International Journal of Advanced Science and Computer Applications (2024); 3(1): 15-30
DOI: 10.47679/ijasca.v3i1.42
https://ijasca.org/index.php/ijasca

RESEARCH ARTICLE

big S-Boxes for Deeply Improved Hill Image Coding

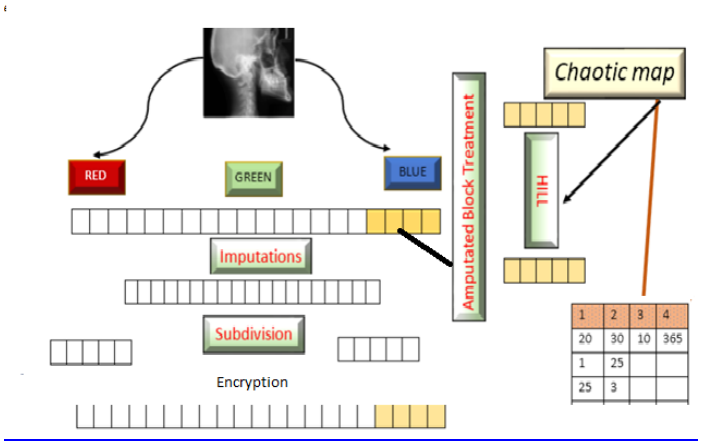
Mariem Jarjar [1]Abid Abdellah[1], Hicham Rrghout [1], Mourad Kattass[1], Abdellatif JarJar [1] and Abdellhamid Benazzi [1]

Published online: 20 November 2023

Abstract

This paper introduces a novel approach to encrypt color images, utilizing multiple chaotic maps and working with any size pixel blocks. The encryption procedure commences by converting the original image into a vector, which is then divided into equal dimension subblocks. These blocks undergo a new Vigenère transformation, employing two large dynamic substitution tables interconnected by an enhanced Hill operation using a large invertible matrix generated from a pseudo-random table, controlled by another binary dynamics table. Additionally, the truncated block undergoes the same encryption process using another pseudo-random invertible matrix. To evaluate the effectiveness and resilience of our algorithm, we conducted numerous simulations on randomly selected images of varying sizes and formats from diverse databases.

Graphical Abstract



Keyword: Chaotic map; Broadcast function; S-Box; large invertible matrices

Notation	1.	⊕Xor operator
	2.	E(x): The whole part of the real number x
	3.	(n, m) original imagei sze
	4.	Residue(n): Sum of integer digits (n)

[1] Mohamed First University, MATSI Laboratory, Oujda, Morocco

*) corresponding author

Mariem Jarjar

Email: jarjarmariem@gmail.com

Article Highlights

The strengths of our method are:

- Large S-Boxes incorporate confusion and diffusion functions.
- Original image Subdivision into arbitrary size sub-blocks.
- Large invertible matrices design.
- Improvement Hill transformation.

Received 10 November 2023 | Received in revised form 18 January 2024 | Accepted 22 February 2024
Available online 24 March 2024

Introduction

The rapid development of chaos theory in mathematics provides researchers with opportunities to further improve some classic encryption systems. In front of this great security focus, many techniques for color image encryption have flooded the digital world, mostly exploiting number theory and chaos [1-2]. Others are attempting to update their policies by improving some classical techniques, such as Hill [3-4], Cesar, Vignere [5-6], Feistel [7-8]. Unfortunately, the lack of any encryption modus operandi and chaining between cipher blocks and clear blocks makes most of these enhancements vulnerable to differential attacks.

Earlier works

Due to its essential trait of being sensitive to beginning circumstances and producing sequences of pseudo-random numbers, chaos has recently attracted renewed interest in the field of cryptography. The chaotic system is ideally suited for image encryption thanks to all these characteristics. As a result, chaos-based image cryptographic protocols are already exploding the web. The majority of these algorithms follow Shannon's rule. In recent decades, several image encryption methods have been proposed in literature, and improved, namely:

The article [9] introduces a color image encryption technique that marries the logistic map and sine map, two chaotic maps, to develop a new, more successful map. The four most significant bits of each channel's pixels are concatenated and then moved in a circular pattern, horizontally and vertically, even before pixels of each channel are scrambled at the pixel level. This is done by swapping rows and columns between different channels in conformity only with random sequences produced by the newly map. In the diffusion process, the control sequences-controlled diffusion sequences are utilized to disperse the muddled pixels of each channel.

The paper's authors [10] describe a novel hybrid color image encryption scheme that blends the Latin square and chaotic systems into a permutation-substitution network. We may introduce the beneficial properties of confusion and diffusion, as well as the tolerance to the integration of noise in the decryption, thanks to the similarity between the two systems.

The authors of the paper [11] have invented a novel color image encryption scheme based on a linear function guaranteed by an invertible multiplier and a 2D hyperchaotic map created by merging two 1D chaotic maps. This approach uses basic row and column shifts to jumble the image before installing a broadcast function to retrieve the encrypted image.

A novel color image encryption system based on various chaotic maps and a Vigenère strategy was described by Ritesh Bansal, et al. involves a single cycle of dissemination and bewilderment. Three phases make up the first stage: backward scatter, the Vigenère scheme's matching operation, and forward scatter. Later on, pixel coordinates are swapped adopting position swapping with a chaotic map.

Our contribution in this article involves two main steps. First, we convert the original image into a vector representation. Next, we divide the vector into blocks of size determined using pseudo-random vectors generated from various chaotic maps employed in the system. To address the issue of linearity, we apply a Hill transformation along with pseudo-random translation vectors and incorporate a new Vigenère technique. This technique is supported by large S-Boxes to ensure robustness

The proposed method

In this section, we will define the chaotic for construct all the pseudo-random tables to set up the confusion and diffusion process.

Step1: Chaotic Parameter Development

Based on chaos [9-10-11], This technique utilizes the most commonly used chaotic maps in the field of cryptography.

