Journal of Information Systems Engineering and Management
2025, 10(62s)
e-ISSN: 2468-4376
https://www.jisem-journal.com/

Research Article

Augment the Security of Biometric Image Transmission Via the Implementation of Cascade Chaotic System Encryption.

Mohammed ARIF[1], Toufik GHRIB[2*], Yousef. A. Abusal[3]
[1]*Faculty of Applied Sciences, Université Kasdi Merbah, Ouargla, Algeria.*
[2]*École Normale Supérieure de Ouargla, Algeria.*
[3]*Ufa State Petroleum Technological University.*
* **Corresponding author:** GHRIB Toufik, ghrib.toufik@ens-ouargla.dz, https://orcid.org/0000-0001-7174-8962

ARTICLE INFO

Received: 29 Dec 2024

Revised: 12 Feb 2025

Accepted: 27 Feb 2025

ABSTRACT

The research investigates the various uses of chaotic maps by presenting a cascade chaotic system (CCS), which functions as a universal chaotic framework. The core idea entails utilizing an affine transformation with chaotically generated transformation sequences (the cascade chaotic system) to reorganize biometric pictures. The CCS can produce numerous new chaotic maps by employing two one-dimensional chaotic maps 1-D as seed maps. The newly created chaotic maps demonstrate elevated parameters, intricate chaotic traits, and improved chaotic efficacy, making them more unpredictable. We introduce a PRNG pseudo-random number generator and a data encryption system that utilize a chaotic map generated by the CCS to explore its possibilities. The effectiveness and reliability of the suggested method were confirmed by many simulations and analysis of biometric images.

Keywords: BIOmetrics security, Chaotic map, Cascade chaotic system , Data encryption, Pseudo-random number generator.

INTRODUCTION

Biometrics [1] denotes an automated system that utilizes measurable physical, physiological, or behavioral traits to identify or authenticate an individual's identification. In recent years, the research community has progressively advocated using biometrics in security applications [2].

Biometric security systems primarily employ recognition or authentication methods. In these systems, an individual's biometric data is generally contrasted with a pre-existing template to ascertain validity or authenticity. Nonetheless, biometrics are not infallible in guaranteeing security, as they are vulnerable to hacking, alteration, and exploitation during communication and transfer via insecure network channels.

Therefore, it is imperative to protect biometric data during these procedures. Two primary ways are typically utilized to safeguard biometric images: information concealing and encryption. Information concealment approaches seek to incorporate data within photographs to assert rightful ownership [3]. In contrast, encryption methods aim to conceal the message's substance, making it incomprehensible [4-5]. Information concealment is inappropriate for biometric security because it deteriorates image quality, thus impacting the precision of biometric authentication and consequently compromising the essential objective of biometrics.

Conversely, encryption preserves perceptual quality and is therefore very suitable for safeguarding biometric photos. The necessity to fulfill security standards has catalyzed the advancement of efficient encryption methods [6]. In recent years, other encryption algorithms have been introduced, including traditional approaches such as AES, RSA, and IDEA, primarily utilized for text or binary data. Nonetheless, current encryption methods are not optimally designed for biometric photos, chiefly because of the significant disparities between textual and biometric data. Recently, numerous chaotic maps have been created [7], which can be classified into two categories: 1) one-dimensional (1-D) and 2) high-dimensional (HD) chaotic mappings.

One-dimensional chaotic maps are mathematical systems that model the progression of a singular variable across discrete time intervals. Examples encompass logistic maps, tent maps, Gaussian maps, and dyadic transformations [7]. These one-dimensional chaotic maps are generally defined by their straightforward structures and ease of implementation. They demonstrate exceptional chaotic characteristics and are employed in numerous security applications [8]. Nevertheless, they also exhibit numerous security flaws.

1) Their chaotic ranges are constrained [8];

2) They possess a limited number of parameters;

3) Their outputs are readily predictable with minimal computational expense [9–11].

Conversely, high-dimensional (HD) chaotic maps represent the dynamics of a minimum of two variables. Prominent instances encompass the Hénon map [12], the Lorenz system [12], the Chen and Lee system [13], and the hyperchaotic system [14]. In contrast to 1-D chaotic maps, HD chaotic maps generally exhibit enhanced chaotic performance, rendering their chaotic orbits more difficult to predict [15].

