A New Visual Cryptography Method Based on the Profile Hidden Markov Model

Hikmetcan ÖZCAN, Fidan KAYA GÜLAĞIZ, Mehmet Ali ALTUNCU, Sümeyya İLKİN, Suhap ŞAHİN
Computer Engineering, Kocaeli University, Kocaeli, 41001, Turkey
mehmetali.altuncu@kocaeli.edu.tr

***Abstract*—Digital image capturing technologies and the internet are widely used today. These technologies make it very easy and fast to capture and share personal images in daily life. This causes difficulties in ensuring the confidentiality of private data and risks such as third persons getting hold of these data. The main goal of this study is to develop a user-friendly, powerful and effective method to encrypt digital images. For this aim, we propose a new block encryption method based on the Profile Hidden Markov Model. The method we propose consists of three main components. These are probability vector (PV), initialization vector (IV) and substitution-box (S-box). Encryption is in 24-bit blocks for color images and 8-bit blocks for grayscale images. The encryption rate in the proposed block encryption method is 0.7747 Mbit/s for color images and 1.0535 Mbit/s for the grayscale images. Theoretical analysis and experimental results confirm that the proposed encryption algorithm can provide high security both for color and grayscale images.**

***Index Terms*—ciphers, cryptography, hidden Markov models, performance analysis, Viterbi algorithm.**

I. INTRODUCTION

Visual cryptography is a cryptographic technique that represents a conversion of an image into a meaningless visual image by using mathematical models [1].

Visual cryptography methods consist of two sections; symmetric and asymmetric. Symmetric encryption is a one-key encryption type. The key used to encrypt the data is also used to decrypt it. Asymmetric encryption, unlike symmetric encryption, contains two keys, the public key, and the private key. There is a mathematical connection between these two keys. In asymmetric encryption, elliptic curve and fractal-based encryption methods are widely used. However, these encryption methods work slower than symmetric encryption methods in terms of encryption and decryption times. Symmetric encryption is also divided into two, as block ciphers and stream ciphers. A stream cipher is a symmetric key cipher where plaintext bits are combined with a pseudorandom cipher keystream using an exclusive-or (xor) operation. Block cipher algorithms are usually preferred for the encryption of non-streaming images [2-6]. Block cipher encryption methods enable fast and secure encryption and decryption through the use of mathematical models on the pixels of an image. There are many methods developed for block encryption; DES (Data Encryption Standard), AES (Advanced Encryption Standard), Blowfish, TEA (Tiny Encryption Algorithm), Image Fusion (IF), CB (Chaos-based), SBOX (Substitution-Box), ECB (Electronic Code Book), CBC (Cipher Block Chaining) are some of the examples [2-3], [7-11].

Many visual cryptography methods based on symmetric encryption have been developed from past to present. Thakur and Kumar [2] developed a simulation to compare the performance analyses of the DES, AES and Blowfish algorithms. They used the speed, block size and key size parameters to compare the performances. As a result of the study, it was concluded that Blowfish yielded better performance than the other encryption algorithms.

Salwa Kamal et al. [12] developed a new image encryption system by encrypting the original image with the key generated from fractal images. They used the same key for decryption and confirmed that the original image was obtained without any loss.

Ahmet et al. [10] developed an image encryption method using the Tiny encryption algorithm, which is a lightweight algorithm for JPEG standard images. The developed system was verified and tested on FPGA (Field Programmable Gate Array) in terms of hardware. The study demonstrated successful results both in terms of compression and encryption.

Erdal and Mehmet Ali [13] used the Knutt/Durstenfeld Shuffle (K/DSA) algorithm to encrypt digital images in their study. It aims to scramble image pixels using a key sequence generated by the proposed algorithm. It was concluded that a user-friendly, powerful and effective system on images of various types and features was obtained in the study.

Naveen et al. [14] proposed a 2-step method for ensuring the security of medical images. At the first phase, they compressed the image using EZW (Embedded Zero Wavelet). At the second phase, they converted the image into a matrix and performed a chaos-based row-column encryption algorithm. The authors stated that by applying these steps in reverse, they obtained the original image without any loss.

In his study, Erdal [8] used Dynamic S-boxes, which are widely used in block cipher encryption. He utilized the K/DSA algorithm in generating a dynamic S-box. He generated a reverse S-box for fast and lossless decryption using the S-box that he designed.

Reyad et al. [9] proposed two image encryption schemes for grayscale and color images. In the first scheme, they split the two-dimensional image into blocks. In the second scheme, they performed the encryption by subjecting all the blocks to XOR operation respectively. In the study, it was stated that the encrypted images were successful in terms of security analysis, sensitivity and stability.

Benssalah et al. [15] compared chaotic and elliptic-curve cryptography algorithms on medical images. It was emphasized that even though the encryption duration of the chaotic method was very short, in practice it required an

Digital Object Identifier 10.4316/AECE.2021.01003

intensive security analysis before it was performed. Whereas in the elliptic curve-based approach, it was stated that the encryption duration was long, despite the formidable discrete logarithm problem.

Rajat et al. [16] proposed a lattice-based secure cryptography system for communications in health systems in smart cities. In the proposed system, a public key is used to securely transmit the sensor data from the patient, such as blood sugar, blood pressure etc. to the doctor and a private key is used for the doctor to decipher this key. When the results of the proposed system were compared with similar studies in terms of communication and calculation costs, it was stated that the system yields more successful results.

Liu et al. [17] proposed a DNA-based probabilistic encryption method on remote-sensing images. They used DNA codes and two-dimensional logistic DNA mask during the encryption phase. In the experimental results, they stated that the proposed algorithm could resist various available spyware for remote-sensing images.

In this study, unlike the studies in the literature, an RGB probability vector is generated for the image to be encrypted using the Profile Hidden Markov Model (PHMM) method. Block cipher encryption is performed on image pixels by an IV generated by random values and a PV obtained through the PHMM method. Thus, the image is encrypted with the new PHMM-based symmetric encryption. Encryption is in 24-bit blocks for color images and 8-bit blocks for grayscale images. The primary benefits of the proposed method are described in four sections:

1. A new image encryption method based on PHMM was designed. In this design, RGB PV was obtained on the original image using PHMM. PV was used to update the IV during the encryption process.
2. IV was generated to increase the randomness of the encryption process and used to encrypt each line of the image.
3. Substitution-box (S-box) was used to increase reliability, quality and complexity of encryption.
4. It was observed that the encryption method that we proposed yielded more successful results compared to differential attacks, other cryptanalysis tests and the results of similar studies.

The following sections of the study are organized as follows. In the second part, the method proposed for image encryption is described in detail. In the third section, encryption and decryption processes on the data set used in similar studies are performed through experimental studies. In the fourth section, performance analysis tests of the proposed encryption method are presented. The article is concluded in the fifth section with the analysis of the results obtained.

II. Proposed Method

Within the scope of the study, a new method that performs encryption and decryption based on the CBC (Cipher Block Chain) encryption method is proposed. In the proposed method, the PV generated through the PHMM method is used.

A. Profile Hidden Markov Model (PHMM)

Profile Hidden Markov Model method was proposed by Krogh et al. [18] for aligning multiple sequences (at least three, biological sequences such as protein sequences). Proteins are amino acid sequences formed by the combination of multiple amino acids. The sequences used for alignment in PHMM are assumed to have a common origin and the connection between them is attempted to be revealed through a model [19-20].

PHMM basically has a start and end state. In addition to these states, three different states (insert, delete and match) can be used in each step. In these models, these three states are not required to exist at the same time. A model including all states of PHMM is given as an example in Fig. 1. I symbolizes the insert states, D symbolizes the delete states, M symbolizes the match states in Fig. 1.

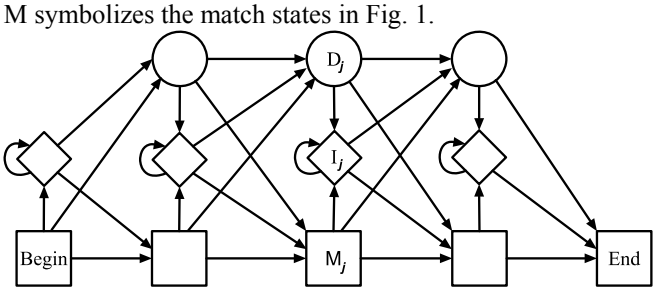


Figure 1. An example of PHMM with all state [20]

PHMM is mathematically expressed by 5 parameters $(Q,V,P(i),A,B)$. Here, $Q=\{q_1,q_2,q_3,\ldots,q_n\}$ represents the set of states, V represents the output alphabet, $P(i)$ represents the probability of being in q_i state at time t, A represents the transition probabilities set, a_{ij}^t represents the probability of being in q_j state at time $t+1$ when in q_i state at time t, B represents the set of output probabilities and $e_i^t(x)$ represents the probability of output x when in q_i state at time t [20]. The formula for transition and output probabilities to be calculated over any training data set is shown in (1) and (2), respectively [21]. The a_{ij} in (1) denotes the probability of transition from state q_i to state q_j and A_iJ expression denotes the number of transitions from state q_i to state q_j. J represents a specific state, whereas J' presents all probable states. The expression $e_i(x)$ in the (2) describes the probability of output x when it is in q_i state and $E_i(x)$ describes the number of occurrences of the output x when it is in q_i state.

$$a_{ij}=\frac{A_iJ}{\sum_{j'}A_iJ'} \tag{1}$$

$$e_i(x)=\frac{E_i(x)}{\sum_{x'}E_i(x')} \tag{2}$$

$$m,n \in N$$
$$i = 1,2,3,\ldots m \text{ and } j = 1,2,3,\ldots n$$
$$\alpha_{ij} \in RGB color$$

$$PM = \begin{bmatrix} \alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,n} \\ \alpha_{2,1} & \alpha_{2,2} & & \alpha_{2,n} \\ \vdots & & \ddots & \vdots \\ \alpha_{m,1} & \alpha_{m,2} & \cdots & \alpha_{m,n} \end{bmatrix} \quad (3)$$

$$\begin{cases} R \text{ if } R > G \,\&\, B, \\ G \text{ if } G > R \,\&\, B, \\ B \text{ otherwise.} \end{cases}, mr_{ij} \in \alpha_{ij} \quad (4)$$

$$MC = \begin{bmatrix} mc_{1,1} & mc_{1,2} & \cdots & mc_{1,n} \\ mc_{2,1} & mc_{2,2} & & mc_{2,n} \\ \vdots & & \ddots & \vdots \\ mc_{m,1} & mc_{m,2} & \cdots & mc_{m,n} \end{bmatrix} \quad (5)$$

$$MC_{e.g.} = \begin{bmatrix} R & G & \cdots & G \\ G & B & & B \\ \vdots & & \ddots & \vdots \\ G & R & \cdots & B \end{bmatrix}_{mxn}$$