1) Chaotic Sequences Development

In this work, we used the three most popular chaotic maps in the field of cryptography.

a) The Logistics Map

The logistic map($\mathbf{u_n}$) $[12-13-14]$is a recurrent sequence described by a simple polynomial of second degree defined by the following equation

Logistics Map (u_n)	
$u_0 \in]0{,}5\ \ 1[\quad,\ \mu \in [3{,}75\ ;\ \ 4]$	(1)
$u_{n+1} = \mu u_n(1-u_n)$	

b) A.J Chaotic Map

This new ($\mathbf{W_n}$) A.J chaotic map[15] is defined by using a piecewise linear function. This chaotic sequence is given by the following formula

A. J Chaotic map

$$\begin{cases} w_0 \in \left[\frac{1}{(1+p)} \quad \frac{p}{(1+p)}\right] \qquad p \in [1{,}47 \;;\quad \varphi] \\ f(w_n) = w_{n+1} \begin{cases} p^2 w_n & \text{if } \; 0 \le w_n \le \frac{1}{1+p} \\ p - p w_n & \text{if } \; \frac{1}{1+p} \le w_n \le 1 \end{cases} \end{cases} \qquad (2)$$

c) The Skew Tent Map (SKTM)

The Skew tent [16 – 17]map ($\mathbf{v_n}$)will be redefined as the next equation

The Skew Tent Map (v_n)

$$\begin{cases} v_0 \in]0 \;\; 1[\qquad p \in]0{,}5 \;\; 1[\\ v_{n+1} = \begin{cases} \frac{v_n}{p} & \text{if } \; 0 \prec v_n \prec p \\ \frac{1 - v_n}{1 - p} & \text{if } p \prec v_{n<1} \end{cases} \end{cases} \qquad (3)$$

The combination of these three chaotic maps will be utilized to generate all the essential parameters required for the effective operation and our innovative technology utilization.

2) Pseudo-random table construction

Our work entails constructing a (CT) chaotic table with coefficients in (G_{256}) of size (3nm; 5), which is required for fusion and diffusion processes, and a (BT) binary table control of size (3 nm; 2). This setup can be seen in the diagram below

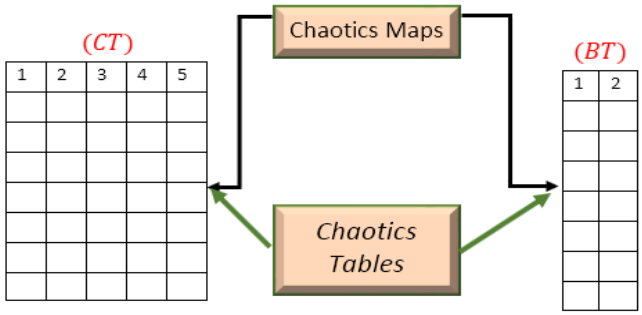


Figure2: Chaotics tables design

This figure is illustrated by the following algorithm:

Algorithm1: Chaotic used table design

1. For i = 1 to 3nm

(CT)**Confusion table in G_{256}**

2. $CT(i;1) = \mathrm{mod}(E(|u(i) - v(i)| * 10^{10},\ 253) + 3)$
3. $CT(i;2) = \mathrm{mod}(E(u(i) * w(i)) * 10^{10}, 253) + 2)$
4. $CT(i;3) = \mathrm{mod}(i) = \mathrm{mod}(E(u(i) * v(i) * 10^{11},\ 253) + 1$
5. $CT(i;4) = \mathrm{mod}(E((u(i) + w(i) + v(i)) * 10^{10},\ 253) + 1$
6. $CT(i;5) = \sup\,(CT(1;i); CT(4;i))$

(**BT**) **Control table in G_2**

7. If $u(i) \ge v(i)$ then $BT(i;1) = 0$ else $BT(i;1) = 1$: end if
8. If $u(i) > \mathrm{Sup}(v(i)\,; w(i))$Then $BT(i;2) = 0$ else $BT(i;2) = 1$
9. end if
10. Next i

Axe2: Getting the image ready for encryption:

In this section, we will describe all the necessary steps for preparing the original image before proceeding to the encryption process[18 – 19 – 20].

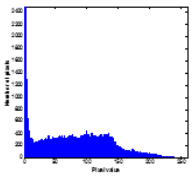
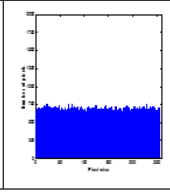
1) Original Image Vectoring

In order to minimize the correlation between adjacent pixels and maximize the entropy value, the original image is first obfuscated with a pseudo-random table (CT) under the control of a decision table (BT). After the three (RGB) color channels extraction and their conversion into size vectors (Or), (Og), (Ob) (1, nm) each, The transition to the vector $X(x_1, x_2, \ldots\ldots, x_{3nm})$ is described by the following algorithm.

Algorithm2: Switching to vector (X)
1. for i = 1 to nm
2. If BT(i; 2) = 0 Then
3. $X(3i-2) = Ob(i) \oplus Inf(CT(3i;1); CT(i;4))$
4. $X(3i-1) = Or(i) \oplus Sup(CT(i;2); CT(3i-1;3))$
5. $X(3i) = Og(i) \oplus sup\ (CT(3i-2;4); CT(3i;5))$
6. Else
7. $X(3i-2) = Or(i) \oplus CT(3i-1;4)$
8. $X(3i-1) = Ob(i) \oplus CT(3i;5)$
9. $X(3i) = Og(i) \oplus Inf(CT(3i;2); CT(i;3))$
10. End if
11. Next i

The decision vector (BT(:2)) governs the alteration of the vector's sign. This process effectively diminishes the high correlation between pixels and maximizes the entropy value, while ensuring robust encryption that withstands brute-force and statistical attacks. This initial phase can be viewed as a moderate original image encryption. The underlying mechanism is detailed in the following table

Table 1: Light encryption

Image	*Histogram*		*Entropie*	
	Originale	*Cypher*	*Original*	*Cypher*
			4,5687	7.9996

This transformation provides a slightly encrypted image, but protects the image from random and frequency attacks. A second round is required to increase the complexity of our method, preventing differential attacks.

2) Arbitrary sub-vectors Subdivision

Initially, the vector (X) will be cut into several identical size sub-blocks arbitrarily calculated from the chaotic maps used.

a) Subblocks Size Common

Let (r) the common encrypted sub-blocks size by our new process. The following algorithm specifies the value of this dimension in further detail.