Nonetheless, HD chaotic maps entail significant computational expenses and pose challenges for hardware implementation. These constraints hinder their efficacy in chaos-based applications, especially in real-time contexts. In light of the erratic performance of 1-D chaotic maps and the implementation challenges associated with high-dimensional chaotic maps, this study introduces a cascade chaotic system (CCS) as a comprehensive 1-D chaotic maps chaotic framework. CCS links two 1-D chaotic maps chaotic maps (seed maps) sequentially. The output of the initial seed map is connected to that of the subsequent seed map.

The output of the second component is reintegrated into the input of the first component for recursive iterations, and it also constitutes the CCS output. CCS, as a universal cascade architecture, can generate new chaotic maps (NCMs) utilizing any two 1-D chaotic maps as seed maps [16].

BACKGROUND

This section presents the notion of CCS, inspired by the cascade structures observed in electronic circuits. This method enables the creation of new 1-D chaotic maps by combining two established one-dimensional chaotic maps.

A. CCS

Fig. 1 illustrates the configuration of CCS, whereby G(x) and F(x) represent two seed maps. CCS serially connects two seed maps. The result of G(x) is entered into F(x), which is subsequently fed back into G(x) for recursive iterations [16].

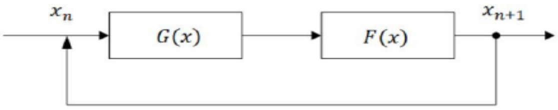

Fig. 1. Structure of CCS.

The suggested CCS can be expressed mathematically as follows, with G(x) and F(x) representing two seed maps.

$$\boldsymbol{x_{n+1}} = \boldsymbol{\Gamma}(\boldsymbol{x_n}) = \boldsymbol{F}(\boldsymbol{G}(\boldsymbol{x_n})). \qquad (1)$$

Any established one-dimensional 1-D chaotic map may serve as a seed map for CCS. Users can configure the seed maps F(x) and G(x) to be either identical or distinct chaotic maps.

1) When F(x) and G(x) are identical one-dimensional chaotic maps, specifically F(x) = G(x), the CCS in (1) is modified accordingly.

$$\boldsymbol{x_{n+1}} = \boldsymbol{F}(\boldsymbol{F}(\boldsymbol{x_n}))\text{ou}\boldsymbol{x_{n+1}} = \boldsymbol{G}(\boldsymbol{G}(\boldsymbol{x_n})). \qquad (2)$$

CCS is a structure in which a 1-D chaotic map is iteratively applied to itself. For instance, when F(x) and G(x) are sine functions, CCS represents the Double-Sine function.

2) When F(x) and G(x) are chosen as distinct chaotic maps, specifically F(x) ≠ G(x), CCS constitutes an alternative one-dimensional chaotic system characterized by (1) or by

$$\boldsymbol{x_{n+1} = G(F(x_n))}. \qquad (3)$$

By modifying the parameters of F(x) and G(x), or the order of the two seed maps, the Chaotic Cryptographic System (CCS) generates a unique one-dimensional chaotic map. The Tent-Logistic and Logistic-Tent maps are fundamentally distinct. The CCS offers users considerable freedom in creating numerous new chaotic maps (NCMs) by adjusting the parameters of F(x) and G(x). Compared to their corresponding seed maps, the chaotic maps produced by the CCS display unique properties, featuring additional parameters and demonstrating more complex chaotic behaviors.

B. Chaotic Behavior Analysis

By serially linking two chaotic maps, G(x) and F(x), the output sequences of the CCS can be organized as G(x), F(x), or a combination thereof. The CCS, as defined in (1), includes all parameters of its seed maps, therefore exhibiting a greater number of parameters and more intricate behaviors than the seed maps [16].

Assume that x and y are two proximate initial CCS values in Eq.(1). Following the initial iteration, the difference |x1 – y1| is delineated by

$$|\boldsymbol{x_1 - y_1}| = |\boldsymbol{\Gamma(x_0) - \Gamma(y_0)}|$$
$$= \frac{|F(G(x_0)) - F(G(y_0))|}{|G(x_0) - G(y_0)|} \frac{|G(x_0) - G(y_0)|}{|x_0 - y_0|} |\boldsymbol{x_0 - y_0}| \qquad (4)$$

