation process.

5.2.1. Statistical Analysis. In this analysis, the distribution and correlation are considered. The histogram of the image can give visual information of the distribution of the intensity levels of red, green, and blue component of a color image. In addition, the numerical correlation is calculated according to a specific expression to determine if the encrypted image presents low correlation, which is desired in a good encryption algorithm.

Histogram Analysis. Cryptographic algorithms must provide a uniform distribution of grey values in the histogram in order to provide adequate strength against a statistical attack. In Figure 9, we present the histograms of the image before and after encryption, considering that the encryption is performed with or without the permutation stage. It can be seen that the distribution of grey values from the original image is concentrated in some grey scale values, whereas the histograms of the encrypted images are nearly uniform. Also, it shows that the encrypted image histograms are the same with or without the permutation stage, because the permutation process only changes the positions of the pixels contained in the image and it does not modify the values of the pixels and therefore does not affect the distribution of the intensity levels of the encrypted image.

Correlation Analysis. Typically, a plain image presents high correlation between adjacent pixels. In this sense, cryptographic algorithms should reduce the correlation in order to provide adequate strength against statistical attack. In Figure 10, we present the results of the horizontal, vertical, and diagonal correlation from the adjacent pixels of the encrypted image with and without permutation process.

The correlation analysis is calculated as follows:

$$E(x) = \frac{1}{N}\sum_{i=1}^{N} x_i,$$
$$E(y) = \frac{1}{N}\sum_{i=1}^{N} y_i, \tag{7}$$

$$D(x) = \frac{1}{N}\sum_{i=1}^{N} \left(x_i - E(x)\right)^2,$$
$$D(y) = \frac{1}{N}\sum_{i=1}^{N} \left(y_i - E(y)\right)^2, \tag{8}$$

$$\operatorname{cov}(x, y) = \frac{1}{N}\sum_{i=1}^{N} \left(x_i - E(x)\right)\left(y_i - E(y)\right), \tag{9}$$