Algorithm3: Common size	
1. r = 0; d = Sup(n; m)	5. Else
2. For i = 1 to nm	6. l = Mod(CT(i; 1) + CT(i; 4); m)
3. If BT(i; 2) = 0 Then	7. End if
4. l = Mod(CT(i; 2) + CT(i; 5); n)	8. r = l + r
	9. Next i:
	10. R= Mod(r; d) + Inf(n; m)
	11. r = residue (R)

Remark

$$\mathbf{4 \le r \le 9 * Log_{10}(Inf(n; m)}$$

We note

Image size adjustment	
1. $3nm \equiv q\ [r]$	(4)
2. $t = \frac{3nm-q}{r}$	
3. $0 \le q < r$	

- ✓ q: size of the box to be imputed if $q \neq 0$
- ✓ (r)Common block size.
- ✓ t: number of size blocks (r)

b) Image vector size adaptation

Initially, the vector (X) will be divided into two sub-vectors (X′) of dimension (1; rt) and one sub-block (W) of size (1; q). This operation is determined by the following algorithm.

Algorithm3: Image size adjustment
(X') construction
1. For i = 1 to rt
2. X'(i) = X(i)
3. Next i
(W') construction
4. For i = rt + 1 to 3nm
5. W(i) = X(i)
6. Next i

This situation is seen in the following figure.

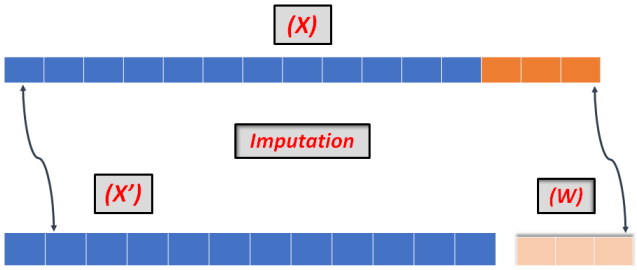


Figure2: Image vector size adaptation

In a second step, the vector (X′) is subdivided into (t) (Ui) sub-blocks of size (1; r) each. This operation can be visualized by the following diagram:

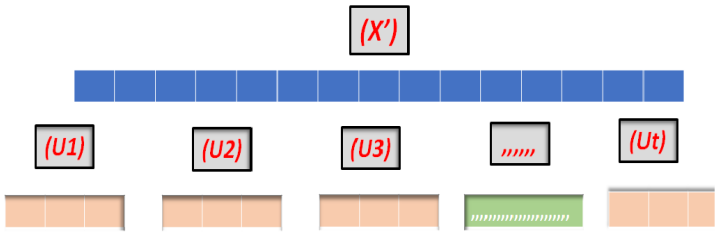


Figure3: Vector (X') subdivision

Axe 3: Encryption of vector (X′) and vector (W).

This transformation, which describes a profound improvement in the Vigenère technique, requires the construction of two (SW1)and (SW2) large S-Box of size (r; 256). each

1) (SW1) development

The construction of the replacement table (SW1) is determined by the following steps

a. First three rows Construction

- The first column is the permutation (P1) obtained by sorting on the first (256) values of the sequence (U).
- The first column is the permutation (P2) obtained by sorting on the first (256) values of the sequence (V)
- The first column is the permutation (P3) obtained by sorting on the first (256) values of the sequence (W)
  - The first line of the table (SW1) is the permutation (P1)
  - The second line of the table (SW1) is the permutation (P2)
  - The third line of the table (SW1) is the permutation (P3)

This construction is performed by the algorithm below:

Algorithm4: First three rows of (SW1)
1. For i = 1 to 256
2. SW1(1; i) = P1(i)
3. SW1(2; i) = P2(i)
4. SW1(3; i) = P3(i)
5. Next i

b. Following rows Definition.

According to the values of the vector obtained by the control vector (BT), the rank line (i > 3) is made up of the line (i – 1) and the line (i – 3) or the line (i – 2) by the line (i – 1) . The algorithm below serves as an explanation of this mechanism.

Algothm5: The following (SW1)rows
1. For $i = 3$ to r
2. For $j = 1$ to 256
3. If $BT(i; 2) = 0$ Then
4. $SW1(i; j) = SW1(i - 1; SW1(i - 3; j))$
5. Else
6. $SW1(i; j) = SW1(i - 2; SW1(i - 1; j))$
7. Nextj, i

Example:

(SW)	*1*	*2*	*3*	*4*	*5*	*6*	*7*	*8*	*CV*
P1	2	5	1	8	3	6	4	7	0
P2	5	1	8	3	6	2	7	4	1
P3	6	5	7	1	3	2	4	8	1
P4	2	5	7	6	8	1	3	4	1
P5	5	3	4	2	8	6	7	1	0

2) (SW2) development

The construction of the replacement table (SW2) is determined by the following steps

a. The first-row construction

- The first column is the permutation (Q1) obtained by sorting on the first (256) values of the sequence $(CT(:5))$.

The first line of the table (SW2) is the permutation (Q1)

Algorithm6: First (SW2)rows
1. For $i = 1$ to 256
2. $SW2(1; i) = Q1(i)$
3. Next i

b. The following rows Definition.

The algorithm provided below describes the mechanism that explains how the row of rank $(i > 1)$ in the control vector (BT) is shifted based on the values of the vector. Depending on the value of $BT: 2)$, the row is either a shift of row $CT(i; 2)$ from the previous row $(i - 1)$ or a shift of row $CT(i; 4)$ from the previous line $(i - 1)$.

Algoithm7: The following (SW2)rows
1. For $i = 3$ to r
2. For $j = 1$ to 256
3. If $BT(i; 1) = 0$ Then
4. $SW2(i; j) = SW2(i - 1; j \oplus CT(i; 2)))$
5. Else
6. $SW2(i; j) = SW2(i - 1; j \oplus CT(i; 4))$
7. Nextj, i

These two S-Boxes will be used as substitution tables in the Vigenère and Hill transformations

3) Substitution and diffusion function design

To establish the replacement functions, we need two pseudo-random vectors with coordinates in (G_r). This generation is given by the following algorithm

Amputated Block Encryption	
1. $QW(i) = \mathrm{mod}\left(((CT(i; 2) + CT(i; 4) * CT(i; 3))); r - 2\right) + 1$	(5)
2. $PW(i) = \mathrm{mod}\left(((CT(i; 1) + CT(i; 5) * CT(i; 2))); r - 3\right) + 2$	

The following formulas determine the new confusion and diffusion functions applied in our approach

VG1 First Substitution function	
$VG1(X(i)) = Y(i) = SW1(QW(i); SW2(PW(i); X(i) \oplus CT(i; 3))$	
VG2 second Substitution function	
$VG2(X(i)) = Y(i) = SW2(PW(i); SW1(QW(i); X(i) \oplus CT(i; 5))$	
DF1 First diffusion function	
$DF1(X(i)) = SW1(QW(i); Y(i - 1)) \oplus X(i)$	**(6)**
DF2 Second diffusion function	
$DF1(X(i)) = SW2(PW(i); Y(i - 1)) \oplus X(i)$	

In order to utilize the advanced Hill method, we must identify two invertible matrices with dimensions $(r; r)$ and $(q; q)$ correspondingly, as well as translation vectors with dimensions $(1; r)$ and $(1; q)$ correspondingly.