After the n^{th} (n→∞) iteration, the difference between x_n and y_n is defined by

$$|\boldsymbol{x_n - y_n}| = |\boldsymbol{\Gamma(x_{n-1}) - \Gamma(y_{n-1})}|$$
$$\approx |\prod_{i=0}^{n-1} \cdot \frac{dF}{dx}|_{G(x_i)}||\prod_{i=0}^{n-1} \cdot \frac{dG}{dx}|_{x_i}||\boldsymbol{x_0 - y_0}| \qquad (5)$$

Then the average change in each iteration from |x –y| to |xn – yn| is

$$\boldsymbol{\Delta_{\Gamma(x)}} \approx \{|\prod_{i=0}^{n-1} \cdot \frac{dF}{dx}|_{G(x_i)}||\prod_{i=0}^{n-1} \cdot \frac{dG}{dx}|_{x_i}|\}^{\frac{1}{n}} \qquad (6)$$

Therefore, LE of $\boldsymbol{\Gamma(x)}$ is defined by

$$\boldsymbol{\lambda_{\Gamma(x)} = \ln(\Delta_{\Gamma(x)})}$$
$$= \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \cdot \ln |\frac{dF}{dx}|_{G(x_i)}| + \lim_{n\to\infty} \frac{1}{n} \sum_{i=0}^{n-1} \cdot \ln |\frac{dG}{dx}|_{x_i}|$$
$$\boldsymbol{\lambda_{\Gamma(x)} = \lambda_{F(x)} + \lambda_{G(x)}}. \qquad (7)$$

Thus, the Lyapunov Exponent (LE) of the Chaotic Cryptographic System (CCS) is obtained from the LE values of the two seed maps. When > 0, the trajectories of the two output sequences of CCS demonstrate considerable divergence as the number of repetitions escalates, resulting in chaotic behavior. A greater positive LE value indicates a more rapid divergence of the two paths, hence amplifying the chaotic behavior. The disordered attributes of CCS can be encapsulated as follows:

1) When λ_(F(x)) > 0 and λ_(G(x)) > 0, λ_(Γ(x)) > 0, and λ_(Γ(x)) > λ_(F(x)) as well as λ_(Γ(x)) > λ_(G(x)). When both seed maps exhibit chaotic behavior, CCS also displays chaos and demonstrates superior chaotic performance compared to its seed maps.
2) When λG(x) ≤ 0 and λF(x) ≤ 0, then λ_(Θ(x)) ≤ 0. CCS exhibits no chaotic behavior when neither seed map is chaotic.
3) When λG(x) > 0 and λF(x) ≤ 0, or λF(x) > 0 and λG(x) ≤ 0, it follows that

$$\lambda_{\Gamma(x)} \begin{cases} > 0 \textbf{ if } \boldsymbol{\lambda_{F(x)} + \lambda_{G(x)}} > 0 \\ \boldsymbol{\leq 0} \textbf{ if } \boldsymbol{\lambda_{F(x)} + \lambda_{G(x)} \leq 0} \end{cases} \qquad (8)$$

CCS will be chaotic if and only if λF(x) + λG(x) > 0, provided that only one seed map is chaotic.

CCS is often chaotic when at least one seed map falls inside the chaotic range. It exhibits superior chaotic performance when both seed maps are chaotic.

PROPOSED DATA ENCRYPTION SYSTEM, RESULTS AND DISCUSSION

Data encryption, a crucial technique in data security, has received much attention for its capacity to transform data into an incomprehensible format. In recent decades, a multitude of data encryption technologies has emerged, including the Digital Encryption Standard (DES), Advanced Encryption Standard (AES), networked data encryption, and various additional encryption algorithms. Chaotic maps, distinguished by their sensitivity to parameters and beginning circumstances, ergodicity, and unpredictability, serve as highly effective instruments for data encryption. These maps, demonstrating enhanced chaotic activity, provide significant security benefits for data encryption. The suggested CCS exhibits commendable chaotic performance and is appropriate for data encryption.

This section introduces an innovative data encryption technique, the Tent-Logistic map (TL-DEA), which use the Tent-Logistic map as a model of chaotic systems (CCS). Numerous existing data encryption algorithms (DEAs), including DES and AES, are designed to encrypt data in binary formats; hence, data in different formats must be transformed to binary before encryption. The conversion procedure may be inefficient for extensive collections, such as high-resolution photos and movies. Conversely, TL-DEA may directly encrypt several data types. Simulations and security assessments have been performed to assess the encryption efficacy [16].

