Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

CONNECTION OF TWO CONVENTIONAL ENHANCED TO ENCRYPT COLOR IMAGES

Said Hraoui
LIASSE, ENSA, SMBA University
Fez, Morocco
said.hraoui@usmba.ac.ma,

Abdellatif JarJar
Moulay Rachid High School,
Taza, Morocco
abdoujjar@gmail.com

Abstract

This document introduces a new cryptosystem mixing two improvement standards generally used for text encryption, in order to give birth a new color image encryption algorithm capable of dealing with known attacks. Firstly, two substitution matrixes attached to a strong replacement function will be generated for advanced Vigenere technique application. At the end of this first round, the output vector is subdivided into size blocks according to the used chaotic map, for acting a single enhanced Hill circuit insured by a large inversible matrix. A detailed description of such a large involutive matrix constructed using Kronecker products will be given. accompanied by a dynamic translation vector to eliminate any linearity. A solid chaining is established between the encrypted block and the next clear block to avoid any differential attack. Simulations carried out on a large volume of images of different sizes and formats ensure that our approach is not exposed to any known attacks.

Article Highlights

This new algorithm is the mixture result of two deeply improved classical systems which we mention the most important changes made.

- New size substitution matrix (256,256) Construction
- Strong replacement functions definition
- Improved Vigenere technology application
- Kronecker products Theoretical reminder
- Kronecker product Application
- Large invertible matrix design
- Single enhanced Hill revolution application

Notation

$$\begin{cases} G_t = {}^{Z}/_{tZ}\ ring \\ A(j:): Line\ number\ j\ of\ matrix\ A \\ A(:j): columm\ number\ j\ of\ matrix\ A \\ \oplus: Xor\ operator\ \otimes: \text{Kronecker product} \\ E(x):\ The\ whole\ part\ of\ the\ real\ number\ x \end{cases}$$

Key word: **Vigenere grid; Involutive matrix; Chaotic map; Broadcast function; genetic operator**

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

I. 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 $[7-8]$.

1) Vigenere's classical technique

This technology is based on static (V)matrix defined by the following algorithm

$$algorithm1 \begin{cases} Fist\ Row \\ For\ \ i=1\ \ to\ 26 \\ V(1,i)=i \\ Next\ i \\ folloying\ Rows \\ For\ \ i=2\ \ to\ 26 \\ For\ \ j=1\ \ to\ 26 \\ V(i,j)=V(i-1,(j+1),26) \\ Next\ j,i \end{cases}$$

Let (P): plain text, (C): cypher text; (K): Encryption key, (V) Vigenere matrix and (l): length of clear text. So

$$equation1 \begin{cases} C_i = V(P_i,K_i) = (P_i + K_i) \mod 26 \\ P_i = V(C_i,K_i) = (P_i - K_i) \mod 26 \end{cases}$$

Even though Vigenere's matrix was known, the encryption was able to withstand several centuries. But, Babagh's cryptanalysis is not efficient in not knowing the size of the encryption key. Several attempts to improve Vigenere's technique have invaded the digital world we quote $[9-10]$. In this work, the new structure of the substitution matrix and its attached replacement function will be described in detail.

2) Hill's classical technique

Hill's method is a classic symmetric encryption system that uses a linear transformation provided by an invertible matrix in a carefully selected ring $[11-12]$.

(C):plain text (C'): cypher text; (K): Encryption key, the method is defined by the following linear formula

$$Equation\ 2 \begin{cases} C' = KC \\ So \\ C = K^{-1}C' \end{cases}$$

Unfortunately, this technique is still exposed to known attacks. Since then, over time, several improvements have been made to the system $[13-14]$. The improvement of the classic Hill technique tracked in this document is to prove the construction of a large invertible matrix whose size will be randomly calculated based on the chaotic graph used. Some advanced techniques in the same axis should be taken into consideration $[15-16]$.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

3) Our contribution

Our contribution in this article is to couple two conventional deeply improved systems to better adapt to color image encryption. These improvements, made on the two classic systems are

$$\begin{cases} Vigenere \\ New\ \ substitution\ matrices\ construction \\ New\ \ \ attached\ replacement\ functions\ definition \\ Application\ of\ a\ diffusion\ ensured\ by\ two\ size\ maatrix\ (256{,}256) \\ Hill \\ Large\ key\ matrix\ Construction \\ Applying\ an\ affine\ transformation \\ Dynamic\ translation\ vector \end{cases}$$

II. THE PROPOSED METHOD

Based on chaos, our new technology is divided into three main stages

$$\begin{cases} first\ phase: \\ action\ on\ a\ single\ pixel\ with\ broadcast\ installation \\ Second\ phase: \\ subdivision\ of\ the\ image\ into\ blocks\ of\ size\ (4r) \\ Third\ phase \\ Construction\ of\ an\ invertible\ matrix\ of\ size\ (4r, 4r) \\ action\ on\ blocks\ of\ (4r) pixels\ by\ a\ Hill\ lap \end{cases}$$

The image below shows this technology steps

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

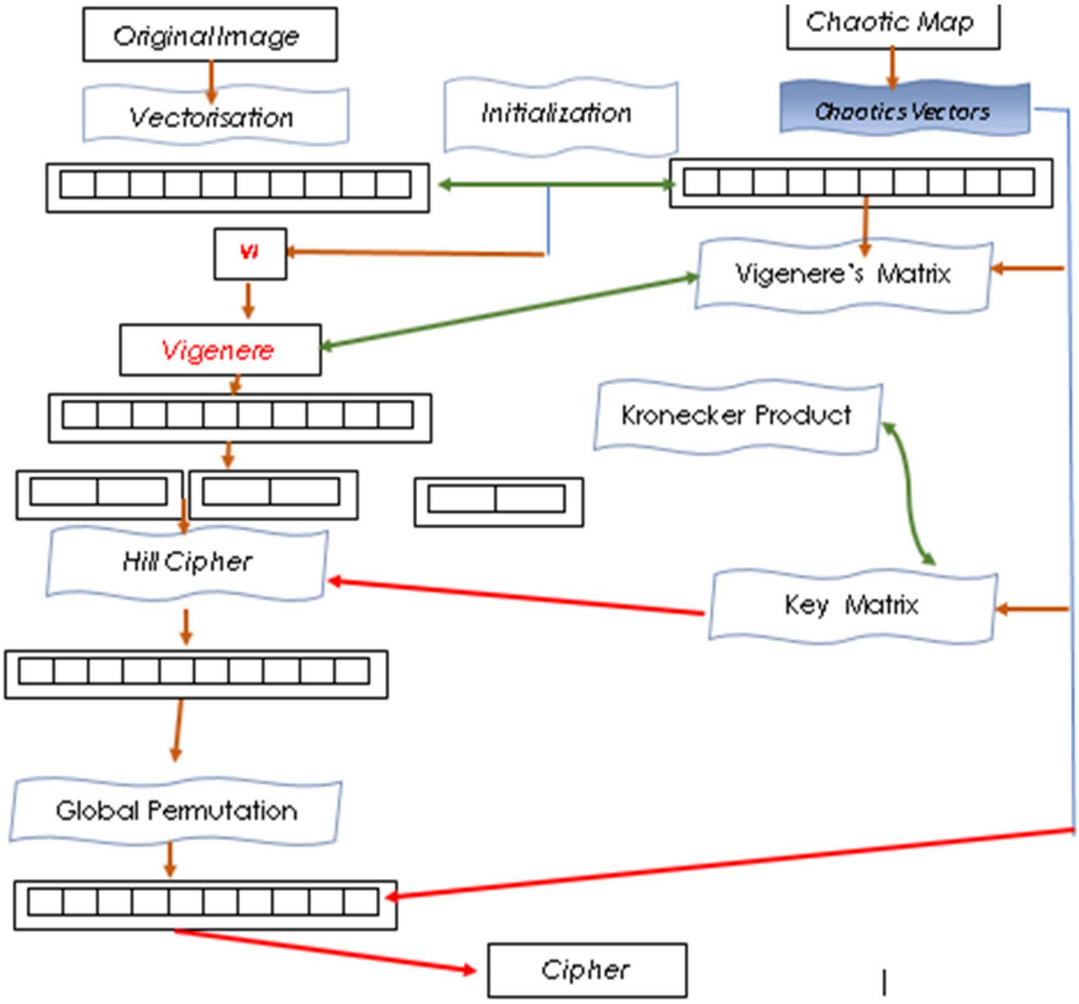


Figure2: Encryption scheme

This new cryptosystem is based on the following axes

AXE 1: CHAOTIC SEQUENCES DEVELOPMENT

In order to build a new algorithm using a single-encryption key, we will use the 2d logistics map. This choice is due to the simplicity of its development and its high sensitivity to the initial parameters.