4) Encryption matrices Development.

Two invertible matrices are required for encryption, one $(K1)$ of dimension $(r; r)$ for the ciphering of the blocks (Ui) and a matrix $(K2)$ of dimension $(1; q)$ for the ciphering of the amputated block $(W$

These two matrices will be accompanied by dynamic translation vectors to overcome the linearity issue of the adapted transformation.

a. $(K1)$ **Encryption matrix development**

The matrix $(K1)$ is an upper triangular matrix of size $(r; r)$, whose main diagonal is 1 and whose last value is odd. The sub-diagonal consists of pseudorandom values

Algorithm8: **Encryption matrix Design**	
1. For $i = 1$ to $r - 1$ 2. $K1(i; i) = 1$ 3. If $BT(i; 2) = 0$ Then 4. $K1(i; i + 1) = CT(i; 2)$ 5. Else 6. $K1(i; i + 1) = CT(i; 5)$; end if	7. Next i 8. If $BT(r; 1) = 0$ Then 9. $K1(r, r) = Mod(2 * CT(I; 3) + +1; 256)$ 10. Else 11. $K1(r, r) = Mod(2 * CT(I; 2) + 1; 256)$ 12. End if
Example:	
$K_1 = \begin{pmatrix} 1 & CT(1:2) & 0 & 0 \\ 0 & 1 & CT(2;2) & 0 \\ 0 & 0 & 1 & CT(3;2) \\ 0 & 0 & 0 & 15 \end{pmatrix}$ with $r = 4$	$K_1 = \begin{pmatrix} 1 & CT(1:3) & 0 \\ 0 & 1 & CT(2;3) \\ 0 & 0 & 21 \end{pmatrix}$ with $r = 3$

b. $(K2)$ **Encryption matrix development**

The matrix $(K2)$ is an upper triangular matrix of size $(q; q)$if $q > 2$, whose main diagonal is 1 and whose last value is odd. The sub-diagonal consists of pseudorandom values

Algorithm9: **Encryption matrix Design**	
1. For $i = 1$ to $r - 1$ 2. $K2(i; i) = 1$ 3. If $BT(i; 1) = 0$ Then 4. $K2(i; i + 1) = CT(i; 1)$ 5. Else 6. $K2(i; i + 1) = CT(i; 4)$ 7. End if	8. Next i 9. If $BT(r; 1) = 0$ Then 10. $K2(r, r) = Mod(2 * CT(I; 3) + +1; 256)$ 11. Else 12. $K2(q; q) = Mod(2 * CT(I; 4) + 1; 256)$ 13. End if
Example:	
$K_2 = \begin{pmatrix} 1 & CT(1:2) & 0 & 0 \\ 0 & 1 & CT(1;4) & 0 \\ 0 & 0 & 1 & CT(3;1) \\ 0 & 0 & 0 & 15 \end{pmatrix}$ with $q = 4$	$K_2 = \begin{pmatrix} 1 & CT(1:1) & 0 \\ 0 & 1 & CT(2;4) \\ 0 & 0 & 19 \end{pmatrix}$ with $q = 3$

5) Translation table construction

We need to create a translation (VT) table for encrypting blocks of size $(r; t)$, as well as a vector (WT) of size $(1; q)$if $q > 2$ to translate the truncated block.

a. (VT) **table development**

The table (VT) of size $(t; r)$ is given by the following algorithm:

Algoithm10: (VT) construction
1. For $i = 1$ to r
2. For $j = 1$ to t
3. If $BT(i: 1) = 0$ then
4. $VT(i; j) = CT(i; 1) \oplus CT(j; 4)$
5. else
6. $VT(i; j) = CT(i; 3) \oplus CT(j; 5)$
7. Next j, i

b. (WT) **vector development**

The vector (WT) of size $(1: q)$ if $q > 2$ which serves as the translation vector of the amputated block in the Hill transformation is given by the following algorithm:

Algoithm11: (WT) construction
1. For $i = 1$ to q
2. $WT(i) = CT(i; 1) \oplus CT(i; 4)$
3. Next i

6) Block processing (W)

To address the vulnerability of differential attacks, the truncated (W) = ($X_{rt+1}; X_{rt+2}; \ldots\ldots; X_{3nm}$) block will undergo a two-step process. Firstly, it will undergo a Vigenère transformation utilizing two S-Boxes, along with the confusion function (VG1) and the diffusion function (DF1). Following that, in the second step, it will undergo an enhanced Hill transformation using the encryption matrix (K2) and the translation vector (WT).

a. Improved Vigenere application

This phase takes place according to the following steps:

i. First initialization value

This value is intended to modify the value of the pixel (X_{rt+1}) and start the block encryption process (W). It is calculated by the following algorithm

Algorithm12: First initialization value	
1. IS = 0	5. Else
2. For i = rt + 2 to 3nm	6. IS = X(i)⊕IS⊕CT(i; 3)
3. If BT(i; 1) = 0 Then	7. Next i
4. IS = X(i)⊕IS⊕CT(i; 5)	

This initialization value is closely related to the control vector BT(:1)

ii. Block Encryption

On the amputated block (W), we apply the new Vigenère technique provided by the confusion function (VG1), and to increase the impact of the avalanche effect, we implement the diffusion function (DF1). This operation is defined by the algorithm below