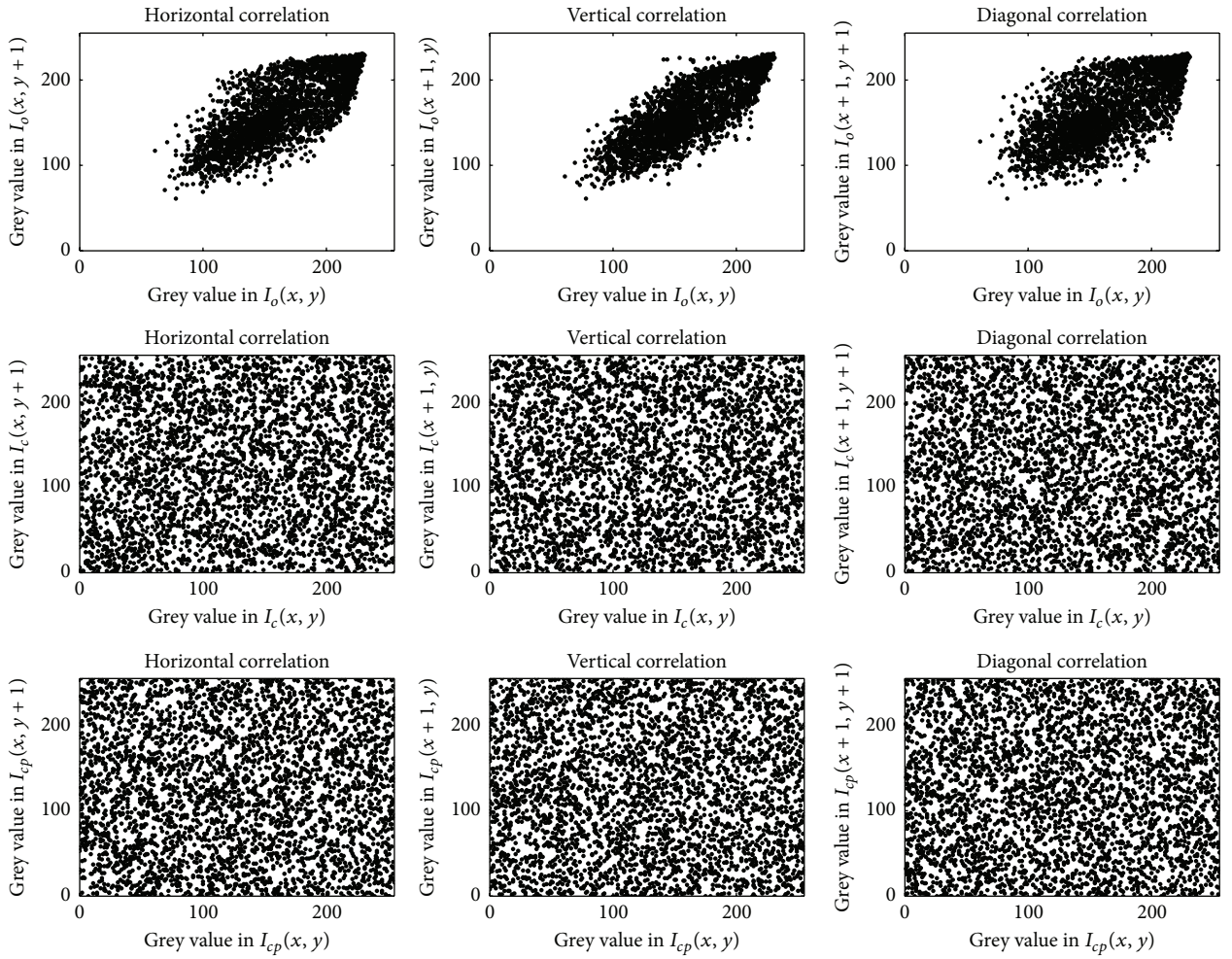


Figure 10: Horizontal, vertical, and diagonal correlation distribution of the original image (I_o), encrypted image without permutation (I_c), and encrypted image with permutation (I_{cp}), by using the key (0.1, 0.1, 0.1).

$$r_{xy} = \frac{\operatorname{cov}(x, y)}{\sqrt{D(x)} \times \sqrt{D(y)}}, \tag{10}$$

where x and y are the grey values of two adjacent pixels in the image, $\operatorname{cov}(x, y)$ is the covariance, D is the variance, and E is mean. Table 1 shows the numerical results of the correlations of the encrypted image with and without permutation process.

By observing the results presented in Figure 10, there are concentration points in plain image correlation, which indicates that there is a high correlation between the adjacent pixels. However, the encrypted image with and without permutation and the plot show low correlation. In numerical results, the values close to ±1 indicate high correlation and values close to 0 indicate low correlation. The results of Table 1 showed low correlation for plain image with approximately 0.8 (close to 1); therefore, this value is an indicator of the high correlation. On the other hand, the encrypted image presents low correlation (close to 0); therefore the encryption process is robust.

Table 1: Correlation coefficient of encrypted image.

Image	Horizontal correlation	Vertical correlation	Diagonal correlation
Plain image	0.8180	0.8670	0.8184
Encryption without permutation	0.0392	0.0082	0.0392
Encryption with permutation	0.0199	−0.0045	0.0199

5.2.2. Exhaustive Attack. This rudimentary but effective attack is known as brute force attack, in which all possible keys are tried until the correct secret key is found and the original message is decrypted. This type of attack is related to the key. Therefore, we check the efficiency of the key in the cryptographic scheme with a key space analysis and secret key sensitivity analysis.

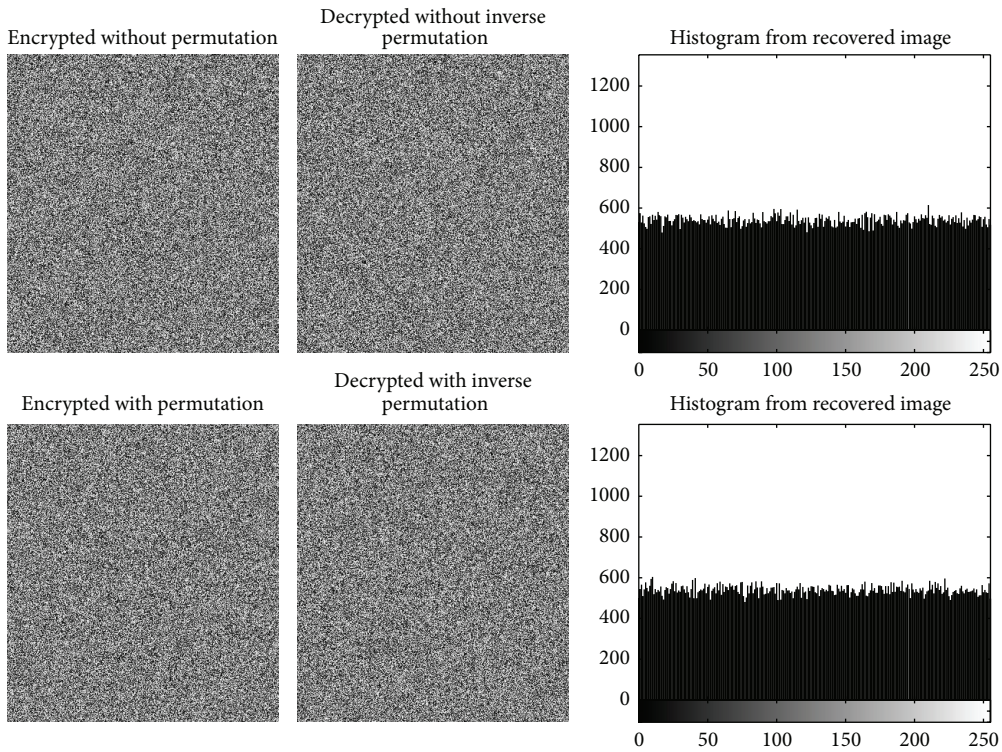
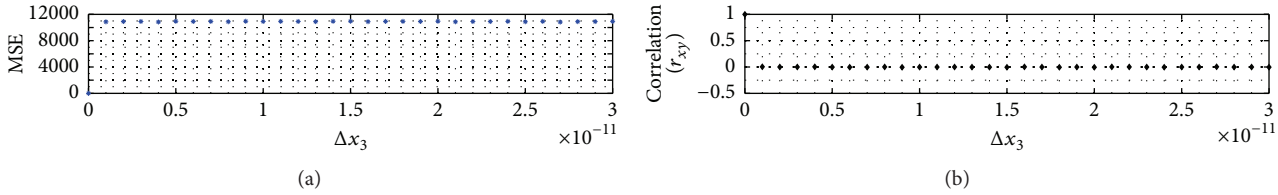