1) 2D Logistics Map

$$\begin{cases} x_0, y_0 \quad \epsilon]0 \quad 1[\\ x_{n+1} = \mu_1 x_n (1 - x_n) + \mu_2 y_n^2 \\ y_{n+1} = \mu_3 y_n (1 - y_n) + \mu_4 (x_n^2 + x_n y_n) \\ \mu_1 \epsilon [2{,}75 \quad 3{,}4] \\ \mu_2 \epsilon [0;15 \quad 0{,}21] \\ \mu_3 \epsilon [2{,}75 \quad 3{,}45] \\ \mu_4 \epsilon [0{,}13 \quad 0{,}15] \end{cases}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

2) Chaotic used vector design

Our work requires the construction of three chaotic vectors(CL), $(KR)$and (KL) , with a coefficient in (G_{256}), and two (CR) , (CV)binary vector will be regarded as the control vector. This construct is seen by the following algorithm

$$Algorithm2 \begin{cases} for\ \ i = 1\ to\ 3nm \\ CL(i) = \ mod\left(E\left(\dfrac{x(i) + \sup\left(x(i), y(i)\right)}{2} * 10^{11}, 254\right) + 1\right) \\ KL(i) = \ mod\left(E\left(\dfrac{x(i) * y(i) + \inf\left(x(i), y(i)\right)v(i)}{2} * 10^{10}, 253\right) + 2\right) \\ KR(i) = E\left(\dfrac{KL(i) + CL(i)}{2}\right) \\ if\ x(i) \geq y(i)\ then \\ CR(i) = 0\ else\ CR(i) = 1 \\ end\ if \\ if\ Cl(i) < KL(i) then \\ CV(i) = 1 \\ else\ CV(i) = 0 \\ end\ if \\ Next\ i \end{cases}$$

3) Global permutation Design

Sorting the (CR)vector in a large decreasing order generates the (PH) global permutation This method is illustrated by the following algorithm

$$Algorithm3 \begin{cases} k = 1 \\ For\ i = 3nm\ to\ 1 \\ If\ CR(i) = 0\ Then\ PH(i) = k \\ k = k + 1 \\ end\ if \\ Next\ i \end{cases} \qquad \begin{cases} For\ i = 1\ to\ 3nm \\ If\ CR(i) = 0\ Then\ PH(i) = k \\ k = k + 1 \\ end\ if \\ Next\ i \end{cases}$$

Example:

(CR)	0	0	1	1	1	0	1	0	1	1	0	1	0	1	1	0	0	1	1	0
Sort	9	8	10	11	12	7	13	6	14	15	5	16	4	17	18	3	2	19	20	1

$$(PH) = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 & 13 & 14 & 15 & 16 & 17 & 18 & 19 & 20 \\ 9 & 8 & 10 & 11 & 12 & 7 & 13 & 6 & 14 & 15 & 5 & 16 & 4 & 17 & 18 & 3 & 2 & 19 & 20 & 1 \end{pmatrix}$$

III. INSTALL THE NEW ENCRYPTION METHOD

Based on chaos, our new technology is based on two encryption processes each using a deep improvement of a classic system.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

AXE 2: VIGENERE UPGRADE

In the first stage, Vigenere's technology was greatly modified by integrating the new substitution matrix provided by the new powerful replacement function.

1) Original image Vectorization

After the three (RGB) color channels extraction and their conversion into size vectors $(\boldsymbol{Vr})$, $(\boldsymbol{Vg})$, $(\boldsymbol{Vb})$ $(\boldsymbol{1, nm})$ each, a concatenation is established to generate a vector $\boldsymbol{X(x_1, x_2, \ldots\ldots, x_{3nm})}$ of size $(1,3nm)$. This operation is described by the following algorithm

$$Algorithm\ 5 \begin{cases} for\ i = 2\ tonm \\ X(3i-2) = Vr(i) \\ X(3i-1) = Vg(i) \\ X(3i) = Vb(i) \\ Next\ i \end{cases}$$

2) Initialization Value Design

The$(IV1)$ initialization value must be recalculated to change the value of the starting pixel. Ultimately, the $(\boldsymbol{IV1})$ value is provided by the next algorithm

$$Algorithm\ 6 \begin{cases} for\ i = 2\ to\ l \\ if\ CV(i) = 0\ then \\ v_1 = v_1 \oplus X(i) \oplus CL(i) \\ Else \\ v_1 = v_1 \oplus X(i) \oplus KL(i) \\ Next\ i \end{cases}$$

This initialization value is set only to modify the value of the seed pixel and start the encryption process. By implementing chaotic mapping in the calculation of the initialization value, the problem of uniform image color (black, white ,...) can be solved.

3) Vigenere's advanced methods

This new technology requires the establishment of two (VG) and (VD) substitution matrices through the process described by the following steps

- permutation (RP) obtained by descending ordering the first 256 $values$ of the sequence (U)
- permutation (RR) obtained by increasing the ordering the first 256 $values$ of the sequence (V),

with the following restrictions

$$Equation\ 7 \quad \begin{cases} if RP(i) = 256\ the\ RP(i) = 0 \\ if RR(i) = 256\ the\ RR(i) = 0 \end{cases}$$

This new construction is entirely supervised by the vector (CR) seen as a decision vector. It is given by the following algorithm

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

$$algorithm7 \begin{cases} Fist\ Row \\ For\ \ i = 1\ \ to\ 256 \\ VG(1,i) = RP(i) \\ VD(1,i) = RR(i) \\ Next\ i \end{cases} \begin{cases} For\ \ i = 2\ \ to\ 256 \\ For\ \ j = 1\ \ to\ 256 \\ if\ CV(i) = 1\ then \\ VG(i,j) = VG\left(i-1, RP\left(mod\left(j + CL(i)\right), 256\right)\right) \\ VD(i,j) = VD\left(i-1, RR\left(mod\left(j + KL(i)\right), 256\right)\right) \\ else \\ VG(i,j) = VG\left(i-1, RP\left(mod\left(j + KL(i)\right), 256\right)\right) \\ VD(i,j) = VD\left(i-1, RR\left(mod\left(j + CL(i)\right), 256\right)\right) \\ end\ if \\ next\ j, i \end{cases}$$

We note that the construction of the two matrices is completely determined by the (CV) decision vector

Example: in $(\boldsymbol{G_8})$

(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

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

a) New Vigenere's mathematical expression

The classical Vigenere transformation expression is given by the following formula

$$Equation8 \quad \{ \quad Y(i) \ = VG\left(CL(i), X(i)\right)$$

The following equation illustrates the effective expression of the $Y(i)$ image of the pixel $X(i)$ through the new Hill technology.

$$Equation9 \begin{cases} if\ CR(i) = 0\ then \\ Y(i) \ = VG(CL(i), VD(KR(i), (mod(a * X(i), 256) \oplus KL(i))) \\ else \\ Y(i) \ = VD(KL(i), VG(KR(i), (mod(c * X(i), 256) \oplus CL(i))) \\ With; a; c \ \in G_{256}^{*} \\ Let \begin{cases} a = mod\left(2 * mod\left(\sum_{i=1}^{n} CL(i)\right) + 1, \quad 256\right) \\ c = mod\left(2 * mod\left(\sum_{i=1}^{nm} KL(i)\right) + 1, \quad 256\right) \end{cases} \end{cases}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

We noticed that this new expression is firmly attached to the (CR) decision vector, and uses two general substitution matrices to contain powerful substitution functions.

2) *First Encryption Process*

With a powerful broadcast function, this first encryption process will protect the system from differential attacks and increase the time complexity of the attack. The figure below illustrates the first encryption process improved using only Vigenere

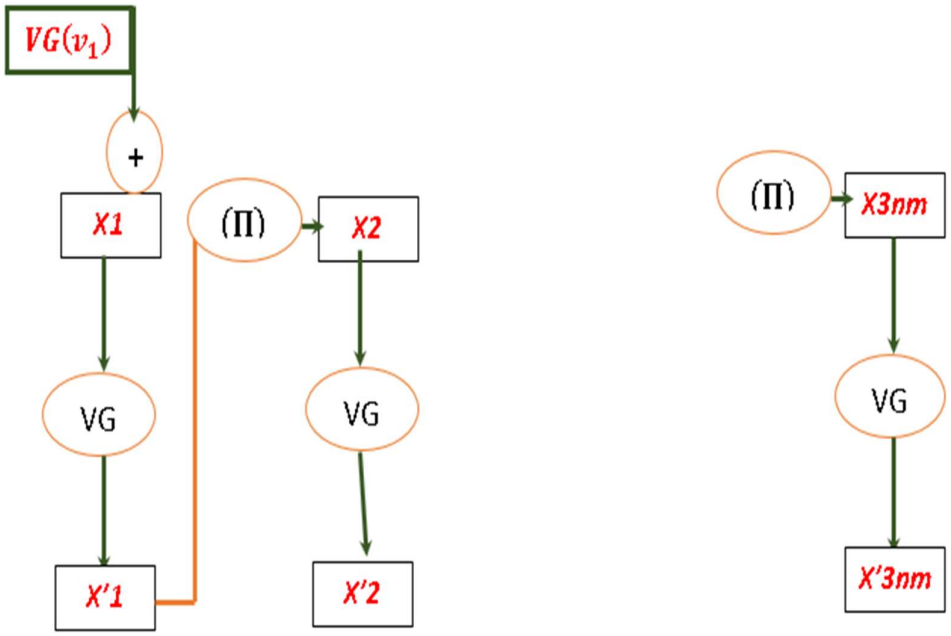


Figure4 : Vigenere Encryption

$$\begin{cases} \text{(VG): advanced substitution matrix} \\ \text{(Π): New diffusion function} \end{cases}$$