Algorithm13: Vigenere Encryption
1. IS=IS⊕ X_{rt+1}
Encryption of the startup pixel.
2. Y(rt + 1) = VG(IS)
Encryption of the subsequent blocks.
3. For i = rt + 2 to 3nm
4. x = DF1(Y(i − 1)
5. Y(i) = VG1(x)
6. Next i
7. Z = mod(K2(Y); 256) ⊕ WT

The cipher of the block (W) imputed is illustrated by the following figure:

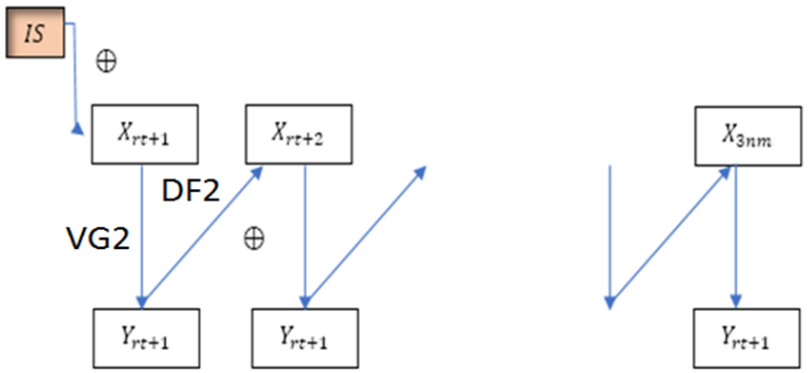


Figure4: Amputated Block Encryption

The vector (Y) obtained will be subjected to the affine Hill transformation given by the equation below:

Algoithm14: (Z) construction
1. Z = mod(K2Y; 256) ⊕ WT

7) Sub-block encryption:

The vector (X′) will be subjected to a Hill transformation ensured by a dynamic (K1) matrix of size (r; r) , accompanied at each iteration by a translation vector from the table (VT), to eliminate any linearity of the transformation

a. Second initialization value

The second initialization value is calculated to change the seed pixel value and start the encryption process. This value is determined by the following algorithm

Algorithm15: Second initialization value	
1. IV = 0	5. Else
2. For i = 2 to 3mn	6. IV = X(i)⊕IV⊕CT(i; 4)
3. If BT(i; 2) = 0 Then	7. Next i
4. IV = X(i)⊕IV⊕CT(i; 3)	

We note, that this calculated value is closely related to the original image and the control vector (BT(: 1)). A simple disturbance on the original image or on one of the parameters of the private key, will generate a new different initialization value and produce a different encrypted image.

b. Cipher function expression

The block (Wi) transformed from the block (Ui) by the fine transformation is given by the following formula:

Algorithm16: **New Hill expression**
1. If BT(i; 1) = 0 Then
2. $W_i = H(U_i)$
3. Else
4. $W_i = (U_i)^t H$

Encryption schema is given by the algorithm below:

Algorithm17: first Encryption
1. IV=IV⊕ X(1)
2. W(1) = VG1(IV) ⊕ WT
3. For i = 2 to r
4. x = VG1(W(i − 1)) ⊕ X(i)
5. If BT(i; 2) = 0 then
6. W(i) = VG2(x)
7. Else
8. W(i) = VG1(x)
9. end if
10. Next i

A vector (Z) is a Hill technique transformation of a vector (W) defined by a matrix (H) and translation vectors (T) and (V). A concatenation between the vector (Z) and the (Y) amputated image vector, represent the image encrypted by our algorithm. The encryption process is seen by the following figure:

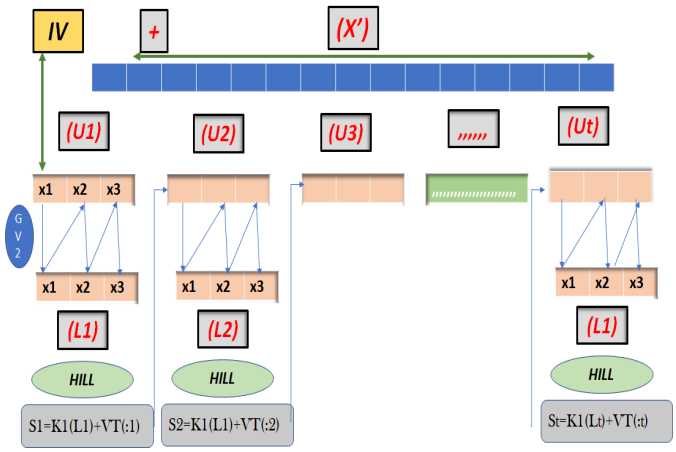


Figure4: (rt) blocs encryption

We note: (XC) final output vector which constitutes the encrypted image. This vector is given by the formula:

Encrypted image
1. For i = 1 to rt
2. XC(i) = S(i)
3. Next i
4. For i = 1 to q
5. XC(i + rt) = W(i)
6. Next i

Axe 4: Decryption encrypted images

Our method is a symmetric encryption system implemented by broadcasting. Therefore, during decryption, we apply the inverse encryption function starting from the last block. All functions used in our system are reversible, so there is a decryption function. The decryption process is based on the following steps:

1. Block Size (r) Calculation
2. Amputation of last block for reverse treatment
3. Application of HILL's inverse technique
4. Inverse affine functions application
5. Inverse Substitution Tables Development
6. Replacement inverse diffusion function
7. Original image reconstruction

1) HILL's inverse formula

The affine Hill transformation used is reversible. In effect

Algorithm16: **New Hill invers**
5. $Y(i) = K2X(i) \oplus WT$
6. $K2X(i) = Y(i) \oplus WT$
7. $X(i) = K2^{-1}(Y(i) \oplus WT)$

2) Vigenere Reciprocal matrix

Likewise, the Vigenère substitution transformation used is invertible. In effect

a. Reverse substitution table

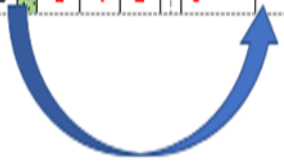
The inverse substitution table of the Vigenère function is given by the algorithm below

Algorithm16: Reverse $S-Box$
1. for $i = 1$ to 256
2. for $j = 1$ to 256
3. $WS1(i, SW1(i, j)) = j$
4. $WS2((i, SW2(i, j)) = j$
5. Next j, i