Figure 11: Secret key sensitivity analysis.



(a) (b)

Figure 12: Secret key sensitivity analysis: (a) MSE curves and (b) correlation curves.

Key Space Analysis. The key space must be greater than $2^{100} = 1.2677 \times 10^{30}$ to resist an exhaustive attack, according to Alvarez and Li reported in [32]. The key used in the encryption scheme is based on the initial condition of the hyperchaotic Rössler map (x_1, x_2, x_3). Therefore, with a precision of 10^{12}, the encryption scheme provides a key space of $10^{12} \times 10^{12} \times 10^{12} = 10^{36}$. In addition, the key space can be increased by 10^{150} if we considered the parameter controls α, β, γ, δ, ς, η, and θ of the hyperchaotic Rössler map and a precision of 10^{15}. Therefore, the key space is suitable to resist an exhaustive attack.

Secret Key Sensitivity Analysis. The cryptographic scheme must be highly sensitive to small variations in the key to resist differential attack; that is, similar secret keys must generate totally different encrypted images. Chaotic systems present highly sensitive to initial conditions; that is, they present different chaotic dynamics when the initial conditions are slightly modified. In Figure 11, we present two encrypted images with and without permutation process by using the key $(0.1, 0.1, 0.1)$ and the second column presents the decrypted image by using incorrect secret key $(0.1, 0.1, 0.100000000001)$. Therefore, the decrypted image cannot be recovered if the secret key is not exactly used such as in encryption process.

In order to show the secret key sensitivity in the encryption process, the correlation (see (10)) and MSE (see (12)) are used. In this analysis, two cryptograms are generated by using the same plain image but with two similar secret keys, which are highly similar to each other. Several pairs of cryptograms are produced by using $(0.1, 0.1, 0.1)$ and $(0.1, 0.1, 0.1 + \Delta x_3)$ as secret keys. In each pair of cryptograms, the mean square error MSE (see Section 5.3) and the correlation coefficient are calculated. In Figure 12(a), the MSE results of several pairs of cryptograms are showed by using increments of Δx_3. Since the MSE values are high and uniform, the proposed scheme presents high sensitivity at small variations in secret keys. On the other hand, in Figure 12(b), the correlation of the same pairs of cryptograms is determined. The results of correlation are close to zero in each test, which indicates that both cryptograms are highly different from each other and