In our study, PHMM is applied on the original image. Each pixel consists of Red(*R*), Green(*G*) and Blue(*B*) color channels. The dimensions of the image are assumed to be *m* px in height and *n* px in width. The set of states is $Q = \{M_1, M_2, \ldots, M_n\}$. In the study, the color lines of the image correspond to the amino acid sequences that will form the PHMM model. A rule was created to determine these sequences. According to the rule, one of the *R*, *G*, *B* values for the pixels in each line of the image was selected according to the maximum color value, and each of the lines obtained this way is considered as an amino acid sequence. Within the framework of this rule, we need to make our data set available before moving on to the PHMM model. Hence, it is necessary to determine the *R*, *G*, *B* value with the highest color value in each pixel of the image, to obtain the maximum color matrix. The steps of obtaining the matrix are mathematically expressed in (3-5) below. *PM* represents the pixel matrix, α_{ij} value represents a pixel, mr_{ij} represents the maximum color value of a pixel and *MC* represents the maximum color matrix with *m* number of rows and *n* number of columns. $MC_{e.g.}$ refers to an example for the maximum color matrix. The maximum color matrix is created by detecting the densest color channel of each pixel in the image. This matrix has great importance for the production of the probability vector. The probability vector is obtained by applying the Profile Hidden Markov Model on the maximum color matrix. This vector determines on which color channel the pixels will be encoded and how the initiation vector will be updated. This process, basically determines the complexity of the encryption. The complexity of encryption algorithm also refers to the strength of encryption algorithm.

In the PHMM method, amino acid sequences are used to establish the model. An example of an amino acid sequence can be expressed as "ACA-G-ATG". In our study, each row of the *MC* matrix (For example: "*RGBGGRRBR*" is considered as amino acid sequences used in the PHMM method. The best output vector to represent each image, namely the Red, Green, Blue sequences, can be obtained by using PHMM and the Viterbi algorithm.

Accordingly, the model will contain the same number of match states as the number of columns shown in Fig. 2. Each cell of the *MC* matrix has one of the Red, Green or Blue values. So, there are no columns with missing value. Therefore, it will not be necessary to use the delete state in the model which means that insert state will not be used either. For this reason, the model will contain the same number of match states as the number of columns, and when in match state, the next state will definitely be a transition. Therefore, the probability of transition between match states will be 1.

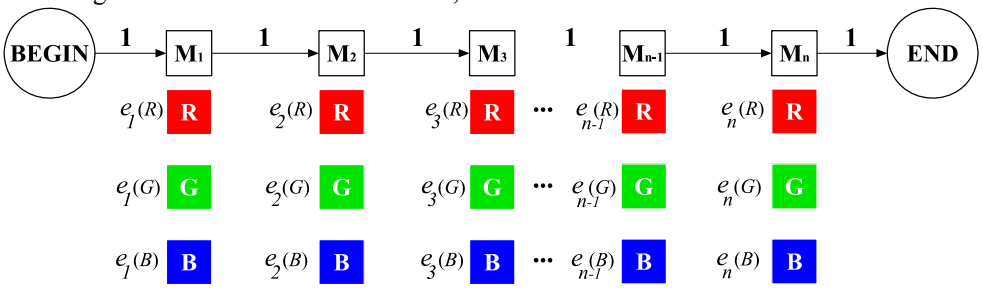


Figure 2. Transition and output possibilities in PHMM

After determining the states in the model, the probabilities of the outputs of each state must be calculated. Output probabilities are calculated by (2). In any state, there are three possible outputs (Red, Green or Blue). When calculating the output probabilities, the relevant column for each state will be taken into consideration and the sum of output probabilities for each state must be 1. In Fig. 2, the output probabilities obtained from a *MC* matrix are shown as an example.

The most probable state series that can be obtained through the model and the most probable output series that can occur in this state series can be obtained by using the Viterbi algorithm. In a way, this is the use of the Viterbi algorithm with the logic of finding the shortest path. Previous studies [22] have shown that this method can be used in this manner. The Viterbi algorithm designed to obtain both the state series and the most probable output

series that can occur in these states, is shown in (6). $\delta_t(j)$ value in the equation represents the path with the highest probability of ending in state j at time t. $\max_j(e_j)$ value represents the output with the highest probability that can be obtained in state j. The expression $\max_i\{\delta_{t-1}(i)a_{ij}\}$ represents to the path with the highest probability of ending in state i at time $t-1$ [20].

$$\delta_t(j) = \max_j(e_j)\max_i\{\delta_{t-1}(i)a_{ij}\} \quad (6)$$

The transition between the probabilities of the highest probable output that can occur at state i and the highest probable output that can occur at state $i+1$ is graphically shown in Fig. 3.

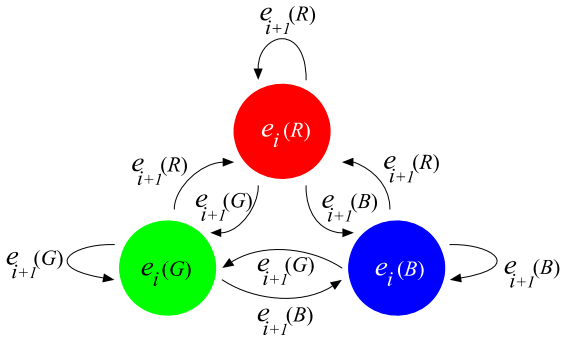


Figure 3. Transition states of outputs in PHMM

The color model with the highest probable outputs for the image after the Viterbi algorithm is obtained in Fig. 4.

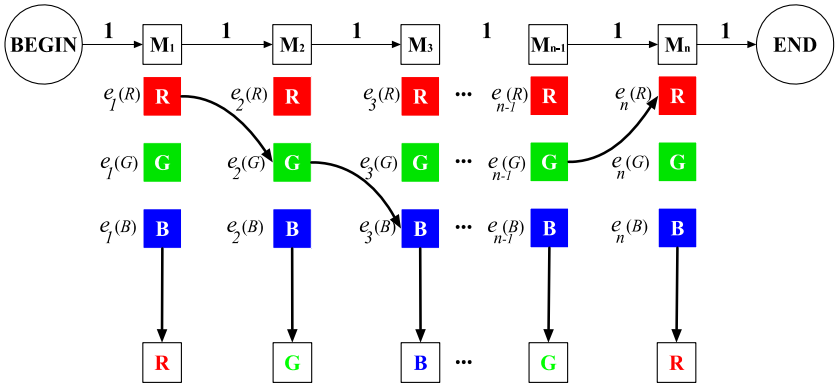
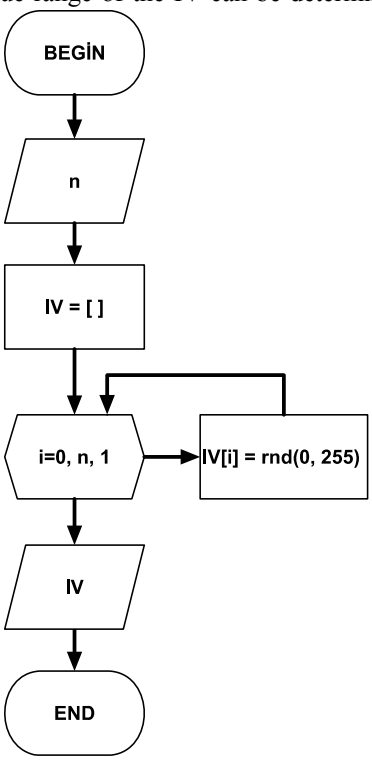


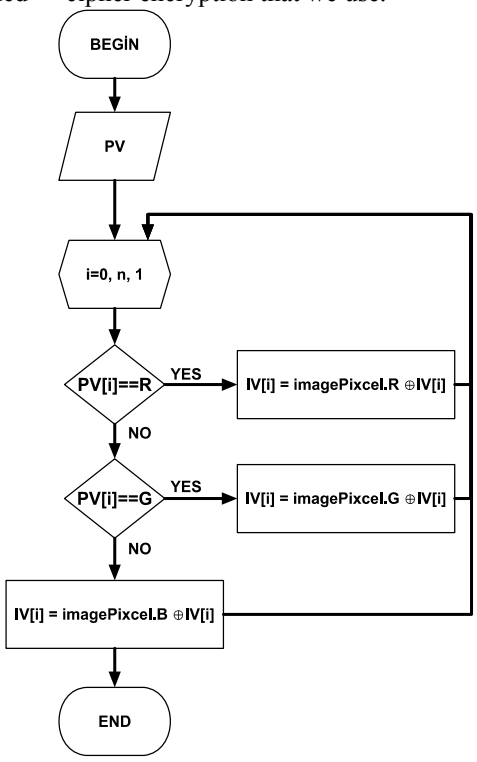
Figure 4. Finding the optimal path using the Viterbi algorithm

B. Initialization Vector (IV)

We can define the initialization vector (IV) as one of the components used to ensure randomness in encryption processes and to increase password security. Cipher Block Chain can be given as an example for algorithms using IV [23]. The size and value range of the IV can be determined according to the method to be used in encryption [24]. Since we use 8-bit colors in the color space in our study, the value range of IV for each element is between 0 and 255. The size of the image to be encrypted is specified as the width(n). The reason for this is to ensure that each column is encrypted with a different initialization value in the block cipher encryption that we use.