EXAMPLE

(VG)	1	2	3	4	5	6	7	0	CR	KL	CL
1	3	5	0	6	2	7	1	4			
2	2	7	1	4	3	5	0	6	1	5	4
3	4	3	5	0	6	2	7	1	1	3	5
4	2	7	1	4	3	5	0	6	0	3	4
5	0		2	7	1	4	3	5	1	4	2

(GV)	1	2	3	4	5	6	7	0
1	7	5	1	0	2	4	6	3
2	3	1	5	4	6	0	2	7
3	0	6	2	1	3	5	7	4
4	3	1	5	4	6	0	2	7
5	5	3	7	6	0	2	4	1



b. Expression of the reciprocal transformation

By following the same logic of Vigenere's traditional technique, we obtain

Algorithm16: Reverse function
1. If $z = SW1(y, x)$
2. Then
3. $x = WS1(y, z)$

c. Reciprocal of the function (GV1)

We note (GV1) the inverse transformation of the function (VG1), it is defined by the following algorithm:

(GV1) definition	
We have	(7)
1. $VG1(X(i)) = Y(i) = SW1(QW(i); SW2(PW(i); X(i) \oplus CT(i; 3))$	
2. So	
3. $SW2(PW(i); X(i) \oplus CT(i; 3)) = WS1(QW(i); Y(i)$	
4. $X(i) \oplus CT(i; 3) = WS2(PW(i); WS1(QW(i); Y(i))$	
5. $X(i) = WS2(PW(i); WS1(QW(i); Y(i)) \oplus CT$	

d. Reciprocal of the function (GV2)

We note (GV2) the inverse transformation of the function (VG2), it is defined by the following algorithm:

(GV2) definition	
We have	(7)
6. $VG2(X(i)) = Y(i) = SW2(PW(i); SW1(QW(i); X(i) \oplus CT(i; 5))$	
7. So	
8. $SW1(QW(i); X(i) \oplus CT(i; 5)) = WS2(PW(i); Y(i)$	
9. $X(i) \oplus CT(i; 5) = WS1(QW(i); WS2(PW(i); Y(i))$	
10. $X(i) = WS1(QW(i); WS2(PW(i); Y(i)) \oplus CT(i: 5)$	

3) Inverse diffusion function

a. Reciprocal of the function (DF1)

We note (FD1) the inverse transformation of the function (DF1), it is defined by the following algorithm:

(FD1) definition	
We have	(7)
11. $DF1(X(i)) = Y(i) = SW1(QW(i); Y(i-1)) \oplus X(i))$	
So	
12. $Y(i-1)) \oplus X(i) = WS1(QW(i); Y(i))$	
13. $X(i) = WS1(QW(i); Y(i)) \oplus X(i)$	

b. Reciprocal of the function (DF2)

We note (FD2) the inverse transformation of the function (DF2), it is defined by the following algorithm:

(FD2) definition	
We have	(7)
14. $DF2(X(i)) = Y(i) = SW2(PW(i); Y(i-1)) \oplus X(i))$	
So	
15. $Y(i-1)) \oplus X(i) = WS2(PW(i); Y(i))$	
16. $X(i) = WS2(PW(i); Y(i)) \oplus X(i)$	

III. Examples and simulations

In order to measure the performance of our encryption system, we randomly select a large number of reference images, and then use our method to test them

1) Brutal assaults

They consist in reconstructing the encryption keys in a random manner.

1.1. Key-space analysis

Our method's chaotic sequence provides high initial condition sensitivity and can fend off any vicious assaults. The components of our system's secret key are:

Logistic Map		PWLCM Map		A.J map	
$u_0 = 0{,}6548$	$\mu = 3{,}965$	$v_0 = 0{,}53$	$d = 0.63$	$w_0 = 0{,}51$	$p = 1.53$

Each parameter is a single precision real so it is encoded in 32 bits only. Therefore, the size of the secret key is greatly exceeding $\approx 2^{32*6} \gg 2^{120} \gg 2^{100}$, which is enough to avoid any brutal attacks.

1.2. Secret key's sensitivity Analysis

Our encryption key has high sensitivity, which means that small degradations in the single parameters used will automatically lead to large differences from the original image. The figure below illustrates this confirmation

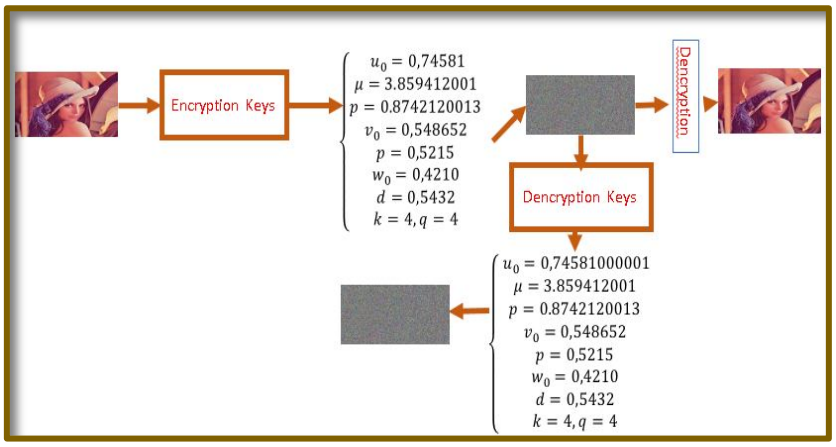
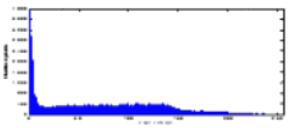
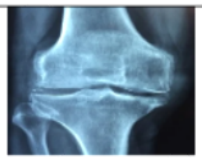
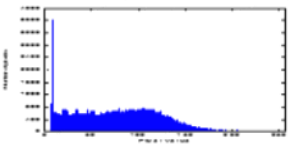
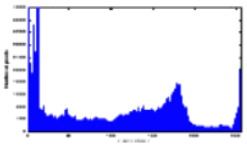


Figure4: Encryption key sensitivity

This makes sure that the original image cannot be recovered without knowing the genuine encryption secret key. In the accompanying table, we show the results of encrypting some of the most prominent images that are used as references in the literature using our novel method.