(Π) The new diffusion function using matrix (VD) is described by the following formula

$$Equation10 \quad \begin{cases} \Pi\big(X(i+1)\big) = VD\big(KR(i), X'(i) \oplus X(i+1)\big) \end{cases}$$

This schema is translated by the algorithm below

$$Algorithm8 \begin{cases} X'(1) = VG\big(KR(1), VD(CL(1), v_1) \oplus X(1)\big) \\ For\ i = 2\ to\ 3nm \\ \Pi\big(X(i)\big) = VD\big(KR(i), X'(i-1) \oplus X(i)\big) \\ if\ CR(i) = 0\ then \\ X'(i) = VG\big(CL(i), \big(mod(a * \Pi\big(X(i)\big), 256) \oplus KL(i)\big)\big) \\ else \\ X'(i) = VG\big(KL(i), \big(mod(c * \Pi\big(X(i)\big), 256) \oplus CL(i)\big)\big) \\ Next\ i \end{cases}$$

We note that this first step uses only substitutions, which ensures an extreme speed in the execution. The output vector $\boldsymbol{X'(x'_1, x'_2, \ldots\ldots, x'_{3nm})}$, will undergo a second encryption attempt.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

AXE 3: APPLYING HILL UPGRADE

Before starting the second encryption stage, we will introduce in detail some mathematical knowledge useful for constructing inversible matrix and, install a hill lap.

1) Math reminder

Several new knowledges with mathematical bases must be clarified

a. related information

In this section, we will introduce the concept of diastereomeric matrices and Kronecker products used to construct large invertible matrices.

i. Involutive matrix

a) Definition

A is an involutive matrix if and only if we have

$$Equation14 \quad A^{-1} = A$$

In other words

$$Equatiopn15 \quad A^2 = I \quad I\ identity\ matrix$$

b. Building the (A) matrix

Assume that the size of $(\boldsymbol{A})$ matrix is $(2r_h, 2r_h)$, and is defined by block as follows

$$A = \begin{pmatrix} A_1 & A_2 \\ A_3 & A_4 \end{pmatrix} A_i\ of\ size\ (r_h, r_h)$$

We got

$$A^2 = \begin{pmatrix} A_1 & A_2 \\ A_3 & A_4 \end{pmatrix}\begin{pmatrix} A_1 & A_2 \\ A_3 & A_4 \end{pmatrix} = \begin{pmatrix} A_1^2 + A_3A_2 & A_2A_1 + A_4A_2 \\ A_1A_{3+}A_3A_4 & A_4^2 + A_2A_3 \end{pmatrix} = \begin{pmatrix} I & 0 \\ 0 & I \end{pmatrix} \quad I: Identity\ Matrix$$

Since matrix (A) is involutive, we get

$$Equation16 \begin{cases} A_1^2 + A_2A_3 = I \\ A_2A_1 + A_4A_2 = 0\ \ with \quad 0: Null\ matrix \\ A_1A_{3+}A_3A_4 = 0 \\ A_4^2 + A_3A_2 = I \end{cases}$$

So

$$Equation17 \begin{cases} A_2A_3 = I - A_1^2 = (I - A_1)(I + A_1) \\ A_3A_2 = I - A_4^2 = (I - A_4)(I + A_4) \end{cases}$$

Since (A_1) matrix is given randomly, other matrices can be selected by the following formula

$$Equation18 \quad \begin{cases} A_2 = k(I - A_1) \\ A_3 = k^{-1}(I + A_1) \\ A_4 = 256 - A_1 \end{cases} \quad (k \in G_{256}^*) \quad and \quad \begin{cases} A_1 \neq I \\ A_1 \neq 0 \end{cases}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

Or We can take

$$Equation19 \quad \begin{cases} A_2 = k(I + A_1) \\ A_3 = k^{-1}(I - A_1) \\ A_4 = 256 - A_1 \end{cases} \; This\ gives \; A = \begin{pmatrix} A_1 & k(I - A_1) \\ k^{-1}(I + A_1) & 256 - A_1 \end{pmatrix}$$

Example

$$\boldsymbol{In\ G_8} \qquad A_1 \begin{pmatrix} 1 & 3 \\ 2 & 5 \end{pmatrix} \quad k = 3\ and\ \ k^{-1} = 3\ So \quad A = \begin{pmatrix} 1 & 3 & 0 & 7 \\ 2 & 5 & 2 & 4 \\ 6 & 1 & 7 & 5 \\ 6 & 2 & 6 & 3 \end{pmatrix}$$

Under these conditions, the obtained matrix (A) is involute. This choice is not unique, and it increases the complexity of matrix (A) reconstruction.

2) Kronecker matrix product

Matrix theory is widely used in cryptography. Due to its difficulties, classical computing techniques are rarely used. For example, it is difficult to reverse the large matrix using conventional methods. At present, the Kronecker product and the tensor product are not only connected with physics or biology, but also take a safe path in cryptography. We are going to take advantage of the Kronecker product to increase the size of the involutive matrix.

a. Definition

$Let\ A = (a_{i,j})\ of\ size\ (n,m)$ and $Let\ B = (b_{i,j})\ of\ size\ (p,q)$, We call the Kronecker product of $A\ and\ B$, denoted by

$$Equation\ 20 \quad C = A \otimes B = \begin{pmatrix} a_{1,1}B & \cdots & a_{1,m}B \\ \vdots & \ddots & \vdots \\ a_{n,1}B & \cdots & a_{n,m}B \end{pmatrix} of\ size\ (np; mq)$$

Example

$$\begin{cases} A = \begin{pmatrix} 1 & 3 \\ 2 & 5 \end{pmatrix} \quad B = \begin{pmatrix} 1 & 2 & 0 \\ 3 & 1 & 2 \\ 0 & 5 & 4 \end{pmatrix} so \quad C = A \otimes B = \begin{pmatrix} 1B & 3B \\ 2B & 5B \end{pmatrix} = \begin{pmatrix} 1 & 2 & 0 & 3 & 6 & 0 \\ 3 & 1 & 2 & 9 & 3 & 6 \\ 0 & 5 & 4 & 0 & 15 & 12 \\ 2 & 4 & 0 & 5 & 10 & 0 \\ 6 & 2 & 4 & 15 & 5 & 10 \\ 0 & 10 & 8 & 0 & 25 & 20 \end{pmatrix} \end{cases}$$

b. Kronecker Product Properties

The product is not commutative

$$A \otimes B \approx B \otimes A$$

c. Reversibility

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

$$Equation\ 21 \quad \begin{cases} if\ A\ is\ invertible\ and\ if\ B\ is\ invertible\ Then \\ (A \otimes B)^{-1} = A^{-1} \otimes B^{-1} \end{cases}$$

Note:

If $A\ and\ B$ are involutive matrices, then their Kronecker product is involutive and we have

$$Equation\ 22 \quad (A \otimes B)^{-1} = A^{-1} \otimes B^{-1} = A \otimes B$$

3) (X') Vector Adaptation

In order to facilitate the implementation of the second encryption stage, the vector (X') must be cut into blocks uniform, we are going to calculate two constants $r_1\ and\ r_2$

$$Equation\ 23 \quad \begin{cases} \boldsymbol{r_1} = \left(mod\left(\frac{1}{n} \sum_{i=1}^{n} |CL(i) - KL(i)|, 4 \right) + 2 \right)\ so\ \ 2 \leq \boldsymbol{r_1} \leq \boldsymbol{5} \\ \boldsymbol{r_2} = \left(mod\left(\frac{1}{m} \sum_{i=1}^{m} |CL(i) - KR(i)|, 3 \right) + 2 \right)\ so\ 2 \leq \boldsymbol{r_2} \leq \boldsymbol{4} \end{cases}$$