(a) Generate of IV



(b) Update of IV

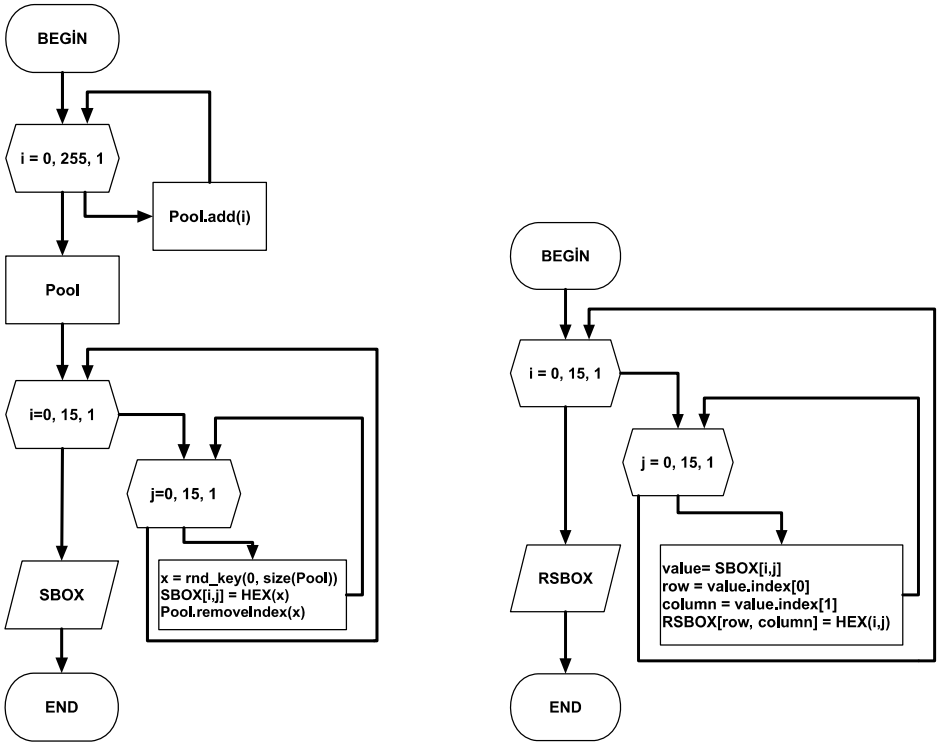
Figure 5. Initialization Vector operations

The flowchart in Fig. 5.a shows how IV is obtained before encryption. IV is updated after each line is encrypted to increase encryption security. Fig. 5.b shows the flow diagram of the update process. Update of the IV is performed according to PV. IV is updated according to the corresponding indexed element of the PV. According to the color in the PV, the color value is retrieved from the original image and subjected to XOR operation with the current value of the IV. In the result of this operation, IV has the new value. In the next line, it is encrypted with its new value.

C. Dynamic Substitution Box and Reverse Substitution Box(S-Box and Reverse S-Box)

Substitution Boxes are of great importance in image encryption and decryption. Therefore, generation of the suitable S-Boxes for encryption and decryption are only possible by having the correct key. At the same time, S-Boxes add scrambling feature to block cipher algorithms. [8].

The flow diagram that enables S-Box generation is shown in Fig. 6.a. According to this flowchart, S-Boxes consist of 16×16 dimensions and 256 integer values. All these integer values are gathered in a number pool. The integer value is found by selecting a random index value from this pool. This integer value is used by conversion into hexadecimal (16 numbers system). This is because the first value of the hexadecimal equivalent is used as the row, and the second value is used as the column. After an integer is used, it is removed from the pool. The sample S-Box obtained after each integer value in the pool is placed is shown in Fig. 7.a. Reverse S-Box is produced according to the S-Box. Fig. 6.b shows the flow diagram that enables the generation of the Reverse S-Box. According to this flowchart, all values starting from the first row and column of the S-Box table are used to be added in each row and column of the Reverse S-Box after each element of the table is separated as row and column. The sample Reverse S-Box obtained according to the S-Box in Fig. 7.a as the result of these operations is shown in Fig. 7.b.



(a) Generate of S-Box (b) Generate of Reverse S-Box

Figure 6. Dynamic Substitution Box operations

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	5	28	76	83	10	2F	A6	BE	62	E7	15	3C	6D	19	4D	79
1	6B	97	AB	AF	BB	4B	78	84	1C	70	77	E1	D8	B9	E0	74
2	41	46	38	33	B2	1	5C	4F	AD	88	68	FB	EA	B6	B1	4
3	49	53	48	C4	13	EE	44	C1	B5	F6	67	A2	BF	CB	E8	A9
4	8	9E	F9	24	DE	18	CA	64	36	5A	2	37	E4	80	98	6C
5	C7	72	95	89	6F	21	5E	55	0	86	8B	C2	9F	C0	94	E9
6	9C	C6	F4	42	BC	8A	25	2D	8D	E5	82	C3	ED	E	DA	31
7	58	FE	DF	43	2B	5D	39	EB	3B	A3	1F	5F	99	D	D1	7F
8	CE	1D	9D	23	8E	50	B4	35	4A	65	FC	91	A4	AA	6A	F5
9	A8	47	57	A1	DB	3F	BA	2E	7D	A	9	14	66	D2	A0	7C
A	16	6	F3	D4	73	BD	B3	F	4E	AE	3	D6	63	C5	45	9A
B	32	2A	CD	51	C9	29	B0	85	71	61	B7	81	40	A5	E3	8C
C	DC	7E	75	9B	D0	AC	7B	FA	A7	3A	3E	22	34	F2	E6	DD
D	D5	5B	F0	20	C	4C	D7	F7	54	69	56	11	B	26	D9	C8
E	EC	87	7	F8	2C	B8	F1	90	96	CF	92	52	EF	59	1B	6E
F	7A	E2	CC	D3	8F	3D	27	30	93	1E	17	1A	60	FD	FF	12

(a) S-Box

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	58	25	4A	AA	2F	0	A1	E2	40	9A	99	DC	D4	7D	6D	A7
1	4	DB	FF	34	9B	A	A0	FA	45	D	FB	EE	18	81	F9	7A
2	D3	55	CB	83	43	66	DD	F6	1	B5	B1	74	E4	67	97	5
3	F7	6F	B0	23	CC	87	48	4B	22	76	C9	78	B	F5	CA	95
4	BC	20	63	73	36	AE	21	91	32	30	88	15	D5	E	A8	27
5	85	B3	EB	31	D8	57	DA	92	70	ED	49	D1	26	75	56	7B
6	FC	B9	8	AC	47	89	9C	3A	2A	D9	8E	10	4F	C	EF	54
7	19	B8	51	A4	1F	C2	2	1A	16	F	F0	C6	9F	98	C1	7F
8	4D	BB	6A	3	17	B7	59	E1	29	53	65	5A	BF	68	84	F4
9	E7	8B	EA	F8	5E	52	E8	11	4E	7C	AF	C3	60	82	41	5C
A	9E	93	3B	79	8C	BD	6	C8	90	3F	8D	12	C5	28	A9	13
B	B6	2E	24	A6	86	38	2D	BA	E5	1D	96	14	64	A5	7	3C
C	5D	37	5B	6B	33	AD	61	50	DF	B4	46	3D	F2	B2	80	E9
D	C4	7E	9D	F3	A3	D0	AB	D6	1C	DE	6E	94	C0	CF	44	72
E	1E	1B	F1	BE	4C	69	CE	9	3E	5F	2C	77	E0	6C	35	EC
F	D2	E6	CD	A2	62	8F	39	D7	E3	42	C7	2B	8A	FD	71	FE

(b) Reverse S-Box

Figure 7. Dynamic Substitution Box example

In the encryption process, the Red, Green and Blue values in each pixel of the original image are processed through the S-Box table one by one. Operations are performed by taking the hexadecimal equivalent of each color value. To give an example on the S-Box in Fig. 7.a and the Reverse S-Box in Fig. 7.b; let the color values of a pixel that we wish to encrypt be R=211, G=167, B=136 and their hexadecimal equivalents will be R=D3, G=A7 and B=88 respectively. For the R value in the S-Box table, the value in row D and column 3th is the new value R=HexToDecimal(20)=32. For G value, the value of row A and column 7th is the new value G=HexToDecimal(0F)=15. For the B value, the value in the 8th row and 8th column is the new value B=HexToDecimal(4A)=74. Accordingly, the pixels in the encrypted line are calculated as R=32, G=15, B=74 respectively. In the decryption process, as in the encryption process, all pixels are retrieved from the Reverse S-Box table. If we continue with the same example, the operations are repeated for the encrypted pixel values, R=32, G=15 and B=74 according to the Reverse S-Box table. For the R value, R=DecimalToHex(32)=20 the value of the 2nd row and the 0th column is D3 and its decimal equivalent is 211. For G value, G=DecimalToHex(15)=0F the value of row 0th and column F is A7 and its decimal equivalent is 167. For the B value, B=DecimalToHex(74)=4A the value of row 4th and column A is 88 and its decimal equivalent is 136. As a result of this process, it is confirmed that the color values obtained from the Reverse S-Box are the same as the original images color values. At the same time, all scrambling operations on the S-Box do not place a significant burden on the encryption process.

D. Proposed Block Chipper Method Mode Encryption

The purpose of developing the encryption scheme is to ensure that the image is encrypted and highly resistant to cryptanalysis attacks. The block-based encryption method proposed in the study consists of three basic steps:

1) Establishing the model of the image with the PHMM method;
2) Obtaining the output sequence with the highest probability through the PHMM model;
3) Listing the resulting output series as the PV of block encryption.

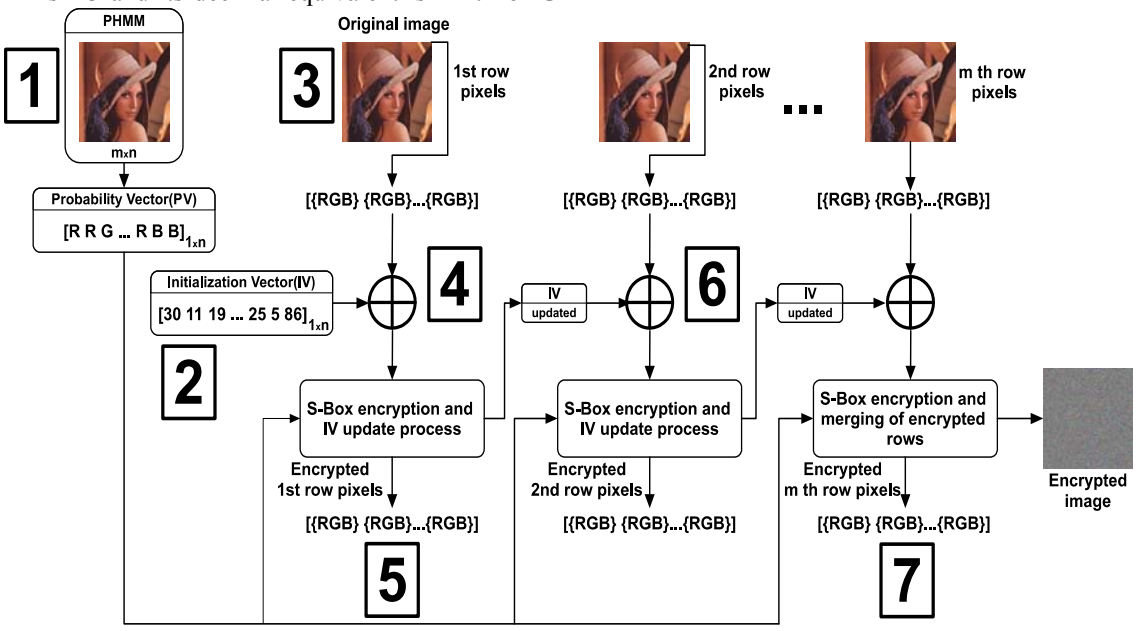


Figure 8. Encryption scheme

First of all, the series that will be used to generate the model are obtained for each line of the image by selecting the color with the highest value among the red, green and blue values for each pixel of the image to be encrypted. The model created to encrypt images contains only the match state. Since the image pixels contain three different color values, there are three possible outputs in each state. These are denoted by the letters Red (*R*), Green (*G*), Blue (*B*). After the model is established, the best output series that can be obtained from the model is determined through the use of the Viterbi algorithm. It has been demonstrated in previous studies [20], [22-25] that the Viterbi algorithm can be used to find the best possible path, as the Dijkstra algorithm. The mathematical expression showing the use of the Viterbi algorithm for this purpose is shown in (6).