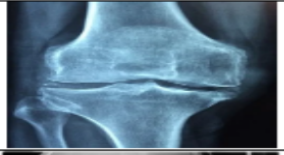
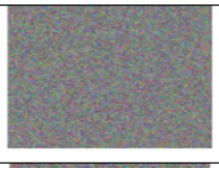
Table2: Images tested by our technology

Nom	Original image	Taille	Histogramme claire
Img1		245x411	
Img2		807x595	
Img3		1057x1200	

2) Visual test

The initial test conducted is the visual assessment, aimed at identifying any resemblances between the original image and the image encrypted using the new cryptographic system. Through our simulations, it is evident that the encrypted image appears entirely distinct from the original image, exhibiting no resemblance whatsoever. This observation provides an initial confirmation of the robustness of our method, as depicted in the accompanying table.

Table3: Visual test of images chooses

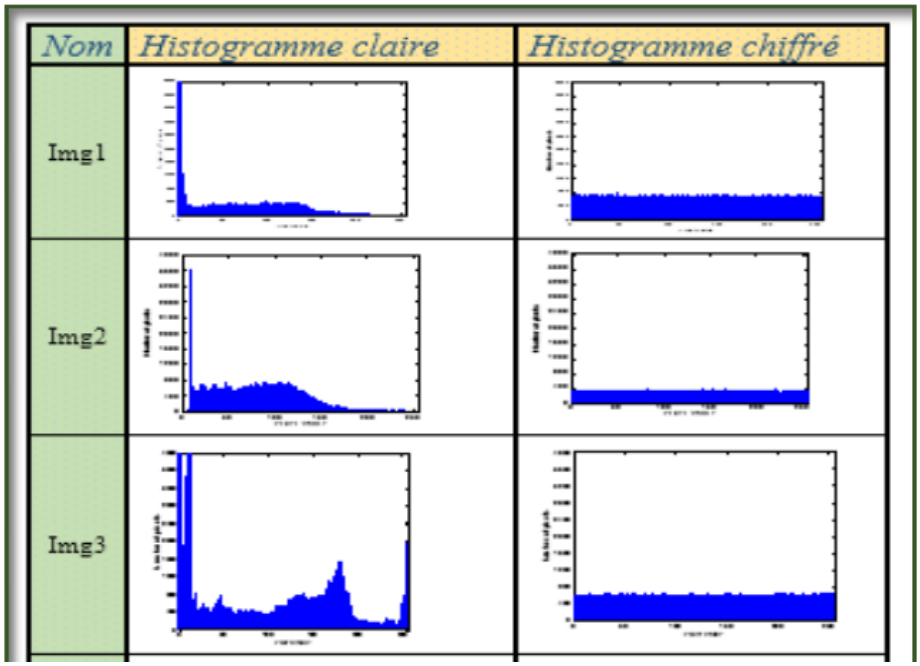
Nom	Image crypté	Image Originale
Img1		
Img2		
Img3		

This table confirms the absence of any information from the original image contained in the encrypted image.

3) Histogram analysis

All of these images that our program evaluated had histograms that are uniformly flat and dispersed. The histograms of the clear images and encrypted images that our algorithm selected are shown in the table below.

Table4: Encrypted image histogram



4) Statistics Attack Security

4.1. Entropy Analysis

Entropy is the measure of the disorder diffused by a source without memory. The entropy expression is determined by the equation below

Entropy	
$H(MC) = \sum_{i=1}^{t} -p(i)\log_2(p(i))$	**(8)**

$p(i)$, is the probability of occurrence of level (i) in the image encrypted by our new method. This value should approach the maximum value (8 bits).

The entropy values of images chosen arbitrarily from a large database of images of different sizes and formats, tested by our method are illustrated in the table below

Table5: entropy increase

Image N°	*Entropy*	
	Original Image	*Cypher Image*
Img1	*4,23747*	*7,9997*
Img2	*7,7666*	*7,9996*
Img3	*7,0157*	*7,9997*

The entropy values of all the images are close to the maximum value which ensures a strong resistance to pa-entropy statistical attacks

4.2. Correlation analysis

Correlation is a technique that compares two images to estimate the displacement of pixels in one image relative to another reference image. The relevant expression is defined by the following equation

Correlation coefficient

$$r = \frac{\operatorname{cov}(x, y)}{\sqrt{V(x)}\sqrt{V(y)}} \quad (9)$$

. The correlation value increases images are illustrated in the following table

Table6: Correlation increase

Name	Original Image			Cypher Image		
	Horizontal	Vertical	Diagonal	Horizontal	Vertical	Diagonal
Img1	0.9196	0.921	0.878	0.0014	0.0016	0.0024
Img2	0.967	0.987	0.956	0.0021	0.0013	-0.0033
Img3	0.968	0.990	0.958	0.0022	0.0020	-0.0014

5) Differential analysis

Let be two encrypted images, whose corresponding free-to-air images differ by only one pixel, from ($\mathbf{C_1}$)and($\mathbf{C_2}$), respectively. The NPCR mathematical analysis of an image is given by the equation below

NPCR Expression

$$NPCR = \left(\frac{1}{nm}\sum_{i,j=1}^{nm} D(i,j)\right) * 100$$

$$\text{With} \quad D(i,j) = \begin{cases} 1 & \text{if} \quad C_1(i,j) \neq C_2(i,j) \\ 0 & \text{if} \quad C_1(i,j) = C_2(i,j) \end{cases} \quad (10)$$

The UACI mathematical analysis of an image is given by the below

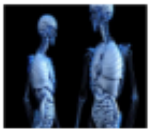
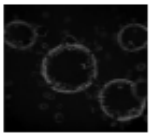
UACI Value

$$\left(\frac{1}{nm}\sum_{i,j=1}^{nm} |C_1(i,j) - C_2(i,j)|\right) * 100 \quad (11)$$

Differential constants are very important in any cipher system. The values collected by the encryption of the reference images are grouped in the following table

Table7: Avalanche effect

NPCR and UACI values.

Image						
NPCR	99.61	99.61	99.60	99.63	99.61	99.62
UACI	33.42	33.45	33.44	33.45	33.47	33.43

6) Avalanche effect

The avalanche effect is an essential characteristic found in almost all cryptographic hash functions and block coding algorithms. It results in increasingly significant alterations as data propagates through the algorithm's structure. This factor determines the extent of the avalanche effect within the cryptographic system. Its impact can be approximated using the equation provided below.