We put

$$16 \leq r = 4\boldsymbol{r_1 r_2} \leq \boldsymbol{80}$$

Therefore, the new vector size (X') can be obtained by the following formula

$$Equation\ 24 \quad \begin{cases} let \quad 3nm \equiv s\ [\boldsymbol{r}] \\ l = 3nm - s \\ t = \frac{l}{r} \end{cases}$$

By truncating the last (s) (X') pixel, and placing it in the(XD) vector after modification, the (XT) adaptive vectorcan be obtained by the following algorithm

$$Algorithm\ 9 \quad \begin{cases} for\ \ i = 1\ \ to\ \ l \\ XT(i) = X'(i) \\ Next\ i \\ If\ s \neq 0\ then \begin{cases} Amputated\ pixel\ storage \\ for\ i = 1\ \ to\ \ s \\ if\ CR(i + l) = 0\ then \\ XD(i) = X'(i + l) \oplus CL(i + l) \\ else \\ XD(i) = X'(i + l) \oplus KL(i + l) \\ end\ if \\ Next\ \ i \\ end\ if \end{cases} \end{cases}$$

4) Switching to a size matrix

The following figure illustrates the transaction from vector to size matrix$(\boldsymbol{t}, r)$ used in the Hill circuit

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

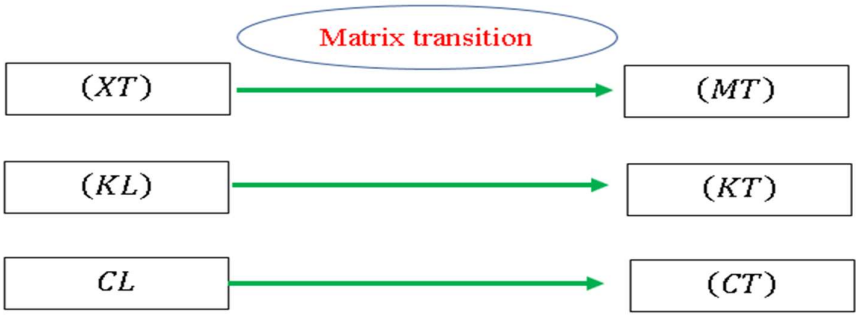


Figure6: Transition to matrices (t, r)

Hill's classical method is predominantly oriented on invertible matrices in accurately indexed rings. Inverting a large matrix remains the main challenge of this technique, which prompts scholars to use only small matrices typically of size <= 4.

2) *New Hill matrix Development*

In our algorithm, in order to overcome this problem, we will introduce in detail a new simple method of constructing large invertible matrix based on involute matrix.

Let (A_1) be an arbitrary matrix of size (r_1, r_1) and let the matrix (A) defined by

$$Equation\ 25 \quad A = \begin{pmatrix} A_1 & k(I - A_1) \\ k^{-1}(I + A_1) & 256 - A_1 \end{pmatrix} \ (k \in G_{256}^*) \ of\ size\ (2r_1, 2r_1)$$

Let (B_1) be an arbitrary matrix of size (r_2, r_2) and let the matrix (B) defined by

$$Equation\ 26 \quad B = \begin{pmatrix} B_1 & h(I - B_1) \\ h^{-1}(I + B_1) & 256 - B_1 \end{pmatrix} \ (h \in G_{256}^*) \ of\ size\ (2r_2, 2r_2)$$

The two matrices $(A)\ and\ (B)$ are involutive

We put

$$Equation\ 27 \quad H = A \otimes B$$

The matrix (C) is invertible and involutive of size $(4r, 4r)$. The matrix (C) is condadate for a key Hill matrix.

3) *New encryption matrix construction*

This second encryption process is illustrated by the following figure

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

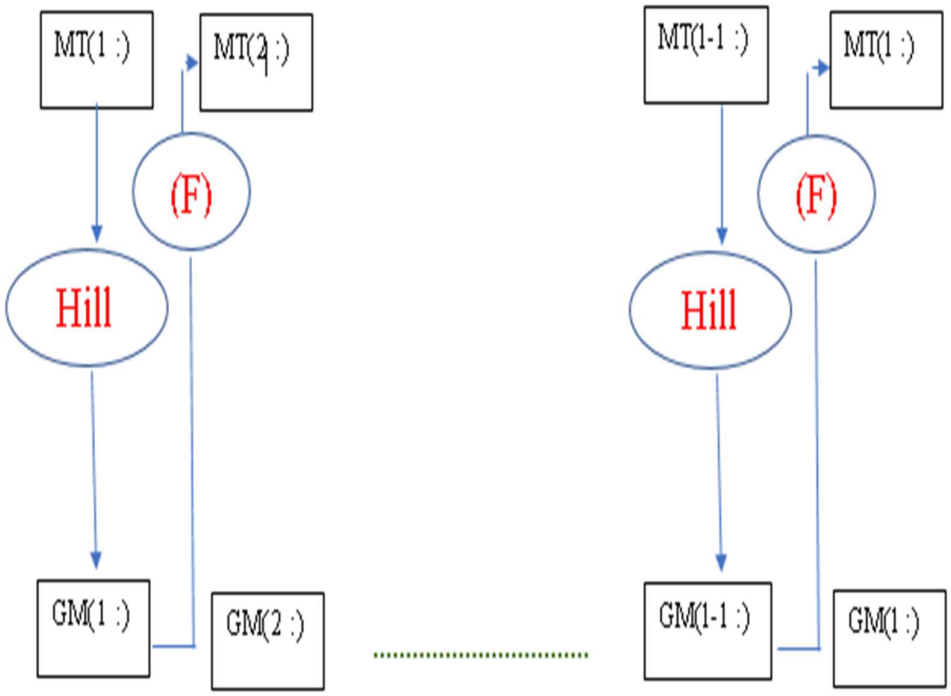


Figure6: Hill Encryption

$$\begin{cases}(HILL): enhanced\ HILL\ function \\ (F): New\ difusion\ function\end{cases}$$

4) Enhanced HILL function expression

$$Algorithm11 \quad Hill \begin{cases} if\ CV(k) = 0\ then \\ GM(k:) = H * MT(k:) \oplus KT(k:) \\ else \\ GM(k:) = H * MT(k:) \oplus CT(k:) \end{cases}$$

Build diffusion to increase the impact of the avalanche effect

$$Equation\ 28\ \{F(MT(k+1:)) = GM(k:) \oplus MT(k+1:) \oplus CT(k:)$$

An insertion of the pixels of the vector (XD) at the end of the obtained vector (CX) on which the permutation (PH) is established by the following algorithm

$$Algorithm16 \begin{cases} for\ i1 = 1\ to\ 3nm \\ XC(i) = CX(PH(i)) \\ Next\ i \end{cases}$$

The vector (XC) represents the encrypted image.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

STEP 5: DECRYPTION ENCRYPTED IMAGES

The solid chaining establishes between the encrypted block and the next clear block in the broadcast process, forcing us to start decryption from the last block using the opposite functions. So, the decryption process should follow these steps

- Read the encrypted image and switch to vector (XC)
- Application of the permutation (HP) inverse of the permutation (PH)

$$Algorithm17\begin{cases} For\ i = 1\ to\ 3nm \\ HP\big(PH(i)\big) = i \\ Next\ i \end{cases}$$

$$Algorithm18\begin{cases} For\ i = 1\ to\ 3nm \\ CX(i) = XC(HP(i) \\ Next\ i \end{cases}$$

- Recalculates the value $(\boldsymbol{r})$ size of the subdivision blocks
- Recalculates $(A)\ and\ (B)$
- Reverse Hill application $(A^{-1} = A, B^{-1} = B\ )\ H = A \otimes B = H^{-1}$

1) Mathematical HILL inverse expression

$$Algorithm11\quad HILL^{-1}\begin{cases} if\ CV(k) = 0\ then \\ MT(k:) = H^{-1}(MT(k:) \oplus KT(k:)) \\ else \\ MT(k:) = H^{-1}(MT(k:) \oplus CT(k:)) \end{cases}$$

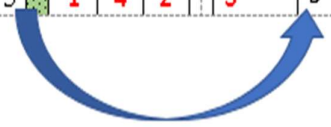
- Reconstruct the image vector and add vector pixels (XD) to the queue
- Reverse Vigenere application

$$Algorithm20\begin{cases} for\ \ i = 1\ \ to\ \ 256 \\ for\ j = 1\ to\ \ 256 \\ GV\big(i, VG(i,j)\big) = j \\ DV\big(\big(i, VD(i,j)\big) = j \\ Next\ j, i \end{cases}$$

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



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

$$Equation21\quad \begin{cases} if\ \ z = VG(y,x) \\ Then \\ x = GV(y,z) \end{cases}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