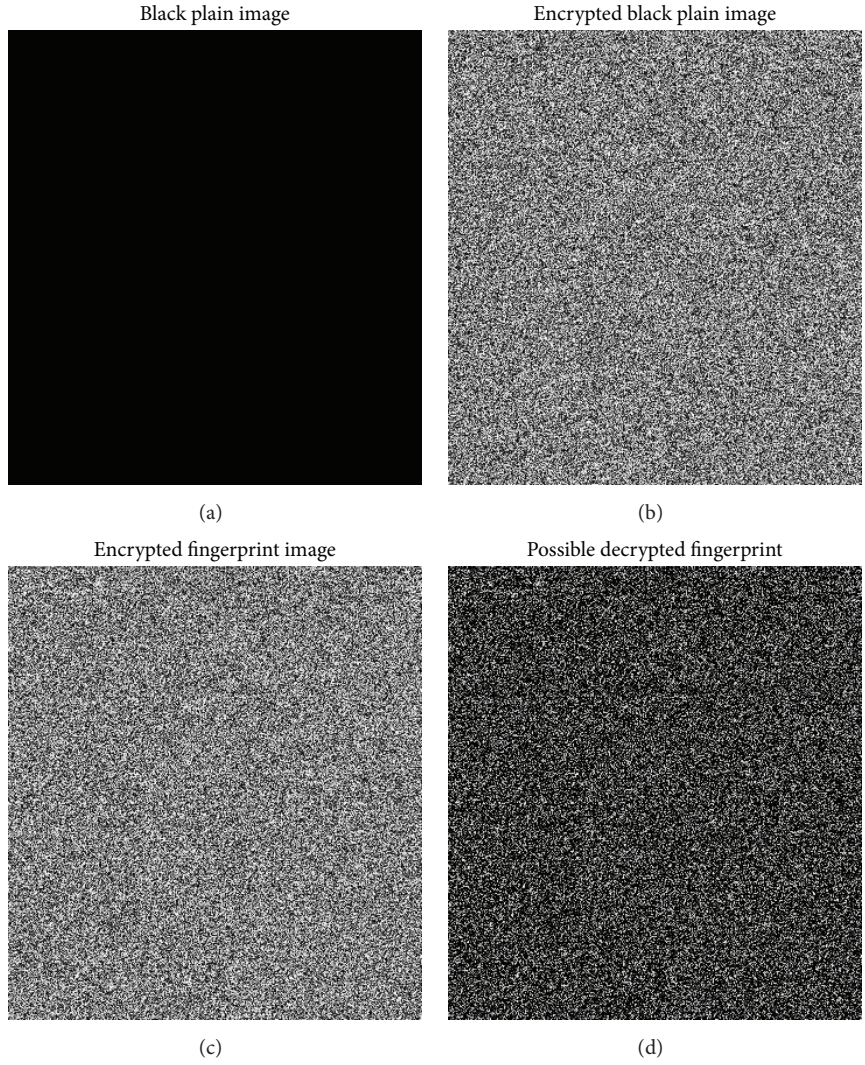


Figure 13: Chosen plain image attack: (a) chosen black plain image, (b) cryptogram of black image and possible secret key, (c) cryptogram of fingerprint image, and (d) decryption of encrypted fingerprint with the possible secret key.

this is another proof of the high sensitivity of secret key in the proposed scheme.

5.2.3. Differential Analysis. The third analysis presented is against differential attack. If the encryption process is weak, an adversary could implement this attack to find a relation between similar plain images and determine the secret key. The analysis consists in encrypting two similar plain images with a small change in just one pixel. After that, the encryption algorithm is applied to both of them by using the same secret key. Then, a comparison between the encrypted images is performed. There are two parameters used to examine the resistance against differential attack, which are NPCR (Number of Pixels Change Rate) and UACI (Unified Changing Average Intensity). These values are calculated as follows:

$$C(i,j) = \begin{cases} 0 & \text{if } T_1(i,j) = T_2(i,j) \\ 1 & \text{if } T_1(i,j) \neq T_2(i,j), \end{cases}$$

$$\text{NPCR} = \frac{\sum_i^M \sum_j^N C(i,j)}{M \times N} \times 100\%,$$

$$\text{UACI} = \frac{\sum_i^M \sum_j^N |T_1(i,j) - T_2(i,j)|}{255 \times M \times N} \times 100\%, \tag{11}$$

where M and N are the height and width of the image and $T_1(i,j)$ and $T_2(i,j)$ are the pixel value in the location (i,j) of the encrypted images. In this analysis, we considered the plain fingerprint image showed in Figure 8. The value of the pixel 127 of the original image is changed to the value of 128. Subsequently, the encryption is applied to both images. Therefore, two cryptograms are generated $T_1(i,j)$ and $T_2(i,j)$. In the encryption process without permutation process, we have $\text{NPCR}_c = 100\%$ and $\text{UACI}_c = 0.7850\%$. This indicates that the cryptographic scheme is robust against differential attack.