AE Value

$$\left(AE = \frac{\sum_i \text{bit change}}{\sum_i \text{bit total}}\right) * 100 \quad (12)$$

The table below gives the rate of change of the bits of the images encrypted by our system

Table8: Avalanche effect

Image N°	AE(%)
Img1	52
Img2	51,23
Img3	53,14

All values returned from the (AE) by our method are all in the range of residual values [76,96 78]. This guarantees that a one- bit change in the clear image will be reflected by a change of at least 76% of the encrypted image's bits.

IV. Math security

The chaotic map employed in our research exhibits extreme sensitivity to the initial conditions, meaning that even slight modifications to the private key or the original image will result in significant alterations to the encrypted image and the generated chaotic vectors. As a result, it becomes practically impossible to recover the original image. Reconstructing S-boxes presents a considerable challenge due to the generation of pseudo-randomly sized S-boxes, characterized by their unpredictable sizes. The system benefits from enhanced protection against unforeseen intrusions, thanks to the novel confusion and the pseudo-random nature of the diffusion function. Furthermore, the system demonstrates remarkable resilience due to the large size of the encrypted matrix and the computational complexity associated with its inverse calculation.

V. Advantages of this process

This method encapsulates several advantages of which we mention

- Encryption keys from chaotic maps are extremely sensitive to initial conditions, making it difficult to recover the real key used.
- The $S-Box$ structure and its use under the control of a chaotic decision vector increases the attack complexity of our technique.
- The pseudo-random size of the substitution table makes it difficult to reconstruct the S-Boxes.
- The large size of encryption matrices
- The random size of the HILL matrix

1) Approach limitation

In theory, every encryption system can be compromised given an enormous amount of time and processing power, utilizing cutting-edge hardware. Currently, all existing systems are considered unbreakable; however, this state of affairs is subject to change due to the rapid advancement of computers. It has created an ongoing race between the ability to crack codes and the strength of the encryption algorithms themselves. Even the most robust and secure encryption systems in use today will inevitably face attacks and potential breaches within a few years. Fortunately, these very tools also hold the potential to pave the way for the development of the next generation of unbreakable encryption techniques.

The limits of our method largely depend on the limits of the choices of the chaotic maps and the construction of the S-Boxes, and the pseudo-random size of the developed S-Boxes.

VI. Conclusion

Our system incorporates a robust implementation of strong chaining between encrypted blocks and subsequent clear blocks. Additionally, the utilization of a large-sized encryption matrix, calculated from chaotic maps, enhances the resilience of our new method. These characteristics are derived from the construction of pseudo-random size substitution tables employed in our innovative technology. By utilizing encryption secret keys and S-Boxes of significant size, our cryptosystem remains safeguarded against brute-force and frequency-based attacks. When subjected to evaluation using various images, our approach exhibited statistical and differential constants that ensured its effectiveness against known attacks

Competing Interests

All authors of this article confirming the absence of any conflict between them, and there are no private or public organizations or laboratories to fund this research, thus avoiding any expected conflicts.
This document does not contain any research or experiments conducted on animals or humans.

Funding

No government or private agency has financial work. This article is a great effort of the authors.

Acknowledgment:

Nothing to report. We publish this article to help the scientific community only

Compliance with Ethical Standards

Our article is in line with the ethics of the newspaper

Research Data Policy and Data Availability Statements

In this article no scientific material or animal is used, except personal computers which remain at our disposal

References

A. Jarjar « Improvement of hill's classical method in image cryptography » International Journal of Statistics and Applied Mathematics 2017, Volume 2 Issue 3, Part A

X Wang, » An image encryption algorithm based on pixel bit operation and nonlinear chaotic system » The Visual Computer, 2022 - Springer

M Jarjar" New technology of color image encryption based on chaos and two improved Vigenère steps » Multimedia Tools and …, 2022 -

Xiuli Chai « Combining improved genetic algorithm and matrix semi-tensor product (STP) in color image encryption» Signal Processing Volume 183, June 2021, 108041

N Avinash « A new kind of encryption and decryption of RGB color image using permutation matrix multiplication » American Institute of …, 2022 - researchgate.net

ST Kamal «A new image encryption algorithm for grey and color medical images EEE …, 2021 - ieeexplore.ieee.org»

A.S alkhalid « cryptanalyze of Hill cipher using genetic algorithm" dalam IEEE hanmument 2015

A Pour Jabbar Kari, « A new image encryption scheme based on hybrid chaotic maps» Multimedia Tools and applications…, 2021 - Springer

Y Qobbi «New image encryption scheme based on dynamic substitution and hill cypher» 2020, 2022 - Springer

Z Hua, « Color image encryption using orthogonal Latin squares and a new 2D chaotic system » Nonlinear Dynamics, 2021

Li X, Meng X, Yang X et al (2018) Multiple-image encryption via lifting wavelet transform and XOR operation based on compressive ghost imaging scheme. »Optics Lasers Engineering 102(3):106–111

S Zhang » A novel image encryption algorithm based on SPWLCM and DNA coding » Mathematics and Computers in Simulation, 2021

A Pourjabba « A new image encryption scheme based on hybrid chaotic maps» Multimedia Tools and …, 2021

X Gao « Image encryption algorithm based on 2D hyperchaotic map » Optics & Laser Technology, 2021

A JarJar "New chaotic map development and its application in encrypted color image" Journal of Multimedia Information System, 2021 - jmis.org

HR Shakir » Chaotic-DNA system for efficient image encryption »Bulletin of Electrical Engineering and …, 2022 - beei.org

MZ Talhaoui, « A new one-dimensional cosine polynomial chaotic map and its use in image encryption» The Visual Computer, 2021

S Wang « An image encryption scheme using a chaotic neural network and a network with multistable hyperchaos» Optik, 2022

B Jasra,4 Color image encryption and authentication using dynamic DNA encoding and hyper chaotic system » Expert Systems with Applications, 2022 - Elsevier

A Abid » Two Enhanced Feistel Steps for Medical Image Encryption » 2022 IEEE 3rd …, 2022 - ieeexplore.ieee.org

NHM Ismail « An improved image encryption algorithm based on Bezier coefficients matrix » journal of King Saud University-Computer and application 2022 - Elsevier