The output series obtained through the Viterbi algorithm is used as the PV for block cipher encryption. Then, the image is encrypted by block cipher encryption. In Fig. 8, the steps in the block cipher encryption process are modelled, and they are listed in detail below:

1) Obtaining the PV from the original image;
2) Obtaining the IV;
3) Encrypting the pixel values in the first line of the original image with IV through XOR;
4) Scrambling the encrypted line according to the S-Box table;
5) Updating the IV by PV;
6) Repeating from step 3 until the pixels of the last line of the original image are encrypted;
7) Merging the lines to obtain the encrypted image.

E. Proposed Block Chipper Method Mode Decryption

As a result of the proposed block-based encryption method, we obtained the PV, IV, S-BOX and the encrypted image output. These four components are used to obtain the original picture without any loss. In case any of these

components are missing or incorrect, the decryption process ends up with meaningless data. In Fig. 9, the decryption process is modelled, and steps are listed in detail below:

1) Preparing the Reverse S-Box table generated through the encrypted image obtained by the proposed block-based encryption method, IV, PV and S-Box, for decryption;
2) Reversing the first line pixels in the encrypted image according to the Reverse S-Box table;
3) Decrypting the line pixels obtained from Reverse S-Box with IV through XOR operation;
4) Updating IV according to PV;
5) Repeating from step 2 until the pixels of the last line of the original image are decrypted;
6) After unlocking all lines, the lines are combined to obtain the original image. After all lines are decrypted, merging the lines to obtain the original image.

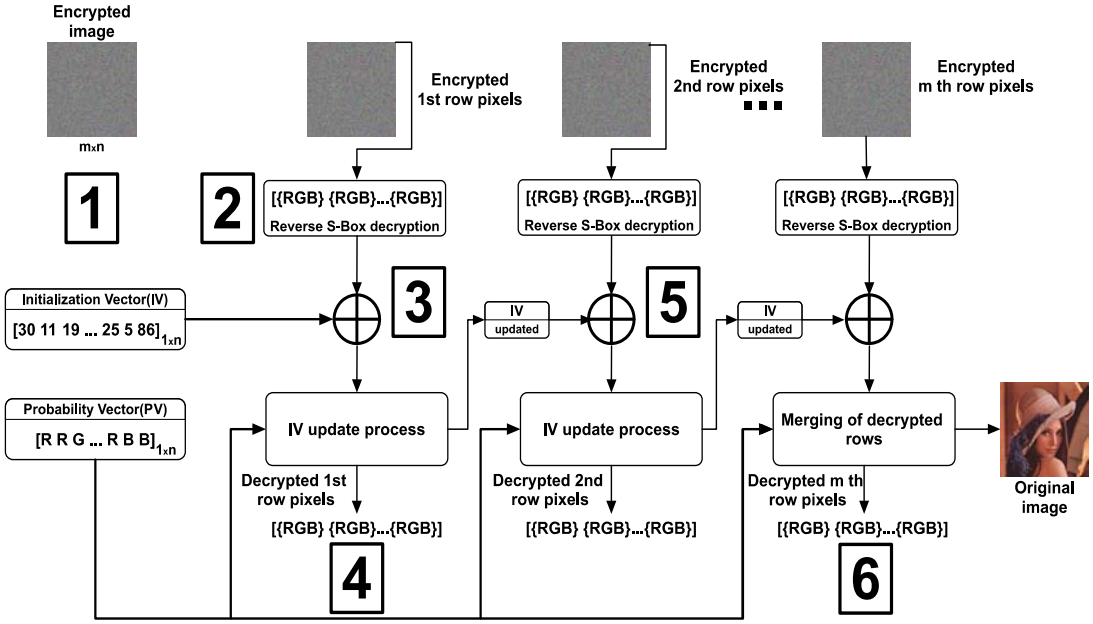
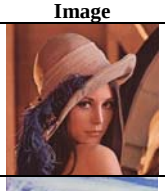
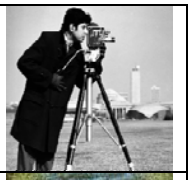
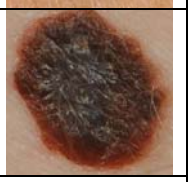
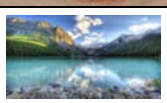


Figure 9. Decryption scheme

III. Experimental Study

The data set that will be used to test the proposed encryption method consists of Lena, Airplane, Cameraman and Baboon images which are used in many similar studies. In addition, two different melanoma images acquired from the open access dataset within the framework of the International Skin Imaging Collaboration (ISIC) Melanoma Project are also added to the dataset [26]. In order to test the performance of the proposed method on very high-resolution images, three different images with 2K, 4K and 8K resolution were added to the test dataset. These images are named in the article as 2K Image, 4K Image and 8K Image. In order to be able to make comparisons with similar studies, color images are used in the dataset being converted to grayscale. The images used in the data set are in .jpeg, .png and .bmp formats. Detailed information on the data set and the images are shown in Table I.

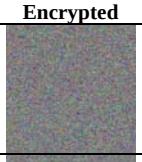
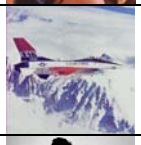
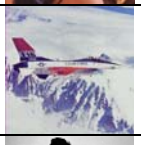
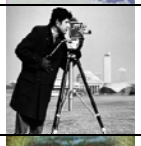
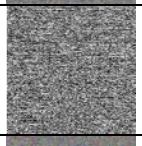
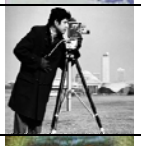
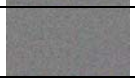
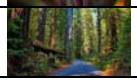
Table I. Dataset used for Image Encryption

Image Name	Image Features	Image
Lena	256x256, .jpg, 12 KB, Color, Gray BT-709(HDTV) Method (0.21Red+0.72Green+0.07Blue)	
Airplane	512x512,.bmp, 192 KB, Color, Gray BT-709(HDTV) Method (0.21Red+0.72Green+0.07Blue)	
Cameraman	256x256, .png, 42 KB	
Baboon	512x512, .jpg, 59 KB, Color, Gray BT-709(HDTV) Method (0.21Red+0.72Green+0.07Blue)	
Melanoma 1	512x512, .jpg, 54 KB, Color, Gray BT-709(HDTV) Method (0.21Red+0.72Green+0.07Blue)	
Melanoma 2	512x512, .jpg, 53 KB, Color, Gray BT-709(HDTV) Method (0.21Red+0.72Green+0.07Blue)	
2K Image	1920x1080, .jpg, 4.84 MB, Color, Gray BT-709(HDTV) Method (0.21Red+0.72Green+0.07Blue)	
4K Image	3840x2160, .jpg, 17.3 MB, Color, Gray BT-709(HDTV) Method (0.21Red+0.72Green+0.07Blue)	
8K Image	7680x4320, .jpg, 94.5 MB, Color, Gray BT-709(HDTV) Method (0.21Red+0.72Green+0.07Blue)	

IV. Performance Analysis

Original and encrypted images were subjected to image cryptanalysis to evaluate the encryption quality. For this purpose, twelve different analysis methods were utilized. Detailed descriptions of the methods are presented in the subsections. Also, test results for encryption and decryption process of the proposed algorithm are shown in Table II.

Table II. Proposed Encryption and Decryption Method Test Results

Image Name	Original	Encrypted	Decrypted
Lena			
Airplane			
Cameraman			
Baboon			
Melanoma 1			
Melanoma 2			
2K Image			
4K Image			
8K Image			

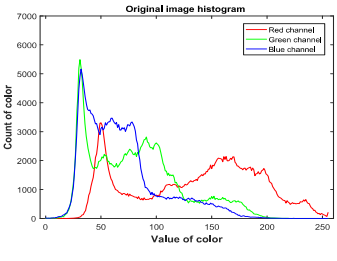
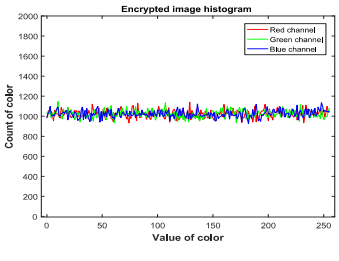
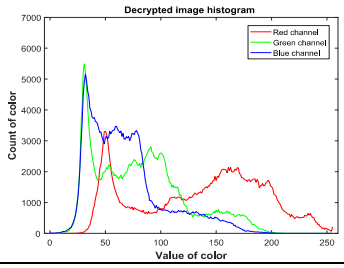
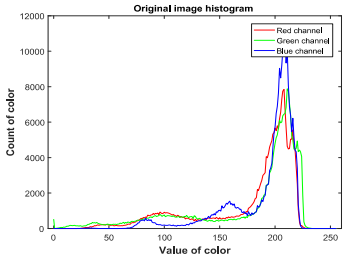
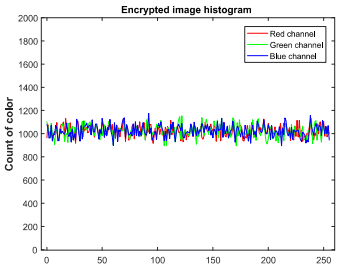
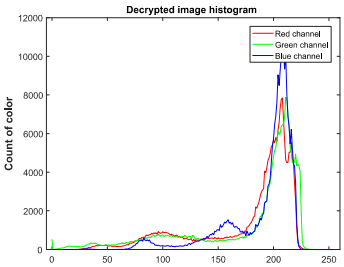
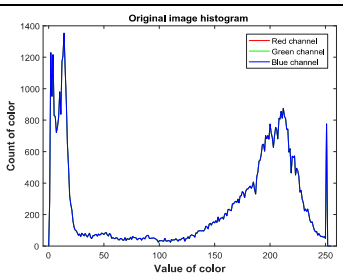
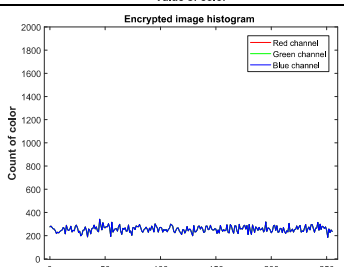
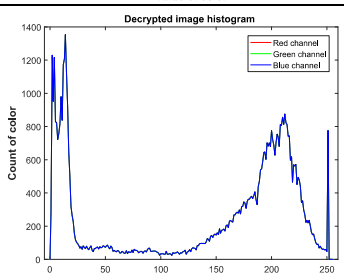
A. Histogram Analysis