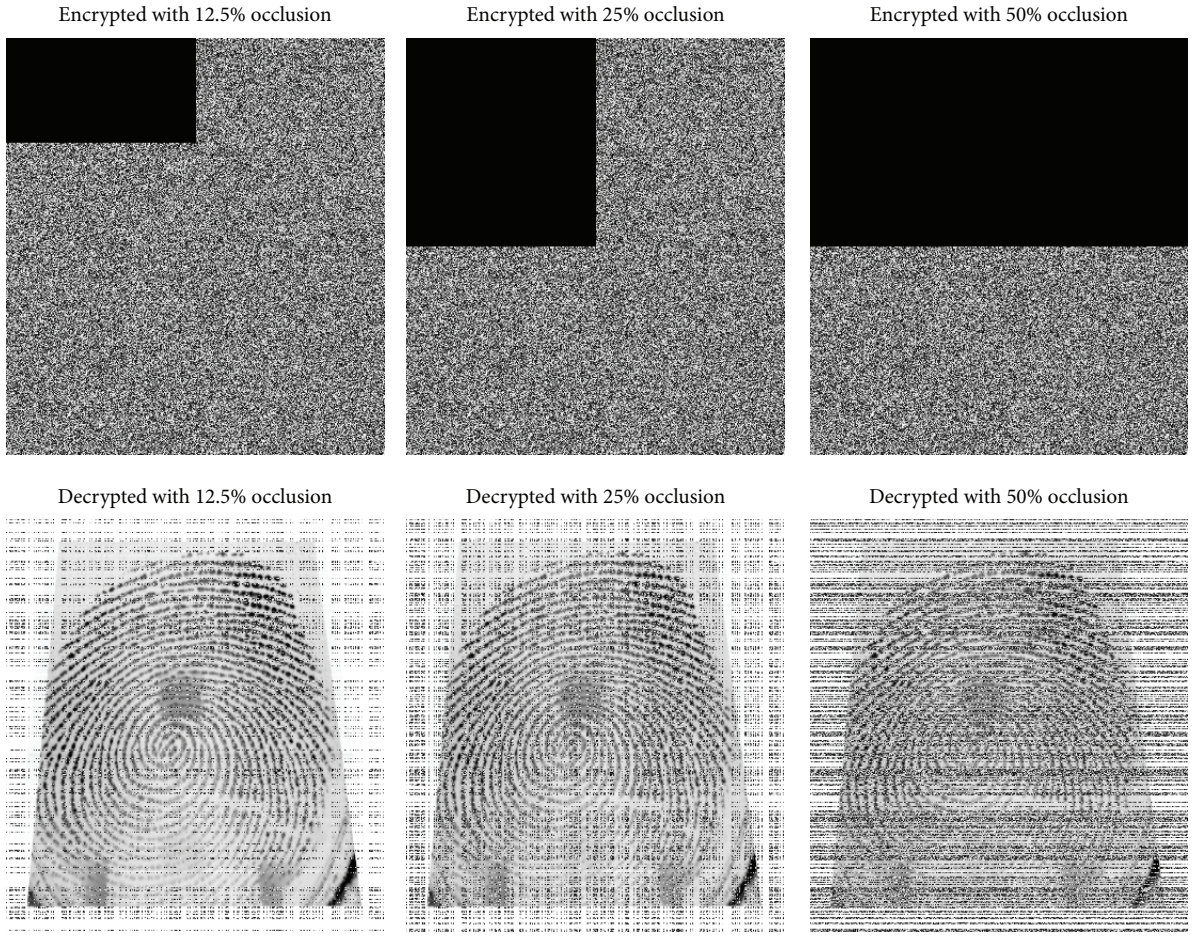


Figure 14: Tolerance against occlusion attacks, considering data loss of 12.5%, 25%, and 50%.

5.3. Mean Square Error Analysis. A method to determine the error between original image and encrypted image is by using the mean square error MSE parameter, which is the existing qualitative squared error between both compared images [33]. The MSE parameter is calculated by the following expression:

$$\text{MSE} = \frac{1}{M \times N} \sum_{i=1}^{M} \sum_{j=1}^{N} \left| F_1(i, j) - F_2(i, j) \right|^2, \qquad (12)$$

where $F_1(i, j)$ is the original image, $F_2(i, j)$ is the encrypted image, and $M \times N$ is size of the original image. Higher values of MSE mean higher immunity against attacks. MSE values close to zero mean that both tested images are highly similar. The MSE calculated from original image and encrypted image showed in Figure 7 is MSE = 12324.3893. On the other hand, the MSE between original image and the correct retrieved image is MSE = 0.

5.4. Chosen/Known Plain Image Attack. The chosen/known plain image attack is a powerful cryptanalyst attack, which has broken several image encryption algorithms based on chaos. In a chosen plain image attack, the cryptanalyst chose a convenient image, for example, an image with all pixels in black to eliminate the function of the plain image over the algorithm (permutation and diffusion) and try to find the secret key (chaos), since its pixel values are zero.

In Figure 13, we present the chosen plain image attack by using the black plain image, which is shown in Figure 13(a). The corresponding encrypted black plain image is shown in Figure 13(b), which can be represented by the secret key or the chaotic sequence used in several cryptograms. Then, the cryptanalyst uses this information as a possible secret key and tries to decrypt other cryptograms that probably were encrypted with that secret key. In this case, the cryptanalyst used the cryptogram of the fingerprint used in Figure 7, which is shown in Figure 13(c). Nevertheless, the corresponding decrypted image (Figure 13(d)) cannot be retrieved correctly, if data of Figure 13(b) are used as secret key. Therefore, the proposed encryption algorithm is robust against this kind of attack.

5.5. Occlusion Attack Analysis. In an occlusion attack, the transmitted encrypted image could lose blocks of information and not all the cryptograms can arrive to the receptor correctly. In this section, we present the robustness of the proposed encryption algorithm against 12.5%, 25%, and 50% of occlusion in an encrypted image. In Figure 14, the