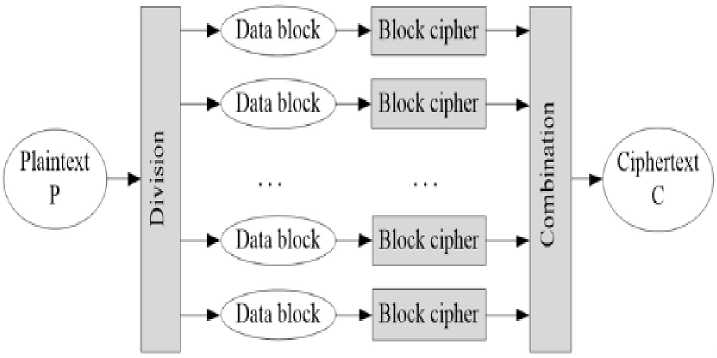

Fig. 2. Proposed TL-DEA.

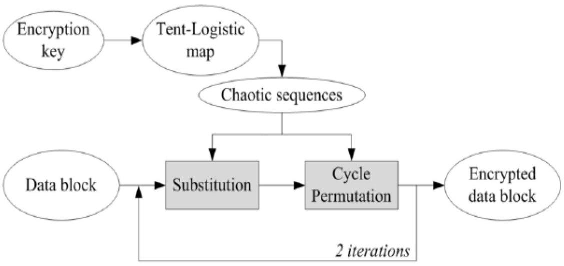

Fig. 3. Block diagram of the block cipher.

A. TL-DEA

Fig. 2 illustrates a block diagram of the planned TL-DEA. The plaintext, referred to as P, denotes the original data, whereas the ciphertext, indicated as C, means the encrypted data. The division operation partitions the plaintext into many data blocks of uniform length. Thereafter, a block cipher is utilized to encrypt each data block separately. The combination procedure subsequently amalgamates all encrypted data blocks into a singular encrypted data sequence, yielding the ciphertext.

Fig. 3 illustrates the block cipher. The encryption key establishes the basic parameters for the Tent-Logistic map. The TL-DEA employs two iterations of substitution and permutation operations to attain efficient confusion and diffusion. Principal Examination: The security key in the TL-DEA architecture has a length of 256 bits. This key is employed to produce two sets of initial values and parameters, as specified in Algorithm 1. Thereafter, the Tent-Logistic map utilizes these values to generate two separate chaotic sequences [16].

```
Algorithm 1 Creation of Initial Variables and Parameters
Input : Security key $K$ with length of 256 bits
Initial value $x_0 \leftarrow \left(\sum_{i=1}^{52} K_i \, 2^{52-i}\right)/2^{52}$
parameter $u \leftarrow \left(\sum_{i=53}^{104} K_i \, 2^{104-i}\right)/2^{52}$
parameter $a \leftarrow \left(\sum_{i=105}^{156} K_i \, 2^{156-i}\right)/2^{52}$
$T \leftarrow \left(\sum_{i=157}^{208} K_i \, 2^{208-i}\right)/2^{52}$
$R_1 \leftarrow \sum_{i=209}^{232} K_i \, 2^{232-i}$
$R_2 \leftarrow \sum_{i=233}^{256} K_i \, 2^{256-i}$
for $i = 1$ to 2 do
    $x_{0i} \leftarrow (x_0 + R_i T) mod\ 1$
    $u_i \leftarrow 1.8 + (u + R_i T) mod\ 0.2$
    $a_i \leftarrow 3.8 + (a + R_i T) mod\ 0.2$
end for
Output : Initial conditions $(x_{01}, u_1, a_1)$ and $(x_{02}, u_2, a_2)$.
```

1) The replacement technique aims to alter the data values in the plaintext by employing two adjacent data values along with a random value obtained from a chaotic sequence. Examine a data block \(P \) of length \(L \) and a chaotic sequence \(S \) of identical length \(L \), produced by the Tent-Logistic map, represented as \(S = \{x_1, x_2, \ldots, x_L\} \). By associating each datum with its predecessor and connecting the initial datum to the final one, the data block is reconfigured into a circular shape. The substitution procedure for each data block is further delineated as follows

$$H_i = \begin{cases} (P_i + P_L + P_{L-1} + \lfloor S \times 2^{20} \rfloor_i) \ mod\ F & \text{if } i = 1 \\ (P_i + C_{i-1} + P_L + \lfloor S \times 2^{20} \rfloor_i) \ mod\ F & \text{if } i = 2 \\ (P_i + C_{i-1} + C_{i-2} + \lfloor S \times 2^{20} \rfloor_i) \ mod\ F & \text{if } i \in [3, L] \end{cases} \qquad (9)$$