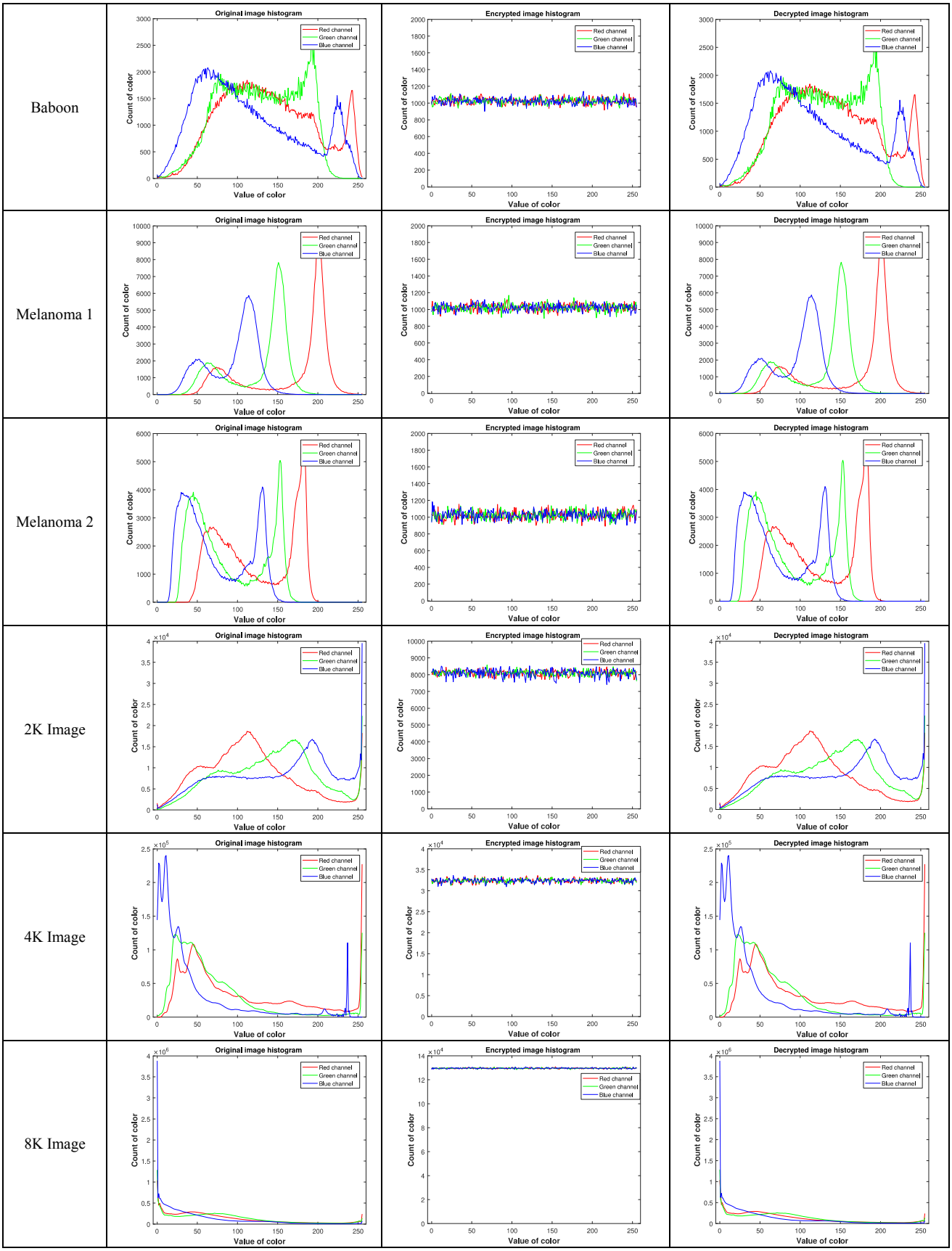
Encrypted text histogram analysis is one of the simplest methods to demonstrate the quality of image encryption. Since a good image encryption method tends to encrypt a plain text image in a random manner, it is desirable to see a properly distributed histogram for the encrypted text image [27]. The results of the histogram analysis for the encryption are shown in Table III. It can be seen in the result of histogram analysis that the pixel values in the encrypted images are distributed homogeneously. Therefore, it can be concluded that it is resistant against statistical attacks.

B.Mean Absolute Error (MAE)

The mean absolute error evaluates whether there is a difference in image quality between the original image and the decrypted image. A mean absolute error value greater than zero indicates a difference in quality of the image and zero indicates that there is no difference in the quality of the image [28].

Table III. Histogram Analysis Results

Image Name	Original	Encrypted	Decrypted
Lena			
Airplane			
Cameraman			



The formula for obtaining the mean absolute error value is shown in (7) [28]. H and W represent the row and the column of the image, respectively. $C(i, j)$ represents the pixel value of row i and column j in the encrypted image. $P(i, j)$ represents the pixel value of row i and column j in the original image.

$$MAE = \frac{1}{H \times W} \sum_{i=0}^{H-1} \sum_{j=0}^{W-1} \left| C(i, j) - P(i, j) \right| \tag{7}$$

When the MAE results are analyzed, the results are zero for all images. This demonstrates that there is no difference in quality between the original image and the decrypted image.

C.Mean Square Error (MSE)

The mean square error between the original image and the decrypted image is calculated to see whether it caused distortions on the image. If the mean square error is greater than zero, it indicates that there is a distortion in the pixels of the picture, and zero indicates that the picture can be reversed without any loss [29]. The formula for obtaining the mean square error value is shown in (8) [29]. *M* and *N* represent the row and the column of the image, respectively. $C(i,j)$ represents the pixel value of row i and column j in the encrypted image. $P(i,j)$ represents the pixel value of row i and column j in the original image.

$$MSE = \frac{1}{M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} \left\| C(i,j) - P(i,j) \right\|^2 \tag{8}$$

When the MSE results are analyzed, the results are zero for all images. This demonstrates that there is no difference in quality between the original image and the decrypted image.

D.Peak Signal-to-Noise Ratio (PSNR)

PSNR is one of the methods used for objective image quality measurement on similar images. In this study, it is used to find out the difference in image quality between the original image and the decrypted image. A higher PSNR value indicates a higher image quality, whereas a low PSNR value indicates high numerical difference between images [30]. In (9) shows the calculation of the PSNR value. *MAX* represents the maximum pixel value used, and *MSE* represents the mean square error value [29]. In order to determine the PSNR values, the original image and the encrypted image are converted to grayscale ($h = 8$ bit) and the mean square error value is calculated over the values. For this reason, *MAX* is specified as $MAX = 2^h - 1 = 255$.

$$PSNR = 10 \log_{10} \left(\frac{MAX^2}{MSE} \right) \tag{9}$$

When the PSNR results are analyzed, the results are minus infinity for all images. This demonstrates that there is no difference in quality between the original image and the decrypted image.

E. Structure Similarity (SSIM)

The structural similarity test shows to what extent two images are similar to each other. A result close to zero indicates that the similarity is low, a result close to one indicates their similarity is quite high and a result of one indicates that the images are identical [31]. In (10-14), μ_x represents the average of x values, μ_y represents the average of y values, σ_{xy} represents the covariance of x and y values, σ_x^2 and σ_y^2 represent the variances of x and y values respectively, C_1 and C_2 represent the values used to balance the division with weak denominator. K_1 and K_2 represent constant values. $K_1 = 0.01$ and $K_2 = 0.03$ values are used. L denotes the dynamic range of pixel values (255 for 8-bit each color). In this study, the value of $L = 2^8 - 1 = 255$ is used [31-32].

$$\begin{aligned} C_1 &= (K_1 L)^2, \\ C_2 &= (K_2 L)^2. \end{aligned} \tag{10}$$

$$\begin{aligned} \mu_x &= \frac{1}{n} \sum_{i=1}^{n} x_i, \\ \mu_y &= \frac{1}{n} \sum_{i=1}^{n} y_i. \end{aligned} \tag{11}$$

$$\begin{aligned} \sigma_x &= \left(\frac{1}{n-1} \sum_{i=1}^{n} (x_i - \mu_x)^2 \right)^{\frac{1}{2}}, \\ \sigma_y &= \left(\frac{1}{n-1} \sum_{i=1}^{n} (y_i - \mu_y)^2 \right)^{\frac{1}{2}}. \end{aligned} \tag{12}$$

$$\sigma_{xy} = \frac{1}{n-1} \sum_{i=1}^{n} (x_i - \mu_x)(y_i - \mu_y) \tag{13}$$

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)} \tag{14}$$

The structural similarity results of the original image (P), the encrypted image (E) and the decrypted image (D) in our study are presented in Table IV.

TABLE IV. SSIM TEST RESULTS

Image Name	P between E	P between D
Lena	0.0101	1.0000
Airplane	0.0122	1.0000
Cameraman	0.0083	1.0000
Baboon	0.0031	1.0000
Melanoma 1	0.0053	1.0000
Melanoma 2	0.0066	1.0000
2K Image	0.0064	1.0000
4K Image	0.0061	1.0000
8K Image	0.0054	1.0000

Test results show that the structural similarity between the original image and the encrypted image is close to zero. In addition, the structural similarity of the original and the decrypted image is recorded as 1. This result indicates that the images are identical.