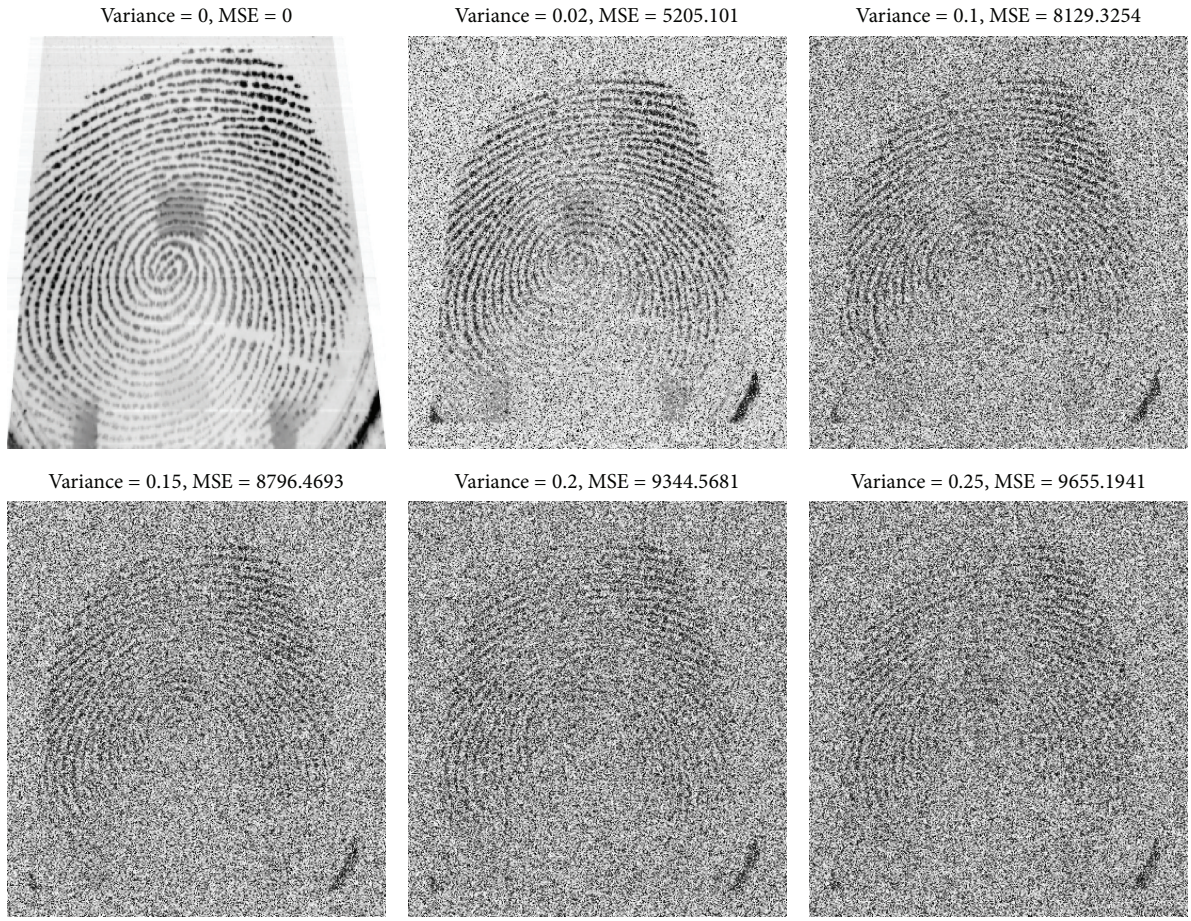


Figure 15: Decrypted image from cryptograms with Gaussian noise added.

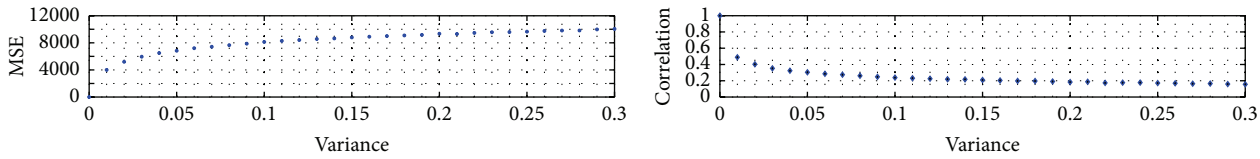


Figure 16: MSE and correlation curves between original and retrieved image in noise attack.

decryption image against three cases of occlusion is showed. The corresponding MSE values are 1844.9842 for 12.5%, 3581.8797 for 25%, and 7101.4179 for 50%. Since the decrypted image can be retrieved, the proposed encryption algorithm can resist an occlusion attack.

5.6. Noise Attack Analysis. This analysis shows the effectiveness of the proposed scheme against noise attack. In contrast to occlusion attack, the encrypted images can lose small portions of data over the encrypted image. In the analysis, encrypted data are distorted by zero-mean white additive Gaussian noise with a standard deviation from 0 to 0.3 with increments of 0.01. Figure 15 shows several decrypted images from encrypted image with added noise. In addition, Figure 16 presents the MSE and correlation curves between original image and recovered image. According to the results, the proposed scheme is robust against noise attack.

5.7. Encryption Time Analysis. The encryption and decryption processes are performed on a Laptop Toshiba Satellite E105-S1402 with operating system Windows Vista, processor speed of 2.26 Ghz, and 4 GB DDR2. The simulation is implemented in MATLAB R2008a software. ".bmp" plain image encryption of greyscale with 355×390 pixels (139,918 bytes) requires just 0.2340 seconds. The decryption process requires just 0.2105 seconds. Therefore, the proposed encryption algorithm can be implemented in real-time applications.