In this instance, F represents the quantity of allowable intensity scales within the plaintext. For example, F equals 2 when the plaintext consists exclusively of binary data, and F equals 256 when the plaintext is represented as 8-bit decimals. The floor procedure is executed as specified in [16].

```
Algorithm 2 Cycle Permutation
Input : Data block $H$ and chaotic sequence $S$. Both are with length of $L$
Rearrange $H, S$ with size of $M \times N$, where $L = M \times N$
Sort each row of $S$ and get the row index matrix $I$. Then
Sorted_$S_{m,n} = S_m, I_{m,n}$, where $m, n \in [1, M] \times [1, N]$
for $j = 1$ to $N$do
    for $i = 1$ to $M$do
```

```
        Find value $\boldsymbol{j}$ in $\boldsymbol{i}$th row of$\boldsymbol{I}$, get its position $(\boldsymbol{i},\ \boldsymbol{j_i})$.
    end for
    Connect values of $\boldsymbol{H}$ in positions $(\boldsymbol{1}, \boldsymbol{j_1})$, $(\boldsymbol{2}, \boldsymbol{j_2})$, ..., $(\boldsymbol{M}, \boldsymbol{j_M})$
    into a circle, and shift them by $\boldsymbol{j}$ positions to upper direction.
end for
Rearrange the permutation result into length of $\boldsymbol{L}$
Output :The permuted result $\boldsymbol{C}$.
```

2) Cycle permutation entails the reorganization of all data positions, as demonstrated in Algorithm 2. For example, consider the row index matrix I as outlined below:

$$I = \begin{bmatrix} 2 & 1 & 4 & 3 \\ 1 & 3 & 2 & 4 \\ 3 & 2 & 4 & 1 \\ 1 & 4 & 3 & 2 \end{bmatrix}$$

Fig. 4 depicts the specific processes employing the index matrix I. Initially, the index value 1 is present in all rows of I, resulting in the positions (1, 2), (2, 1), (3, 4), and (4, 1). The data associated with these points in data block H are then arranged in a circular configuration and relocated one position upward. Consequently, the index value 2 is located in I, yielding the coordinates (1, 1), (2, 3), (3, 2), and (4, 4). The data at these locations in H are likewise arranged in a circular manner and displaced two positions higher. This technique is reiterated until the maximum index value in I is attained. Upon concluding a permutation cycle, the data can be efficiently segregated from all contiguous data.

By executing the replacement and circular permutation once more using an alternative chaotic sequence, an encrypted data block is generated.

B. *Simulation Results*

An effective cryptographic method must be able to convert different types of plaintext into ciphertext that seems as random noise. In our experimental research, we employed binary data and 8-bit decimal data, including graphics, as the plaintext to assess the encryption effectiveness of the proposed TL-DEA. The simulations were executed utilizing MATLAB R2015a on the Windows 10 Pro operating system.