2) Mathematical Vigenere inverse expression

We have

$$Algorithm8\begin{cases} For\ i = 2\ to\ 3nm \\ \Pi(X(i)) = VD\left(KR(i), X^{'}(i-1) \oplus X(i)\right) \\ if\ CR(i) = 0\ then \\ X'(i) = VG\left(CL(i), \left(mod(a * \Pi(X(i)), 256) \oplus KL(i)\right)\right. \\ else \\ X'(i) = VG\left(KL(i), \left(mod(c * \Pi(X(i)), 256) \oplus CL(i)\right)\right. \\ Next\ i \end{cases}$$

So

$$\begin{cases} X'(i) = VG\left(CL(i), \left(mod(a * \Pi(X(i)), 256) \oplus KL(i)\right)\right. \\ \left(mod(a * \Pi(X(i)), 256) \oplus KL(i) = GV(CL(i), X'(i)\right). \\ \left(mod(a * \Pi(X(i)), 256) = GV(CL(i), X'(i)) \oplus KL(i)\right. \\ \Pi(X(i)) = a^{-1}\left(GV(CL(i), X'(i)) \oplus KL(i)\right) \\ VD(KR(i), X'(i-1) \oplus X(i)) = a^{-1}\left(GV(CL(i), X'(i)) \oplus KL(i)\right) \\ X'(i-1) \oplus X(i) = DV\left(KR(i), a^{-1}\left(GV(CL(i), X'(i)) \oplus KL(i)\right)\right) \\ X(i) = DV\left(KR(i), a^{-1}\left(GV(CL(i), X'(i)) \oplus KL(i)\right)\right) \oplus X'(i-1) \end{cases}$$

AXIS6: 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.

a) Key-space analysis

The chaotic sequence used in our method ensures strong sensitivity to initial conditions and can protect it from any brutal attacks. The secret key to our system consists of

$$x_0 = 0{,}7655412001\ ,\ \mu_1 = 3.89231541,$$
$$y_0 = 0.865421331,\quad \mu_2 = 0{,}563215$$
$$\mu_3 = 1{,}3561 \qquad \mu_4 = 0{,}563215$$

If we use single-precision real numbers $\mathbf{10^{-10}}$ to operate, the total size of the key will greatly exceed $\approx \mathbf{2^{180}} \gg \mathbf{2^{110}}$, which is enough to avoid any brutal attacks.

b) Secret key's sensitivity Analysis

Our encryption key has a high sensitivity, which means that a small degradation of a single parameter used will automatically cause a large difference from the original image. The image below illustrates this confirmation

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

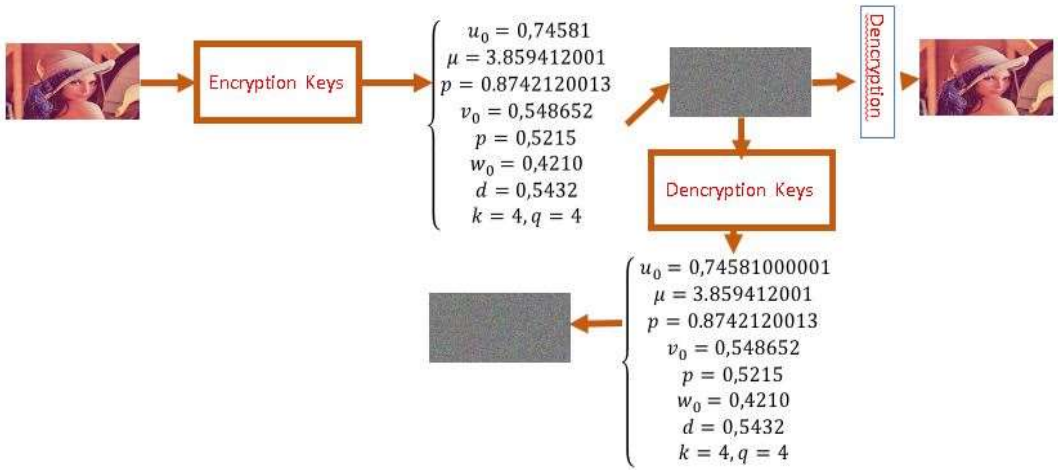


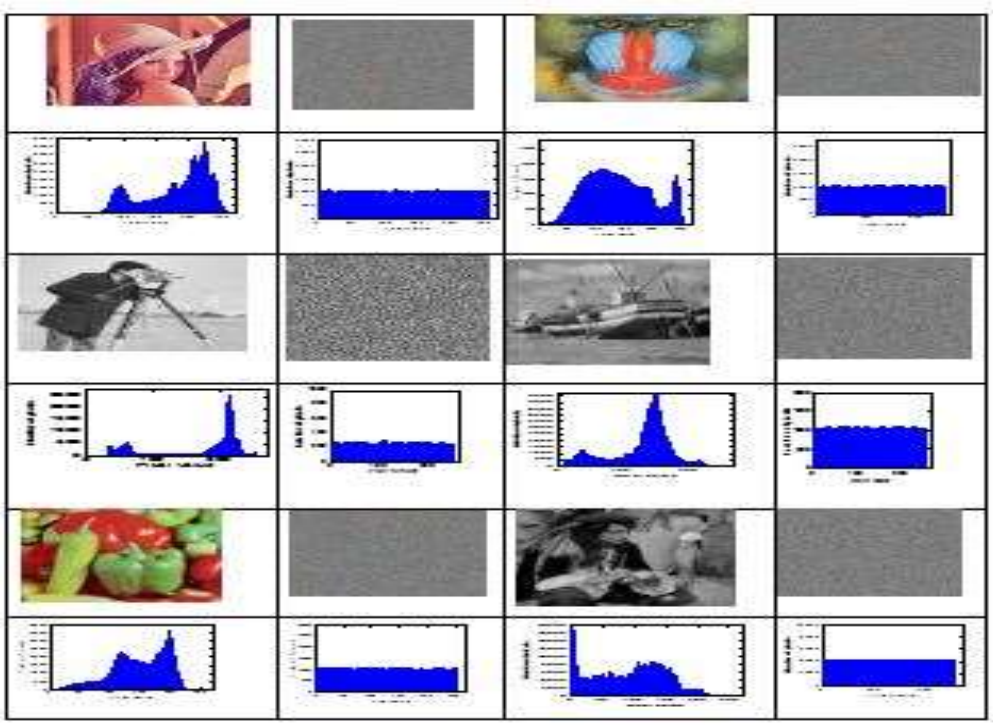
Figure11: Encryption key sensitivity

This ensures that in the absence of the real encryption key, the original image cannot be restored.

a) Hystogram analysis

all images tested by our algorithm have a uniformly distributed histogram. This reflects that the entropy of the encrypted images is around 8, which makes the system immune to histogram attacks. The table1 shows that the horizontal correlation values of the encrypted images are close to zero. This ensures high security against correlation attacks.

Table 1: Encrypted image histogram



2) Statistics Attack Security

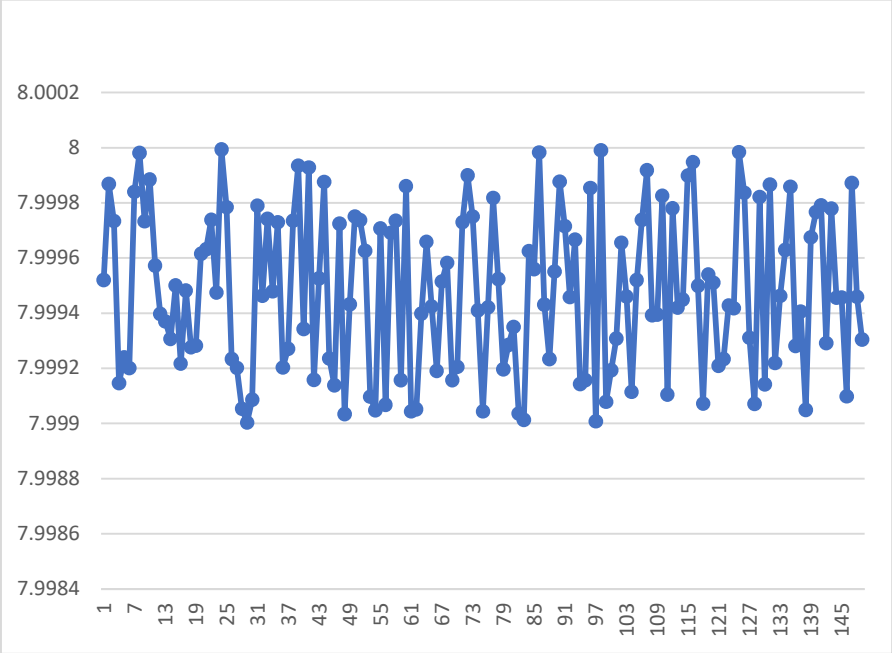
a) Entropy Analysis

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

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

$$Equation\ 23 \qquad H(MC) = \frac{1}{t}\sum_{i=1}^{t} -p(i)\ log_2(p(i))$$

The entropy values on the 150 *images* arbitrarily chosen from a large database of images of different sizes and formats, tested by our method are represented graphically by the following figure



Fugure12: Entropy of 150 images

All the entropy values of the images tested by our algorithm are close to 8, which confirms the uniformity of the histogram. This proves that the method is far from a statistical attack.

a) Entropy statistical analysis

We will study the uniformity of the distribution of entropy released by the test.