F. Information Entropy (IE)

Information entropy analysis is a method used to measure the level of uncertainty in encrypted data. Uncertainty is the situation that makes the difference in data and is unknown to third parties. The more uncertainty in the encrypted data, the more difficult it is to approximate the original image. The accepted information entropy value for image encryption is specified as 8. The calculation of the uncertainty value is shown in (15) [28]. In the equation $P(s_i)$ represents the probability of using the color value (e.g. *R*=32).

$$H(s) = -\sum_{i=0}^{2^n - 1} P(s_i) \log_2 \left[P(s_i) \right] \tag{15}$$

Test results of IE used to measure the complexity of the encrypted image in our study are shown in Table V. (*) sign shows the results obtained when the proposed block cipher

encryption method is used.

TABLE V. IE TEST RESULTS

Image Name	Image Type	Red	Green	Blue	Gray
Lena*	Original	7.5805	7.1336	6.8448	7.2238
	Encrypted	7.9989	7.9990	7.9990	7.9987
	Decrypted	7.5805	7.1336	6.8448	7.2238
Airplane*	Original	6.7544	6.8163	6.2799	6.7544
	Encrypted	7.7219	7.7443	7.7032	7.9968
	Decrypted	6.7544	6.8163	6.2799	6.7544
Cameraman*	Original	7.2074	7.2074	7.2074	7.2074
	Encrypted	7.9927	7.9927	7.9927	7.9927
	Decrypted	7.2074	7.2074	7.2074	7.2074
Baboon*	Original	7.7066	7.4744	7.7522	7.3812
	Encrypted	7.9991	7.9993	7.9991	7.9992
	Decrypted	7.7066	7.4744	7.7522	7.3812
Melanoma 1*	Original	6.6979	6.6238	6.6747	6.6412
	Encrypted	7.9989	7.9988	7.9989	7.9977
	Decrypted	6.6979	6.6238	6.6747	6.6412
Melanoma 2*	Original	6.9630	6.8892	6.8347	6.8755
	Encrypted	7.9983	7.9983	7.9984	7.9974
	Decrypted	6.9630	6.8892	6.8347	6.8755
2K Image*	Original	7.6989	7.7469	7.8386	7.6816
	Encrypted	7.9997	7.9997	7.9994	7.9997
	Decrypted	7.6989	7.7469	7.8386	7.6816
4K Image*	Original	7.5416	7.0954	6.6616	7.4416
	Encrypted	7.9998	7.9998	7.9998	7.9998
	Decrypted	7.5416	7.0954	6.6616	7.4416
8K Image*	Original	7.4595	7.4892	6.8090	7.4494
	Encrypted	7.9999	7.9999	7.9999	7.9999
	Decrypted	7.4595	7.4892	6.8090	7.4494
Lena [33]	Original	-	-	-	-
	Encrypted	-	-	-	7.9977
	Decrypted	-	-	-	-
Lena [34]	Original	7.4451	7.4451	7.4451	7.4451
	Encrypted	7.7333	7.7333	7.7333	7.7333
	Decrypted	7.4451	7.4451	7.4451	7.4451
Baboon [34]	Original	7.1839	7.1839	7.1839	7.1839
	Encrypted	7.7289	7.7289	7.7289	7.7289
	Decrypted	7.1839	7.1839	7.1839	7.1839
Aerial 1 [17]	Original	-	-	-	7.3424
	Encrypted	-	-	-	7.7289
	Decrypted	-	-	-	-
Aerial 3 [17]	Original	-	-	-	3.8595
	Encrypted	-	-	-	7.9993
	Decrypted	-	-	-	7.0000

When the IE results are analyzed, it can be seen that the entropy results of the images used in our data set after the encryption are very close to the ideal entropy value of 8. Therefore, it can be concluded that the randomness and irregularity of the encrypted pictures are at the desired level. It can be clearly seen that better results are achieved in our study, compared to the information entropy analysis results of the methods suggested in [17], [33-34].

G. Differential Attack

Differential attack [35] is an efficient method to use pairs of plain-images related by a constant difference and compare the difference of the corresponding cipher-image for statistical patterns in their distribution. Usually, attackers use the encryption algorithm, making certain minor changes to plain images. They try to find the relationship between the plain and the encrypted image by observing the changes in the encrypted image. This analysis aims to assess sensitivity to the plain-image [17]. Number of pixels change rate (NPCR) and unified average changing intensity (UACI) are two methods used to measure the difference between encrypted images.

The number of pixels change rate is obtained by comparing the pixels in the original image and the encrypted image. It has a ratio between 0 and 100. The value of zero indicates that the pictures are identical and the value of 100 indicates that the pixels are completely changed [36]. How to obtain the NPCR value is presented in (16). H and W represent the row and the column of the image, respectively. C_1 and C_2 represent the two pictures. T indicates whether the pixels between the two images are equal.

$$NPCR = \frac{\sum_{i,j} T(i,j)}{H \times W} \times 100\% \tag{16}$$

$$T(i,j) = \begin{cases} 0, & C_1(i,j) = C_2(i,j), \\ 1, & C_1(i,j) \neq C_2(i,j). \end{cases} \tag{17}$$

In our study, the changing pixel rates between the original image and the encrypted image are shown in Table VI. The test results are interpreted as pass and fail, pursuant to the NPCR test results in the study of Wu et al. [36].

TABLE VI. NPCR TEST RESULTS

Image Name	NPCR [%]	According to [36]
Lena*	100.0000	Pass
Airplane*	99.9996	Pass
Cameraman*	99.6200	Pass
Baboon*	100.0000	Pass
Melanoma 1*	100.0000	Pass
Melanoma 2*	100.0000	Pass
2K Image*	99.9974	Pass
4K Image*	99.9997	Pass
8K Image*	99.9847	Pass
Lena [33]	99.6323	Pass
Lena [34]	99.1001	Fail
Baboon [34]	99.4200	Fail
Aerial 1 [17]	99.6235	Pass
Aerial 3 [17]	99.6010	Pass

(*) sign shows the results obtained when the proposed block cipher encryption method is used. When the test results are analyzed, it can be seen that all pixels of Lena, Baboon, Melanoma 1 and 2 are 100% changed, while Airplane and Cameraman are very close to 100%. In addition, all the images in our dataset successfully passed the tests compared to [36] (NPCR=99.5710%). This shows us that the algorithm proposed in our study resistant to differential attacks. It can also be observed that [17], [33] have successfully passed the NPCR test, whereas [34] has failed the NPCR test. It can be clearly seen that our method has better results, compared to the NPCR analysis results of the methods suggested in [17], [33].

UACI measures the average changed intensity between the original image and the corresponding encrypted image. A UACI value with high intensity strongly resists differential attack [36].

$$UACI = \frac{1}{H \times W} \sum_{i,j} \frac{|C_1(i,j) - C_2(i,j)|}{255} \times 100\% \tag{18}$$

In our study, the changes between the original image and the encrypted image are shown in Table VII. The test results are interpreted as pass and fail, pursuant to the UACI test results in the study of Wu et al. [36].

TABLE VII. UACI TEST RESULTS

Image Name	UACI [%]	According to [36]
Lena*	34.1192	Pass
Airplane*	33.5091	Pass
Cameraman*	36.7897	Pass
Baboon*	33.5512	Pass
Melanoma 1*	34.7576	Pass
Melanoma 2*	33.7174	Pass
2K Image*	34.0016	Pass
4K Image*	33.7884	Pass
8K Image*	35.9451	Pass
Lena [33]	34.5960	Pass
Lena [34]	33.2129	Fail
Baboon [34]	33.2791	Fail
Aerial 1 [17]	33.3371	Fail
Aerial 3 [17]	33.4765	Pass

(*) sign shows the results obtained when the proposed block cipher encryption method is used. When the test results are analyzed, all the images in our data set have successfully passed the tests compared to [36] (256×256; UACI=33.2255%, 512×512; UACI=33.3445%). According to the results of the differential attack, it is proved once again that the proposed algorithm is more resistant with regards. It can also be seen that Lena in [33] and Aerial 3 in [17] have successfully passed the UACI test, while Lena in [34] and Aerial 1 in [17] have failed the UACI test. It is clearly seen that better results are obtained compared with the UACI analysis results of the methods proposed in [17], [33-34].

H.Correlation Analysis (Pearson Correlation)

In decrypted images, there is a binary correlation between adjacent image pixels. As the result of correlation analysis, whether there is a linear relationship, and the degree of this relationship if any, is calculated by the correlation coefficient (CC). Correlation coefficient takes a value between -1 and +1. If the correlation coefficient is -1 there is an absolute negative linear relationship, if it is +1 here is an absolute positive linear relationship, if it is 0 there is no relationship between the two variables. If there is a correlation in the encrypted images, it can be used by unauthorized persons to partially or fully restore the image. Horizontal, vertical and diagonal-correlation coefficient (Pearson Correlation [37]) between adjacent pixels of an image can be mathematically expressed as in (19-22) [38]. Here, x and y represent adjacent pixel sequence in three directions, and z represents the total number of adjacent pixels selected from the image. $M(x)$ denotes the mean of x. $D(x)$ denotes the variance of x.

$$M(x)=\frac{1}{z}\sum_{i=1}^{z}x_i, \tag{19}$$

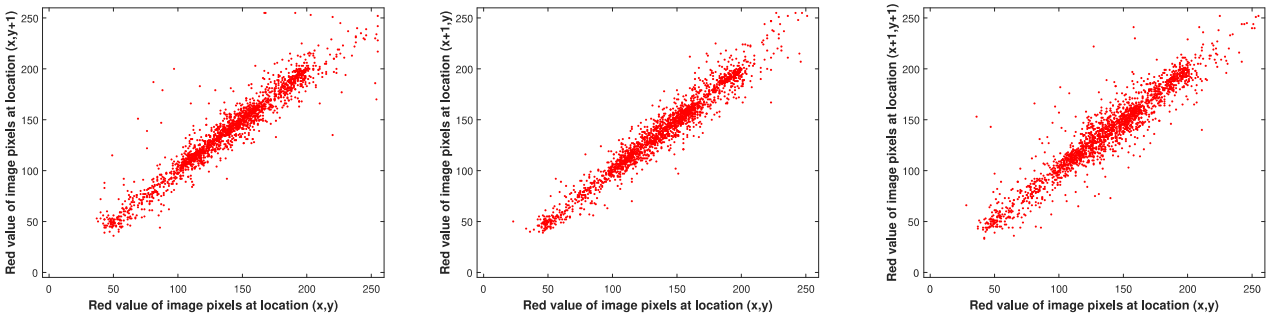
$$D(x)=\frac{1}{z}\sum_{i=1}^{z}\left[x_i-M(x)\right]^2, \tag{20}$$

$$Conv(x,y)=\frac{1}{z}\sum_{i=1}^{z}\left[x_i-M(x)\right]\left[y_i-M(y)\right], \tag{21}$$