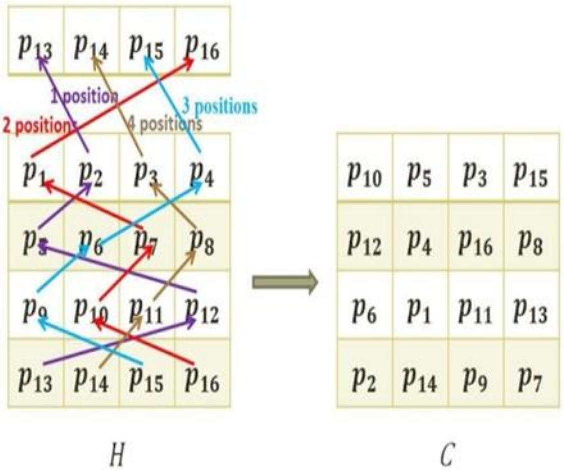

Fig. 4. Diagram of cycle permutation

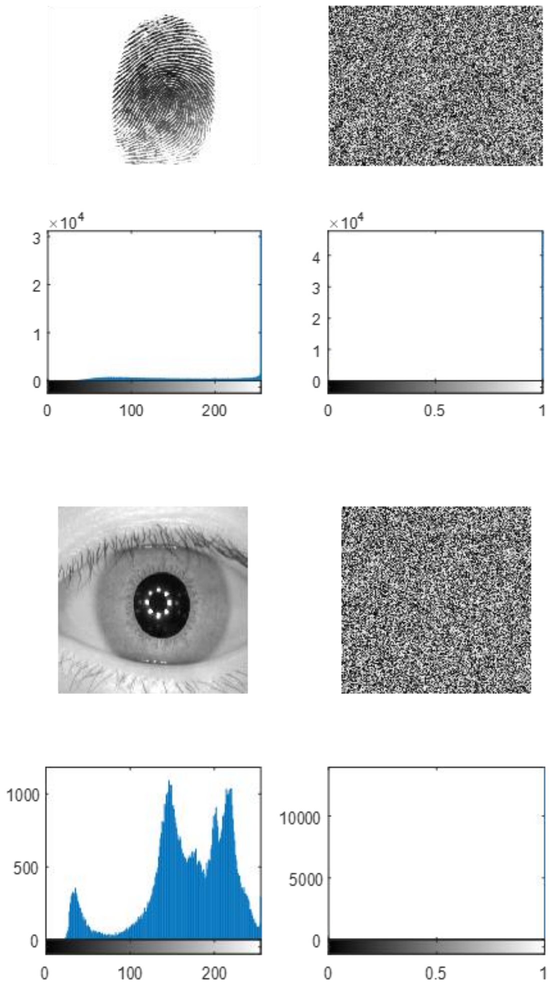

Fig. 5. Encryption results of fingerprint and iris images

To encrypt the binary data, we employed a binary image as a representative example. A binary image, represented as a two-dimensional matrix, can be considered a data block and immediately processed using a block cipher. The results of the encryption are depicted in Fig. 5. The binary data, denoted by 0 and 1 in the ciphertext, are evidently randomly dispersed over all places. As a result, no information regarding the original data is identifiable. TL-DEA may also encrypt data in other formats, including digital photos and films, where pixels are generally represented by eight or more bits. This approach facilitates direct pixel-level encryption, demonstrating greater efficiency and convenience than bit-level encryption.

C. Security Analysis

Security is the foremost characteristic of cryptographic systems. A robust cryptographic system must exhibit resistance to recognized attacks. To assess the security efficacy of the proposed TL-DEA, we performed a security analysis utilizing digital images encoded in eight bits. This investigation included essential sensitivity tests, differential attack evaluations, and assessments of noise and data loss assaults.

1) The Key Sensitivity Test is an essential assessment of an encryption algorithm's sensitivity to security keys. This assessment includes two fundamental components: 1) Encryption key sensitivity, signifying that even a trivial modification in the encryption keys produces an entirely distinct ciphertext, and 2) decryption key sensitivity, which guarantees that the original plaintext can solely be recovered using the appropriate security keys, with any minor alteration in the keys resulting in an unrecognizable decryption result. Figure 6 illustrates the outcomes of the critical sensitivity analysis. Security keys K2 and K3 are generated from the original security key K1, differing by one bit. As demonstrated, when a plaintext image P [Fig. 6(a)] is encrypted using K2 and K3, differing by a single bit, the resultant encrypted outputs are completely unlike, as depicted in Fig. 6(b) and (c). Figure 6(d) emphasizes these distinctions. In contrast, when a ciphertext picture [Fig. 6(b)] is decoded with two security keys that differ by one bit, the resulting decrypted outputs are entirely distinct, as illustrated in Fig. 6(f) and (g). Only the appropriate security key can precisely reconstruct the original plaintext, as illustrated in Fig. 6(e). Thus, the proposed TL-DEA demonstrates considerable sensitivity to security keys in both encryption and decryption procedures.