(a) Position parameter analysis

The values derived from the entropy by applying our approach to over 150 images in our image database, constitute a statistical series with position, dispersion and concentration parameters have been recalculated to verify the safety of our approach.

The purpose of this analysis is to show that the distribution follows a reduced central normal distribution. So

$$\text{Equation 24} \begin{cases} Q_1 = \ First\ quartile \\ Q_2 = \ Second\ quartile \\ Q_3 = \ Third\ quartile \end{cases}$$

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

Position Parameters

Average	Max	Min	Q1	Q2	Q3
7,999480662	7,999993872	7,999004283	7,99921863	7,99946043	7,999737

The moustache box of the entropy is illustrated in the diagram in Figure below

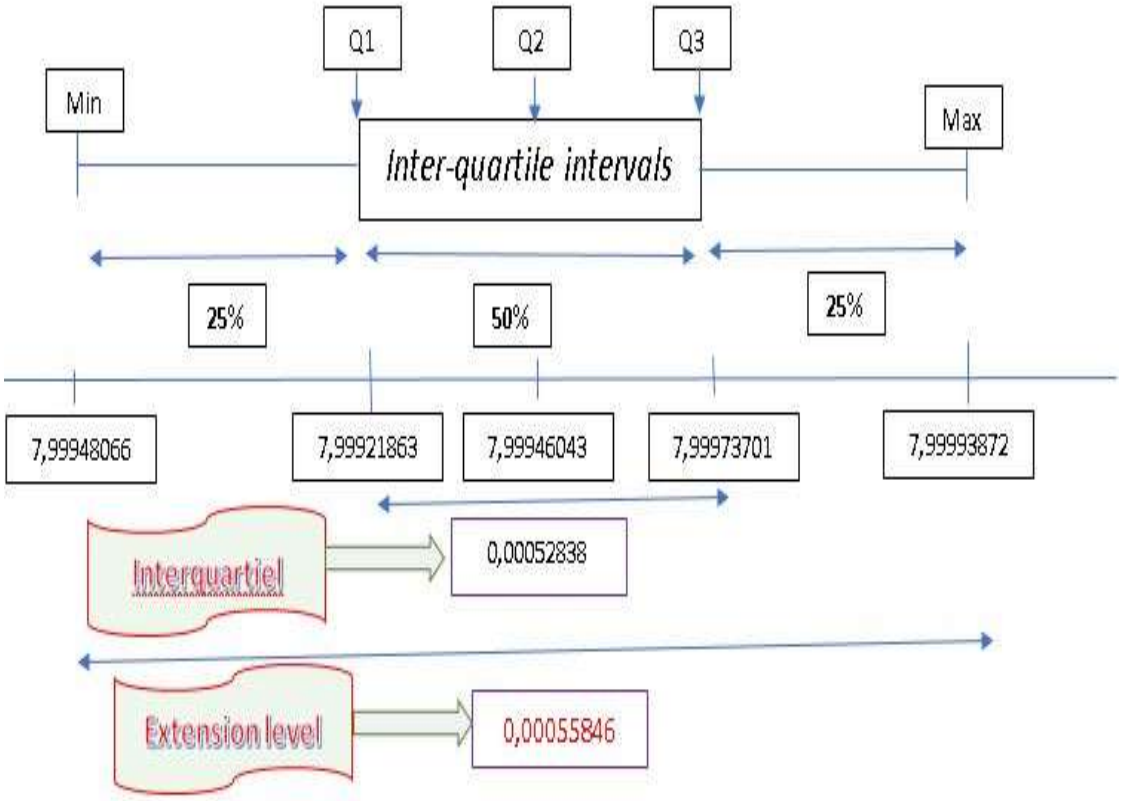


Figure13: Entropy moustache box

a) Correlation analysis

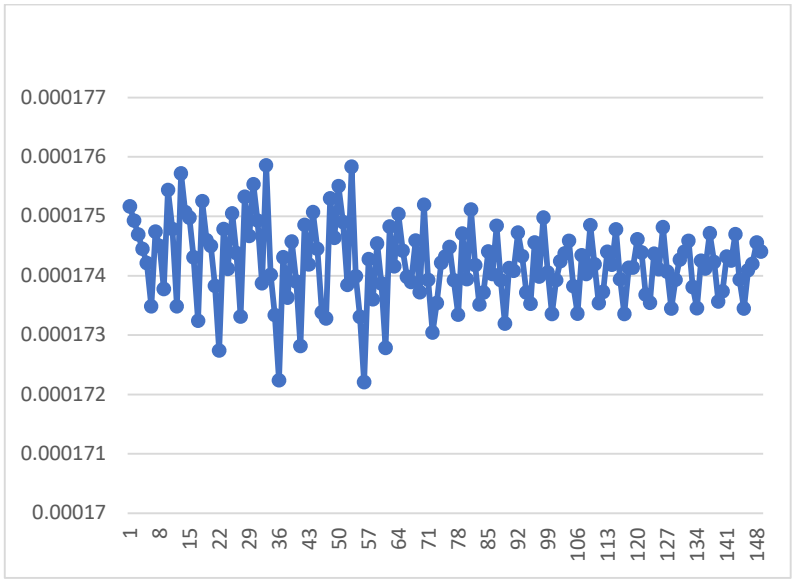
Correlation is a technique that compares two images to estimate the displacement of pixels in one image relative to another reference image. The relevant expression is defined by the following equation

$$equation\ 29\ \ correlation \qquad r = \frac{cov(x,y)}{\sqrt{V(x)}\sqrt{V(y)}}$$

(a) Horizontal Correlation

The following figure graphically represents the simulation of 150 color images of the same size, which are selected from an image database of various sizes, formats and related values

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1



Fugure15: Entropy of 150 images

(b) Vertical Correlation

Simulations made on 150 *images* of the database gave the vertical correlation scores are displayed in Figure below

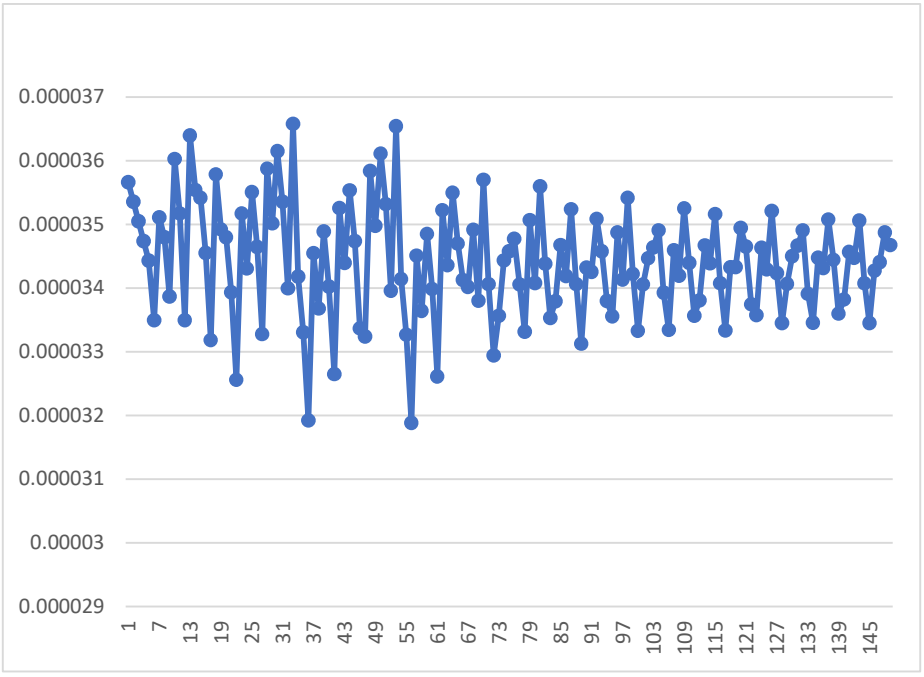


fIgure16: Vertical correlation of 150 images

Figure 17 shows that the vertical correlation values of the encrypted images are close to zero. This ensures high security against correlation attacks.