Table 2: Comparison of space key.

	Proposed	Reference [14]	Reference [11]	Reference [24]	Reference [25]	Reference [26]
Space key	10^{150}	10^{154}	10^{108}	10^{78}	10^{140}	10^{84}

Table 3: Speed of encryption/decryption.

	Speed of encryption	Speed of decryption	Units
Proposed	0.2340	0.2105	Seconds
Reference [14]	0.52	—	Seconds
Reference [26]	1.5	2	Seconds

6. Comparison with Similar Schemes in Literature

In this section, we present an important comparison with recent schemes reported in literature to show the effectiveness of the proposed scheme. The histogram generated by our scheme presents a uniform distribution, due to the highly uniform distribution of the values of the hyperchaotic sequences that benefit the encryption process at statistical level.

In Table 2, we present a space keys comparison with recent encryption schemes. In our scheme, we use just one hyperchaotic system to obtain a big space key. In [14], the authors use two 1D chaotic systems and two 3D chaotic systems to present a similar space key.

Table 3 presents a comparison of the speed encryption. The proposed scheme is faster than others where hyperchaotic or many chaotic systems are used to encrypt images.

7. Conclusions

In this paper, we present a robust and fast fingerprint image encryption algorithm scheme by using a hyperchaotic map. The security analysis verifies the security capabilities of the proposed scheme to be used in real applications and enforce the security of the biometric systems. The encryption process presents high security when the permutation stage is omitted. However, the correlation is lower when the proposed permutation presses is applied, which benefits the strength of the encryption scheme against statistical attacks.

Competing Interests

The authors declare that they have no competing interests.

Acknowledgments

This work was supported by the CONACYT, México, under Research Grant 166654.

References

[1] M. A. Murillo-Escobar, C. Cruz-Hernández, F. Abundiz-Pérez, R. M. López-Gutiérrez, and O. R. Acosta Del Campo, "A RGB image encryption algorithm based on total plain image characteristics and chaos," *Signal Processing*, vol. 109, pp. 119–131, 2015.

[2] M. A. Murillo-Escobar, C. Cruz-Hernández, F. Abundiz-Pérez, and R. M. López-Gutiérrez, "A robust embedded biometric authentication system based on fingerprint and chaotic encryption," *Expert Systems with Applications*, vol. 42, no. 21, pp. 8198–8211, 2015.

[3] A. Aguilar-Bustos, C. Cruz-Hernández, R. López-Gutiérrez, E. Tlelo-Cuautle, and C. Posadas-Castillo, "Hyperchaotic encryption for secure e-mail communication," in *Emergent Web Intelligence: Advanced Information Retrieval*, pp. 471–486, Springer, London, UK, 2010.

[4] C. Cruz-Hernández, R. López-Gutiérrez, A. Aguilar-Bustos, and C. Posadas-Castillo, "Communicating encrypted information based on synchronized hyperchaotic maps," *International Journal of Nonlinear Sciences and Numerical Simulation*, vol. 11, no. 5, pp. 337–350, 2010.

[5] A. Y. Aguilar-Bustos, C. Cruz-Hernández, R. M. López-Gutiérrez, and C. Posadas-Castillo, "Synchronization of different hyperchaotic maps for encryption," *Nonlinear Dynamics and Systems Theory*, vol. 8, no. 3, pp. 221–236, 2008.

[6] A. Y. Aguilar-Bustos and C. Cruz-Hernández, "Synchronization of discrete-time hyperchaotic systems: an application in communications," *Chaos, Solitons and Fractals*, vol. 41, no. 3, pp. 1301–1310, 2009.

[7] E. J. Ngamga, A. Buscarino, M. Frasca, G. Sciuto, J. Kurths, and L. Fortuna, "Recurrence-based detection of the hyperchaos-chaos transition in an electronic circuit," *Chaos*, vol. 20, no. 4, Article ID 043115, 2010.

[8] M. L. Barakat, A. S. Mansingka, A. G. Radwan, and K. N. Salama, "Generalized hardware post-processing technique for chaos-based pseudorandom number generators," *ETRI Journal*, vol. 35, no. 3, pp. 448–458, 2013.

[9] E. Inzunza-González and C. Cruz-Hernández, "Double hyperchaotic encryption for security in biometric systems," *Nonlinear Dynamics and Systems Theory*, vol. 13, no. 1, pp. 55–68, 2013.

[10] O. Farooq and S. Datta, "Signal-dependent chaotic-state-modulated digital secure communication," *ETRI Journal*, vol. 28, no. 2, pp. 250–252, 2006.

[11] G. Bhatnagar and Q. M. J. Wu, "Enhancing the transmission security of biometric images using chaotic encryption," *Multimedia Systems*, vol. 20, no. 2, pp. 203–214, 2014.