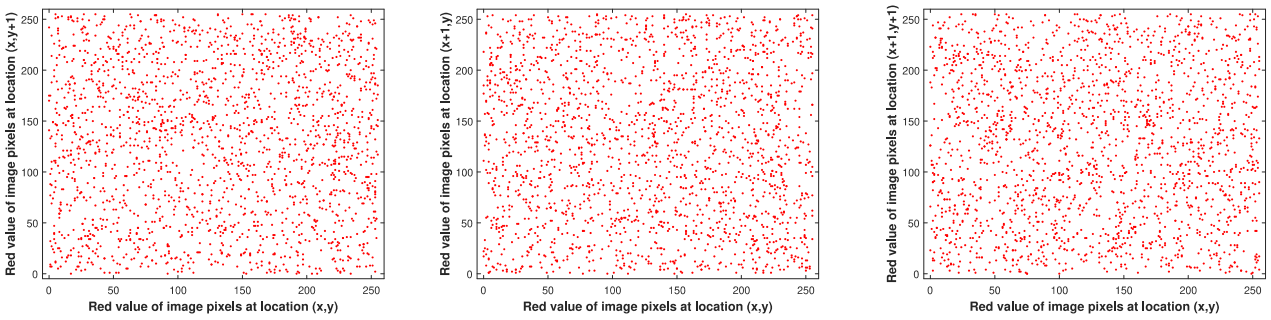
$$\gamma_{xy}=\frac{Conv(x,y)}{\sqrt{D(x)}\sqrt{D(y)}}. \tag{22}$$

In our study, 2000 horizontal, vertical and diagonal pixel adjacencies on the original and encrypted Lena images are randomly selected and correlation analysis was performed. Correlation analysis was performed separately for each color channel. Fig. 10 show the distribution of the correlation coefficients for the red color channel, the green color channel and the blue color channel respectively.

Correlation analysis was performed for the original and encrypted versions of all the all images in our data set. Table VIII shows the horizontal, the vertical and the diagonal correlation coefficients in detail. (*) sign shows the results obtained when the proposed block cipher encryption method is used. In addition, P denoted for plain image and C denoted for cipher image.



(a)



(b)

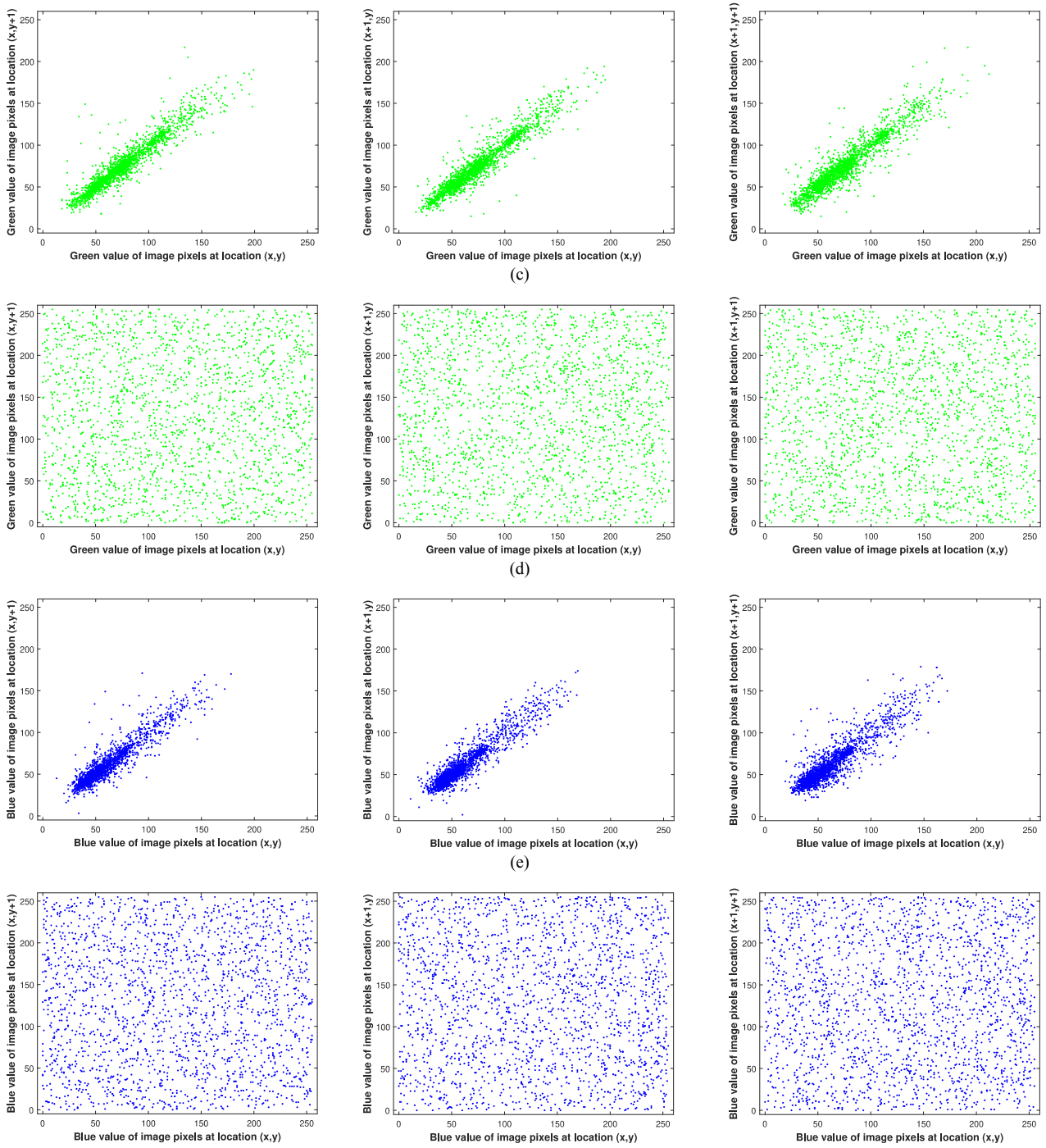


Figure 10. Distribution of correlation coefficients: (a) The red color distribution of the horizontal, vertical and diagonal correlation coefficients of adjacent pixels of plain image. (b) The red color distribution of the horizontal, vertical and diagonal correlation coefficients of adjacent pixels of cipher image. (c) The green color distribution of the horizontal, vertical and diagonal correlation coefficients of adjacent pixels of plain image. (d) The green color distribution of the horizontal, vertical and diagonal correlation coefficients of adjacent pixels of cipher image. (e) The blue color distribution of the horizontal, vertical and diagonal correlation coefficients of adjacent pixels of plain image. (f) The blue color distribution of the horizontal, vertical and diagonal correlation coefficients of adjacent pixels of cipher image

TABLE VIII. CORRELATION COEFFICIENT BETWEEN HORIZONTAL, VERTICAL AND DIAGONAL ADJACENT PIXELS

Name	Horizontal	Red	Green	Blue	Vertical	Red	Green	Blue	Diagonal	Red	Green	Blue
Lena*	P	0.9765	0.9679	0.9487	P	0.9836	0.9700	0.9634	P	0.9680	0.9471	0.9288
	C	-0.0369	0.0126	0.0368	C	0.0401	-0.0014	-0.0145	C	0.0155	-0.0421	-0.0398
Airplane*	P	0.9603	0.9687	0.9497	P	0.9696	0.9668	0.9421	P	0.9407	0.9425	0.9094
	C	0.0393	0.0388	0.0606	C	0.0489	0.0793	0.0582	C	0.0673	0.0767	0.0464
Cameraman*	P	0.9403	0.9289	0.9336	P	0.9676	0.9616	0.9668	P	0.9121	0.8988	0.9158
	C	0.0083	0.0141	0.0012	C	-0.0136	0.0178	-0.0376	C	-0.0027	0.0064	0.0199
Baboon*	P	0.9275	0.8472	0.9094	P	0.8704	0.7617	0.8784	P	0.8608	0.7346	0.8410
	C	-0.0339	0.0277	0.0087	C	-0.0023	0.0060	0.0127	C	0.0241	0.0006	-0.0083
Melanoma 1*	P	0.9911	0.9831	0.9756	P	0.9948	0.9907	0.9847	P	0.9884	0.9785	0.9679

	C	-0.0396	0.0048	0.0188	C	-0.0398	0.0107	0.0056	C	0.0029	-0.0180	0.0194
Melanoma 2*	P	0.9943	0.9930	0.9914	P	0.9951	0.9938	0.9928	P	0.9915	0.9904	0.9873
	C	-0.0110	-0.0344	0.0276	C	0.0165	0.0106	-0.0018	C	0.0362	0.0235	-0.0154
2K Image*	P	0.9789	0.9773	0.9824	P	0.9743	0.9702	0.9820	P	0.9628	0.9602	0.9739
	C	-0.0198	0.0130	0.0324	C	0.0058	0.0050	0.0290	C	0.0278	-0.0342	0.0121
4K Image*	P	0.9969	0.9977	0.9985	P	0.9978	0.9983	0.9990	P	0.9968	0.9975	0.9982
	C	-0.0214	-0.0269	-0.0382	C	-0.0191	-0.0139	0.0205	C	0.0475	0.0318	0.0091
8K Image*	P	0.9574	0.9497	0.9448	P	0.9488	0.9496	0.9343	P	0.9089	0.9115	0.8929
	C	0.0436	-0.0231	0.0422	C	-0.0405	-0.0239	0.0048	C	-0.0150	-0.0090	0.0059
Lena [33]	P	0.9249	0.9249	0.9249	P	-	-	-	P	-	-	-
	C	0.0002	0.0002	0.0002	C	-	-	-	C	-	-	-
Lena [34]	P	0.9608	0.9845	0.9850	P	-	-	-	P	-	-	-
	C	0.0489	-0.0624	-0.0666	C	-	-	-	C	-	-	-
Aerial 1 [17]	P	0.9473	0.9473	0.9473	P	0.8963	0.8963	0.8963	P	0.8472	0.8472	0.8472
	C	0.0017	0.0017	0.0017	C	0.0153	0.0153	0.0153	C	0.0046	0.0046	0.0046
Aerial 3 [17]	P	0.7633	0.7633	0.7633	P	0.6178	0.6178	0.6178	P	0.5698	0.5698	0.5698
	C	0.0107	0.0107	0.0107	C	0.0181	0.0181	0.0181	C	0.0022	0.0022	0.0022

I. Key Space Analysis

An acceptable image encryption algorithm must have a sufficiently large key area to resist attacks [39]. Our study is based on an encryption algorithm that can be used on color and grayscale images, through PV, IV, S-Box and Reverse S-Box keys. Our algorithm uses *1xn* size IV, PV and 16×16 size S-Box switch. The probability of IV and PV being present at once is shown in (23). The S-box we use consists of 256 different integers. For this reason, since the value of $k_{max} = 256$ the probability of finding the right S-box at once is shown in (24) [8].

$$P(PV)=\frac{1}{3^n}, \quad P(IV)=\frac{1}{256^n}. \tag{23}$$

$$P(S)=\frac{1}{\sum_{i=1}^{k_{max}}(k_{max}-i)!} \tag{24}$$

J. Computation Efficiency

Encryption processes and encryption speed have great importance for password security. The hardware used for encryption and decryption processes includes an Intel Core i5 2.4 GHZ processor, 8 GB RAM, 512 GB HDD and Windows 10 operating system. Under this conditions, the encryption speed for the block cipher encryption method we propose is 0.7747Mbit/s for color images and 1.0535Mbit/s for grayscale images. It is possible to state that the encryption speed of our proposed method is better compared to the encryption rate (average 0.6513Mbit/s) in [17].