(c) Diagonal Correlation

Simulations made on 150 images of the database gave the diagonal correlation scores are displayed in Figure below

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

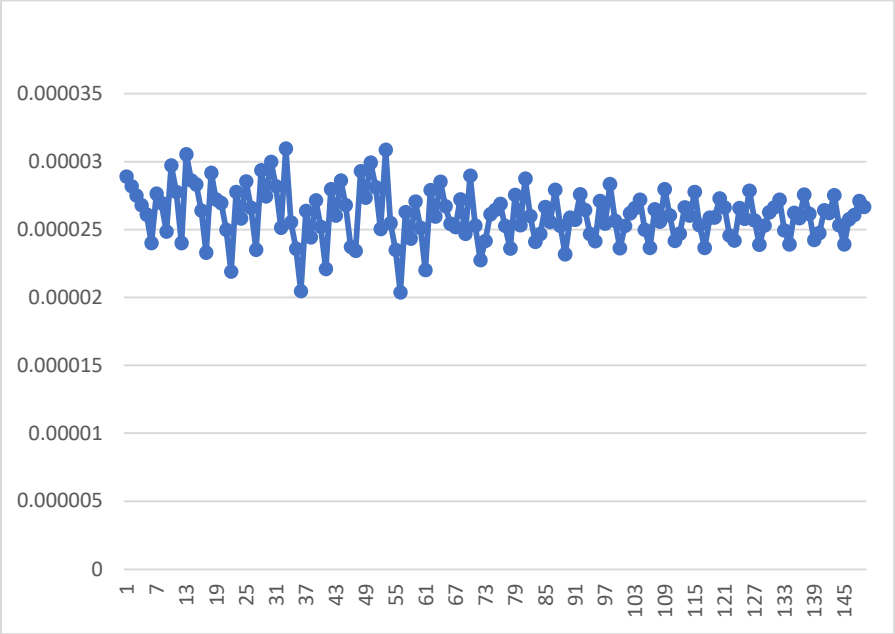


figure17: Diagonal correlation of 150 images

Figure above shows that the diagonal correlation values of the encrypted images are close to zero. This ensures high security against correlation attacks.

b) Differential analysis

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

$$Equation\ 30 \qquad NPCR = \left(\frac{1}{nm}\sum_{i,j=1}^{nm} D(i,j)\right) * 100$$

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

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

$$Equation\ 31 \qquad UACI = \left(\frac{1}{nm}\sum_{i,j=1}^{nm} Abs\big(C_1(i,j) - C_2(i,j)\big)\right) * 100$$

The study of the 150 $selected\ images$ revealed the following diagram

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

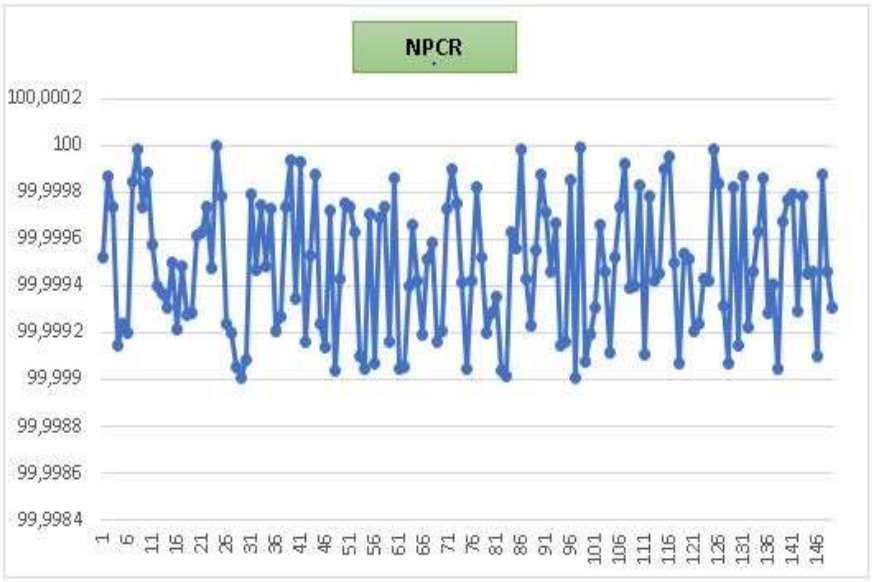


figure18: NPCR of 150 images

All detected values are inside the confidence interval [99,63 99,95]. These values are largely sufficient to affirm that our crypto system is protected from known differential attacks

The study of the 150 selected images revealed the following diagram

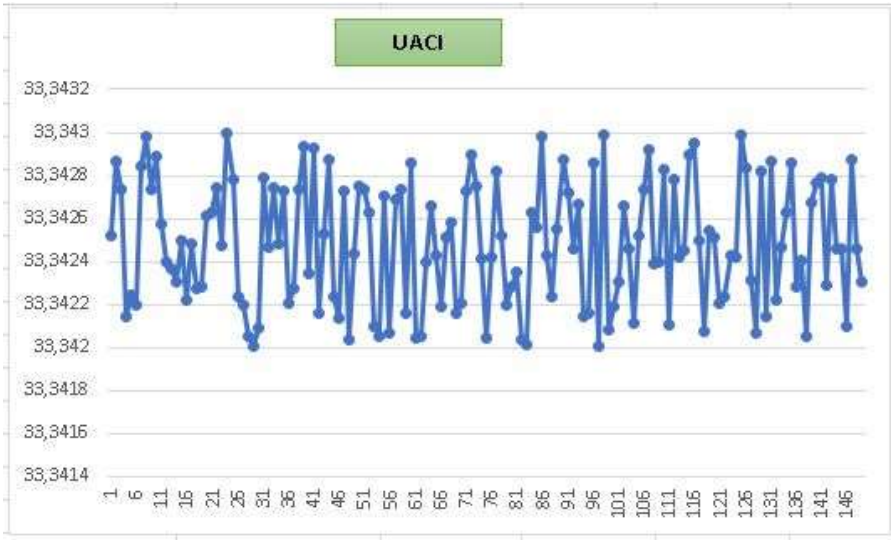


figure19: UACI of 150 images

All detected values are inside the confidence interval [33; 34 33,35]. These values are largely sufficient to affirm that our crypto system is protected from known differential attacks.

a) Avalanche effect

The avalanche effect is a required property in virtually all cryptographic hash functions and block coding algorithms. It causes progressively more important changes as the data is propagating in the structure of the algorithm. This constant determines the avalanche impact of the cryptographic structure in place. It is approximated by the equation below

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

$$Equation32 \quad AE = \left(\frac{\sum_i bit\ change}{\sum_i bit\ total}\right) * 100$$

Figure below depicts the evaluation of the AE score for 150 images examined by our approach.

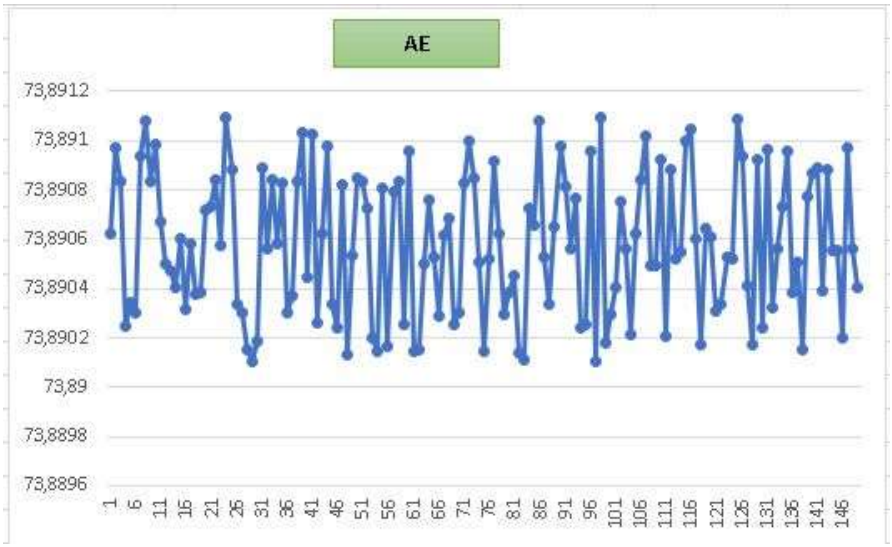


Figure20: Avalanche effect

All values returned from the (AE) by our method are all in the range of residual values $[73{,}96 \quad 74; 02]$. This guarantees that a one- bit change in the clear image will be reflected by a change of at least 78% of the encrypted image's bits.

b) Signal-To-Peak Noise Ratio (PSNR)

(a) MSE

The image quality estimation to be based on the pixel change was obtained by processing the $(PSNR)$ values and the (MSE). It is calculated by the following equation

$$Equation\ 33 \quad MSE = \sum_{i,j} (P(i,j) - C(i,j))^2$$

$(P(i,j))$ *; pixel of the clear image*

$(C(i,j))$*: pixel of the cypher image*

(b) PSNR

The signal-to-peak noise ratio, often abbreviated $PSNR$, is an engineering term for the ratio between a signal's maximum possible power and the power of distorted noise that affects the precision of its display. The $PSNR$ mathematical analysis of an image is given by the next equation

$$Equation34 \ \ PSNR = 20Log_{10}\left(\frac{I_{max}}{\sqrt{MSE}}\right)$$

For RGB color images, the definition of $PSNR$ is the same except that the MSE is the sum of all square value changes. In the alternative, for color images, the image is transcoded

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

into a separate color space and the $PSNR$ is displayed for each channel in that color space.