2) Examination of Differential Attacks: A cryptographic system demonstrating strong diffusion characteristics can endure differential attacks. To quantitatively evaluate the diffusion characteristics of TL-DEA, we utilized the pixel change rate (NPCR) and the unified average changed intensity (UACI) [16]. The NPCR and UACI for two pictures, C1 and C2, are mathematically described as follows:

$$
\begin{gathered}
\mathbf{NPCR}\ (C_1, C_2) = \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{D(i,j)}{L} \times \mathbf{100\%} \\
\mathbf{UAC\,I}(C_1, C_2) = \sum_{i=1}^{M} \sum_{j=1}^{N} \frac{|C_1(i,j) - C_2(i,j)|}{T \times L} \times \mathbf{100\%} \\
D(i,j) = \begin{cases} \mathbf{0}, & \text{if } C_1(i,j) = C_2(i,j) \\ \mathbf{1}, & \text{if } C_1(i,j) \neq C_2(i,j) \end{cases}
\end{gathered}
\tag{10}
$$

From the perspective of this discussion, C1 and C2 are two encrypted images that are formed from two plaintext images that are distinct from one another by a single pixel. In this context, the letter T represents the maximum pixel intensity that is allowed, and the letter L shows the total number of pixels that are contained within the image. The Number of Pixels Change Rate (NPCR) examines the proportion of pixels that are different between the two encrypted images, whilst the Unified Average Changing Intensity (UACI) evaluates the differences in pixel intensities. Both of these metrics are used to compare the two encrypted images.

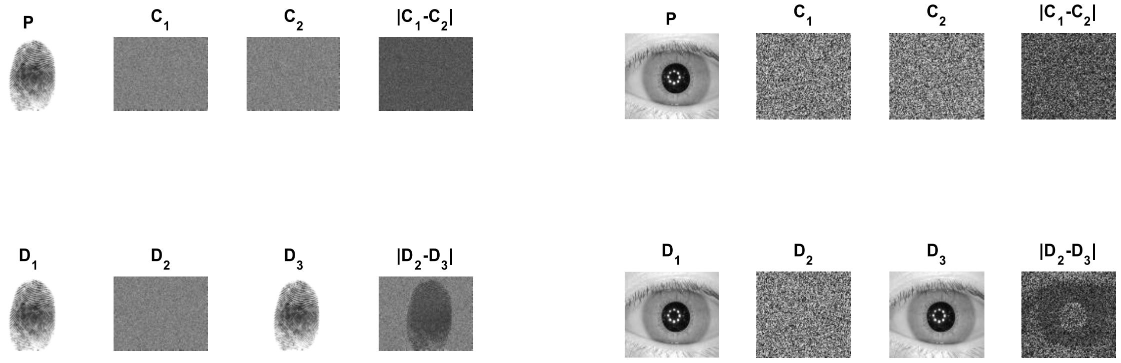

Fig. 6. Key sensitivity analysis.

Plaintext image P. Ciphertext image C1 utilizing key K1. Ciphertext image C2 utilizing key K2. Difference between ciphertext images, |C1 - C2|. Decrypted image D1 from C1 utilizing key K1. Decrypted image D2 from C1 utilizing key K2. Decrypted image D3 from C1 utilizing key K3. Difference between decrypted images, |D2 – D3|.

TABLE I. NPCR and UACI findings from TL-DEA utilizing the plaintext images

File name	NPCR %	UACI %
empriente1.tif	99.5925	33.5221
empriente2.tif	99.5732	33.4574
empriente3.tif	99.6536	33.3889

empriente4.tif	99.5983	33.4354
empriente5.tif	99.6116	33.4733
iris1.jpg	99.6104	33.3876
iris2.jpg	99.6216	33.5481
iris3.jpg	99.6004	33.3258
iris4.jpg	99.6093	33.3781
iris5.jpg	99.5970	33.4016
Moyenne	**99.6068**	**33.4318**

To construct a changed test image, the experimental process involved setting a single pixel in each test image to zero. This was done in order to see the results. Following that, TL-DEA was applied to both the original and changed photos, making use of the same security key throughout the process. The NPCR and UACI measurements were then utilized in order to conduct an analysis on the encrypted outputs. These measures are presented in Table I, which contains the results. It was discovered that the average values of NPCR and UACI were 99.6068% and 33.4318%, respectively. These values are surprisingly near to the theoretical ideal values of NPCR and UACI, which are 99.609% and 33.464%, respectively [32]. Based on this, it can be concluded that TL-DEA possesses exceptional diffusion qualities and is able to tolerate differential attacks.

CONCLUSION

Regarding the secure transformation of biometric photos, this study studies the enhancement of a novel Chaotic Cryptographic System (CCS) for the purpose of enhancing its security. Based on the evaluation and comparative analysis, it has been determined that the newly developed chaotic maps display a higher degree of unpredictability and superior chaotic performance. Additionally, in comparison to the chaotic maps that are already in existence, the newly developed chaotic maps have a greater number of parameters and more intricate chaotic properties.