[12] H.-I. Hsiao and J. Lee, "A novel fingerprint image encryption algorithm based on chaos using APFM nonlinear adaptive filter," in *Proceedings of the IEEE 17th International Symposium on Consumer Electronics (ISCE '13)*, pp. 95–96, IEEE, Hsinchu, Taiwan, June 2013.

[13] R. Liu, "Chaos-based fingerprint images encryption using symmetric cryptography," in *Proceedings of the 9th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD '12)*, pp. 2153–2156, Sichuan, China, May 2012.

[14] H.-I. Hsiao and J. Lee, "Fingerprint image cryptography based on multiple chaotic systems," *Signal Processing*, vol. 113, pp. 169–181, 2015.

[15] F. Abundiz-Pérez, C. Cruz-Hernández, M. Murillo-Escobar, and R. López-Gutierrez, "Fingerprint image encryption based

on rossler map," in *Proceedings of the International Conference on Communications, Signal Processing and Computers*, pp. 193–197, Interlaken, Switzerland, February 2014.

[16] O. Mirzaei, M. Yaghoobi, and H. Irani, "A new image encryption method: parallel sub-image encryption with hyper chaos," *Nonlinear Dynamics*, vol. 67, no. 1, pp. 557–566, 2012.

[17] X.-Y. Wang and H.-l. Zhang, "A novel image encryption algorithm based on genetic recombination and hyper-chaotic systems," *Nonlinear Dynamics*, vol. 83, no. 1-2, pp. 333–346, 2016.

[18] X.-Y. Wang, Y.-Q. Zhang, and X.-M. Bao, "A novel chaotic image encryption scheme using DNA sequence operations," *Optics and Lasers in Engineering*, vol. 73, pp. 53–61, 2015.

[19] Q. Zhang, L. Guo, and X. Wei, "A novel image fusion encryption algorithm based on DNA sequence operation and hyper-chaotic system," *Optik*, vol. 124, no. 18, pp. 3596–3600, 2013.

[20] A. Jain, R. Bolle, and S. Pankanti, *Biometrics: Personal Identification in Networked Society*, vol. 479, Springer Science & Business Media, 2006.

[21] P. Ambalakat, "Security of biometric authentication systems," in *Proceedings of 21st Computer Science Seminar*, SA1-T1, pp. 1–7, Academic Press, 2005.

[22] D. Brooks, "Assessing vulnerabilities of biometric readers using an applied defeat evaluation methodology," in *Proceedings of the 3rd Australian Security and Intelligence Conference*, Edith Cowan University, November 2010.

[23] D. Maltoni, D. Maio, A. K. Jain, and S. Prabhakar, *Handbook of Fingerprint Recognition*, Springer Science & Business Media, 2009.

[24] M. K. Khan, J. Zhang, and K. Alghathbar, "Challenge-response-based biometric image scrambling for secure personal identification," *Future Generation Computer Systems*, vol. 27, no. 4, pp. 411–418, 2011.

[25] S. Zhao, H. Li, and X. Yan, "A secure and efficient fingerprint images encryption scheme," in *Proceedings of the 9th International Conference for Young Computer Scientists (ICYCS '08)*, pp. 2803–2808, Hunan, China, November 2008.

[26] G. Bhatnagar and Q. M. J. Wu, "Chaos-based security solution for fingerprint data during communication and transmission," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 4, pp. 876–887, 2012.

[27] K. Nandakumar, A. K. Jain, and A. Nagar, "Biometric template security," *EURASIP Journal on Advances in Signal Processing*, vol. 2008, Article ID 579416, 2008.

[28] M. K. Khan, L. Xie, and J. Zhang, "Chaos and NDFT-based spread spectrum concealing of fingerprint-biometric data into audio signals," *Digital Signal Processing*, vol. 20, no. 1, pp. 179–190, 2010.

[29] C. Militello, V. Conti, S. Vitabile, and F. Sorbello, "Embedded access points for trusted data and resources access in HPC systems," *The Journal of Supercomputing*, vol. 55, no. 1, pp. 4–27, 2011.

[30] A. H. B. Muñoz, "Ataques tipo side-channel a sistemas biométricos de reconocimiento de huella dactilar," Tech. Rep., Universidad Autónoma de Madrid, 2010.

[31] U. Uludag and A. K. Jain, "Attacks on biometric systems: a case study in fingerprints," in *Electronic Imaging*, vol. 5306 of *Proceedings of the SPIE*, pp. 622–633, 2004.

[32] G. Alvarez and S. Li, "Some basic cryptographic requirements for chaos-based cryptosystems," *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering*, vol. 16, no. 8, pp. 2129–2151, 2006.

[33] H. Li, Y. Wang, H. Yan, L. Li, Q. Li, and X. Zhao, "Double-image encryption by using chaos-based local pixel scrambling technique and gyrator transform," *Optics and Lasers in Engineering*, vol. 51, no. 12, pp. 1327–1331, 2013.