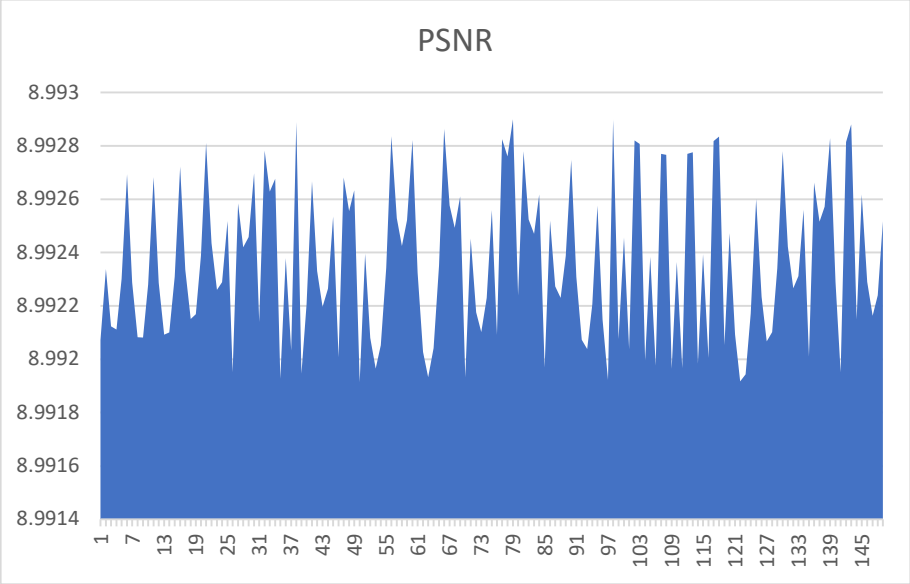


fIgure21: ¨PSNR of 150 images

All values returned from the $PSNR$ by our method are all in the range of residual values [8,99 8,993].

a) Speed analysis

For an evaluation of the execution time, our algorithm is tested on a personal computer "Intel core i5 3337 U 1.86 Gz CPU 8 GB ram. We use Matlab as programming software. We measure the encryption and decryption time of the tested images.

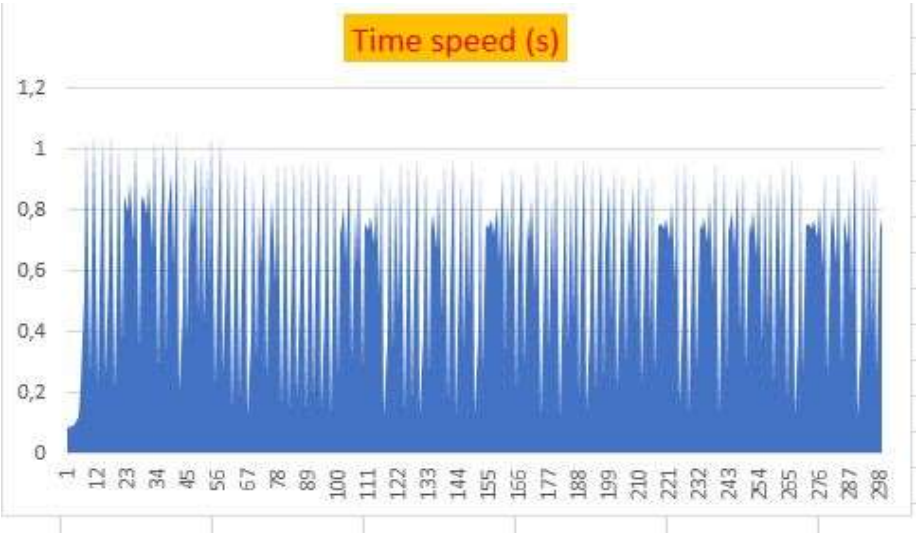


fIgure22: time of 300 images

4) Math Security

The first round eliminates any correlations and protects the system from differential attacks. Not knowing the size of the encryption matrix will increase the complexity of the attack. Without knowing the encryption key, it is difficult to replicate the structure of this matrix.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 19 April 2021 doi:10.20944/preprints202104.0507.v1

IV. CONCLUSION

The first encryption stage is provided by the deep enhancement of Vigenere's classic system, which can protect our new algorithm from any differential attack, while the second stage uses dynamic affine transformation, which consists of a large invertible matrix of uncertain size Provide without knowing the secret encryption keys, are difficult to construct from Kronecker's products, making the system resistant to any known attacks. The system only uses the replacement, which greatly reduces the execution time.

Conflict of Interest

I am the sole author of this article, and there are no private or public organizations or laboratories to fund my research, thus avoiding any expected conflicts.

This document does not contain any research or experiments conducted on animals.

References

[1] A.P.USiahaan "Genetic algorithm in Hill cipher encryption"international association of scientific innovation and research (IASR) vol 15,no,1 2016

[2] A.S alkhalid « cryptanalyse of Hill cipher using genetic algorithm" dalam IEEE hanmument 2015

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

[4] Imam Saputra, Mesran, Nelly Astuti Hasibuan3, "Vigenere Cipher Algorithm with Grayscale Image Key Generator for Secure Text File" International Journal of Engineering Research & Technology (IJERT), Vol. 6 Issue 01, January-2017

[5] Vaka Vamshi Krishna Reddy, Sreedhar Bhukya 2, "ENCRYPT AND DECRYPT IMAGE USING VIGENERE CIPHER", International Journal of Pure and Applied Mathematics, Volume 118 No. 24 2018

[6] A.P.U siahaan" three pass protocol in Hill cipher encryption technique" international journal of science and research (IJSR) vol 5 nà 7 2016 pp 1149-1152

[7] I Gede Arya Putra Dewangga, Tito Waluyo Purboyo, Ratna Astuti Nugrahaeni," A New Approach of Data Hiding in BMP Image Using LSB Steganography and Caesar Vigenere Cipher Cryptography" International Journal of Applied Engineering Research, Volume 12, Number 21 (2017) pp. 10626-10636

[8] Overbey.J.traversW and W ydylo J 2005 " On the key space of the Hill cipher" Cryptologia 29(1), 59-72

[9]. Saeednia,S 2000 "haow to make the Hill secure" Cryptologia 24(2), 353-360

[10]. Lin.C.H.Lee.C.Y and Lee.C.Yu 2004" comments on saeednia's improved scheme for the Hill cipher." Journal of the chineese institute of engineers 27/5, 743-746

[11] 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

[12] lYongWang all » A chaos-based image encryption algorithm with variable control parameters" Chaos, Solitons & Fractals Volume 41, Issue 4, 30 August 2009, Pages 1773-1783"

[13] Jan Sher Khan all, "Chaos based efficient selective image encryption" Multidimensional Systems and Signal Processing volume 30, pages943–961(2019)"

[14] H Li, Y Wang, Z Zuo - Optics and Lasers in Engineering, 2019 "Chaos-based image encryption algorithm with orbit perturbation and dynamic state variable selection mechanisms" Volume 115, April 2019, Pages 197-207

[15] Rongjun Ge all « A Novel Chaos-Based Symmetric Image Encryption Using Bit-Pair Level Process" July 8, 2019, date of current version August 7, 2019.

[16] Mohamed JarJar “Further improvement of the HILL method applied in image encryption” Procedia computer sciences 00(2019)000-000

[17] Shams Mahmoud Abd Ali » Novel Encryption Algorithm for Securing Sensitive Information Based on Feistel Cipher”Test engeenering managenment Page Number: 10 - 16 Publication Issue:19 Volume: 80 September-October 2019

[18] Zhi-hua Gan » A chaotic image encryption algorithm based on 3-D bit-plane permutation » Neural Computing and Applications (2019) 31:7111–7130

[19] Khan JS, Ahmad J. Chaos based efficient selective image encryption. Multidimensional Systems and Signal Processing. 2019; 30(2): 943-961.

[20] Liu H « Color image encryption based on one time keys and robust chaotics maps” Computer and mathematic application 59(2010),3320-3327

[21] Liu H « Image encryption using DNA complementary rule and chaotic maps” Applied soft computing 12(2012)1457-1466

[22] Ingyuan Wang » Image encryption algorithm for synchronously updating Boolean network based on matrix semi tensor product” Information sciences 507/2020 16-35

[23] Ismail IA, Amin M, Diab H. A digital image encryption algorithm based a composition of two chaotic logistic maps. IJ Network Security. 2010; 11(1): 1-10.