We use the Tent-Logistic map as an example of New Chaotic Maps (NCMs) within the CCS framework. We also introduce the Tent-Logistic Pseudo-Random Number Generator (TLPRNG) and the Tent-Logistic Data Encryption Algorithm (TL-DEA) in order to demonstrate the potential advantages that the proposed CCS could bring to chaos-based applications. In addition, we evaluate the effectiveness of TL-DEA with regard to the encryption of data and the study of security. It has been demonstrated through these findings that TL-DEA is able to provide a high level of security for a variety of data formats, effectively defending against differential assaults, as well as noise and data loss threats.

REFRENCES

[1] Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: Handbook of fingerprint recognition. Springer, Berlin (2003)

[2] Uludag, U., Pankanti, S., Prabhakar, S. , Biometric cryptosystems: issues and challenges. Proc. IEEE 92, 948–960 (2004)

[3] Li, C., Wang, Y., Ma, B., Zhang, Z.: Multi-block dependency based fragile watermarking scheme for fingerprint images protection. Multimed. Tools Appl., 1–20 (2012)

[4] Li, P., Yang, X., Qiao, H., Cao, K., Liu, E., Tian, J.: An effective biometric cryptosystem combining fingerprints with error correction codes. Exp. Syst. Appl. 39(7), 6562–6574 (2012)

[5] Lee, H., Teoh, A.B.J., Jung, H.G., Kim, J.: A secure biometric discretization scheme for face template protection. Future Gener. Computer Systems 28(1), 218–231 (2012)

[6] Gaurav Bhatnagar and Q. M. Jonathan Wu: Enhancing the transmission security of biometric images using chaotic encryption. Multimedia Systems. Springer-Verlag Berlin Heidelberg (2013)

[7] R. C. Hilborn, Chaos , and Nonlinear Dynamics: An Introduction for Scientists and Engineers, 2nd ed. Oxford University, New York, NY, USA Press, 2001.

[8] Y. Zhou, L. Bao, and C. L. P. Chen, “A new 1D chaotic system for image encryption,” Signal Process., vol. 97, pp. 172–182, Apr. 2014.

[9] D. Arroyo, R. Rhouma, G. , Alvarez, S. , & Li, V. Fernandez, “On the security of a new image encryption scheme based on chaotic map lattices,” Chaos, vol. 18, no. 3, Sep. 2008, Art. ID 033112.

[10] H. C. Papadopoulos and G. W. Wornell, “Maximum-likelihood estimation of a class of chaotic signals,” IEEE Trans. Inf. Theory, vol. 41, no. 1, pp. 312–317, Jan. 1995.

[11] X. Wu, H. Hu, and B. Zhang, “Parameter estimation only from the symbolic sequences generated by chaos system,” Chaos Soliton. Fract., vol. 22, no. 2, pp. 359–366, Oct. 2004.

[12] G. Chen and X. Yu, Chaos Control: Theory and Applications, vol. 292. Berlin, Germany: Springer 2003.

[13] H.-K. Chen and C.-I. Lee, “Anti-control of chaos in rigid body motion,” Chaos Soliton. Fract., vol. 21, no. 4, pp. 957-965, 2004.

[14] C. Shen, S. Yu, J. Lu, and G. Chen, “A systematic methodology for constructing hyperchaotic systems with multiple positive Lyapunov exponents and circuit implementation,” IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 61, no. 3, pp. 854–864, Mar. 2014.

[15] T. Gao and Z. Chen, “A new image encryption algorithm based on hyper-chaos,” Phys. Lett. A, vol. 372, no. 4, pp. 394–400, Jan. 2008.

[16] Yicong Zhou, Zhongyun Hua, Chi-Man Pun, C. L. Philip Chen, : Cascade Chaotic System With Applications, IEEE TRANSACTIONS ON CYBERNETICS, 2014

[17] J. Lu and G. Chen, “A time-varying complex dynamical network model and its controlled synchronization criteria,” IEEE Trans. Autom. Control, vol. 50, no. 6, pp. 841–846, Jun. 2005.

[18] Y. Zhou, K. Panetta, S. Agaian, and C. L. P. Chen, “(n, k, p)-Gray code for image systems,” IEEE Trans. Cybern., vol. 43, no. 2, pp. 515–529, Apr. 2013.