K. Time Complexity

Time Complexity is a correlation that shows how many times a program or function must execute each operation in order to fulfill its function. Time complexity is important for encryption algorithms. In the proposed block cipher encryption algorithm, it is $\theta(N^2)$ or the generation of the PV, S-Box and RS-Box, and $\theta(N)$ for the generation of the IV. The time complexity in the proposed block cipher algorithm in Algorithm I for images with row (*m*) and column (*n*) is $\theta(N^2)$.

ALGORITHM I. ENCRYPTION TIME COMPLEXITY

```
for iteration=1,2,…,n do // T=2n+2
|   for iteration=1,2,…,m do // T=m(2m+2)
|   |Encrypting the pixel with IV // T=nm
|   |Blending the pixel with S-Box // T=nm
|   |Updating IV according to PV // T=nm
|   end for
end for
//T(n,m)=(2n+2)+n(2m+2)+3nm=5nm+4n+2
//T(n)=5nn+4n= θ(N^2)
```

L. Advantages and Disadvantages of Proposed Visual Cryptography Method

In the previous sections, the mathematical model of the visual cryptography method we proposed has been explained and subjected to cryptanalysis tests. In this chapter, the advantages, and disadvantages of the proposed visual cryptography method are explained. Disadvantage of the proposed method is increased encryption time due to large IV and PV size (according to plain image width). Advantages of proposed visual cryptography method can be listed as follows:

- Simple to use;
- Simple mathematical description;
- Fast and compact implementation in hardware;
- High encryption performance in both color and grayscale images;
- In the decryption process, there is no loss of data such as the size or resolution of the image;
- High security.

M. Comparison of some common visual cryptography algorithms

A comparison of the commonly used visual cryptography algorithms with the algorithm we proposed is shown in Table IX.

When we compare the proposed method with the DES algorithm, it is seen that our method produces less output data by using less SBox, input, round data than the DES algorithm. In these conditions, we can say that the proposed method is more flexible, faster and safer than the DES

algorithm. If we compare the proposed method with the AES algorithm, it is seen that the methods give similar results in terms of security. However, the output value of the AES algorithm is approximately two and a half times of our method, and this situation brings extra cost to the AES algorithm, especially in the decryption phase. Blow-Fish algorithm requires extra cost compared to proposed method due to the high round and output values. It also lags behind the proposed method in terms of security. When compared to the ECB algorithm, we can say that our method is more flexible and more secure, although it has same round, output and time complexity values with the ECB algorithm. Finally, when we compare the proposed method with the CBC algorithm, although they use similar inputs and produce similar outputs, it is understood from the performance analysis that our method is better than the CBC algorithm in terms of security.

TABLE IX. COMPARISON OF SOME COMMON VISUAL CRYPTOGRAPHY ALGORITHMS

Algorithm	Num. S-Box	Key Size	Block Size	Input	Round	Output	Structure	Flexible	Features	Speed	Time Complexity
DES [2]	8	64	64	6	16	4	Feistel	No	Not Security	Slow	$2^{39}-2^{41}$
AES [2]	1	128,192,256	128	8	10,12,14	8	Substitution Permutation	Yes	Excellent Security	Fast	2^{48}
Blow-Fish [2]	4	32-448	64	8	16	32	Feistel	Yes	Secure Enough	Fast	$\theta(N)$
ECB [3]	-	Variable	64	2	1	2	Secret Key	No	Low Security	Fast	$\theta(N^2)$
CBC [3]	-	Variable	128	3	1	2	Secret Key	Yes	High Security	Fast	$\theta(N^2)$
Proposed	**1**	**Variable**	**24**	**3**	**1**	**3**	**PHMM, Secret Key**	**Yes**	**Excellent Security**	**Fast**	$\theta(N^2)$

V. CONCLUSION

In our study, a new encryption architecture based on PHMM is developed. This architecture consists of the main components of PV, IV and S-Box. PV is generated from the original image through the PHMM model according to the RGB probability model. IV is generated by random values between the lengths of the original image. The S-Box is generated by values 0 between 255 randomly placed in a 16×16 matrix, provided that each one is used once. The fundamental difference of our architecture is pixel-based encryption that is, separately encrypting the color channels in the pixels. However, no pixels are replaced during the encryption process. This way, the decryption process is very fast. In addition, the IV used to encrypt the lines in the image is updated with the PV when the encryption of a line is completed. Thus, the line to be encrypted is encrypted independently of the previous line. Additionally, the increased encryption security in parallel to the increase in the size of the image to be encrypted is one of the significant features of our algorithm. Theoretical analysis and experimental results show that the proposed encryption algorithm provides excellent security for both color and grayscale images. Also the proposed method has shown the same performance in terms of encryption speed and time complexity in very high-resolution images (For example 2K, 4K, 8K). However, in high resolution images, the encryption time is prolonged due to the increased width of the images. In addition, it is demonstrated by the results of the performance analysis tests that the proposed method yields better results than commonly used visual cryptography algorithms and many similar studies.

REFERENCES

[1] M. Naor and A. Shamir, "Visual cryptography," in Proc. Advances in Cryptology --- EUROCRYPT'94, Berlin, Germany, 1995, pp. 1-12. doi:10.1007/BFb0053419

[2] J. Thakur and N. Kumar, "DES, AES and Blowfish: Symmetric key cryptography algorithms simulation based performance analysis," International Journal of Emerging Technology and Advanced Engineering, vol. 1, pp. 6-12, 2011

[3] Y. Kumar, R. Munjal and H. Sharma, "Comparison of symmetric and asymmetric cryptography with existing vulnerabilities and countermeasures," International Journal of Computer Science and Management Studies, vol. 11, pp. 60-63, 2011

[4] E. Mansoor, S. Khan and U. B. Khalid, "Symmetric algorithm survey: A comparative analysis," International Journal of Computer Applications, vol. 61, no. 20, pp. 12-19, 2013

[5] T. Chuman, W. Sirichotedumrong and H. Kiya, "Encryption-then-compression systems using grayscale-based image encryption for JPEG Images," IEEE Transactions on Information Forensics and Security, vol. 14, no. 6, pp. 1515-1525, 2019. doi:10.1109/TIFS.2018.2881677

[6] X. Wu and W. Sun, "Generalized random grid and its applications in visual cryptography," IEEE Transactions on Information Forensics and Security, vol. 8, no. 9, pp. 1541-1553, 2013. doi:10.1109/TIFS.2013.2274955

[7] M. Preishuber, T. Hütter, S. Katzenbeisser, A. Uhl, "Depreciating motivation and empirical security analysis of chaos-based image and video encryption," IEEE Transactions on Information Forensics and Security, vol. 13, no. 9, pp. 2137-2150, 2018. doi:10.1109/TIFS.2018.2812080

[8] E. Guvenoglu, "A dynamic S-BOX design method for image encryption," El-Cezeri: Journal of Science and Engineering, vol. 3, no. 2, pp. 179-191, 2016. doi:10.31202/ecjse.264182

[9] O. Reyad, M. A. Mofaddel, W. M. Abd-Elhafiez and M. Fathy, "A novel image encryption scheme based on different block sizes for grayscale and color images," in Proc. 2017 12th International Conference on Computer Engineering and Systems (ICCES), Cairo, Egypt, 2017, pp. 455-461. doi:10.1109/ICCES.2017.8275351

[10] A. C. Bagbaba, B. Ors, O. S. Kayhan and A. T. Erozan, "JPEG image encryption via TEA algorithm," in Proc. 2015 23nd Signal Processing and Communications Applications Conference (SIU), Malatya, Turkey, 2015, pp. 2090-2093. doi:10.1109/SIU.2015.7130282

[11] S. I. Bejinariu, R. Luca and H. Costin, "Nature-inspired algorithms based multispectral image fusion," in Proc. 9th International Conference and Exposition on Electrical and Power Engineering (EPE), Iasi, Romania, 2016, pp. 10-15. doi:10.1109/ICEPE.2016.7781293

[12] S. K. Abd-El-Hafiz, A. G. Radwan, S. H. Abdel-Haleem, M. L. Barakat, "A fractal-based image encryption system," IET Image Processing, vol. 8, no. 12, pp. 742-752, 2014. doi:10.1049/iet-ipr.2013.0570

[13] E. Guvenoglu and E. M. Esin, "Image encryption based on Knutt / Durstenfeld Shuffle Algorithm," Journal of Polytechnic, vol. 12, no. 3, pp. 151 – 155, 2009. doi:10.2339/2009.12.3

[14] C. Naveen, T. V. S. Gupta, V. R. Satpute, A. S. Gandhi, "A simple and efficient approach for medical image security using chaos on EZW," in Proc. 2015 Eighth International Conference on Advances in Pattern Recognition (ICAPR), Kolkata, India, 2015, pp. 1-6. doi:10.1109/ICAPR.2015.7050653

[15] M. Benssalah, Y. Rhaskali and M. S. Azzaz, "Medical images encryption based on elliptic curve cryptography and chaos theory," in Proc. 2018 International Conference on Smart Communications in Network Technologies (SaCoNeT), El Oued, Algeria, 2018, pp. 222-226. doi:10.1109/SaCoNeT.2018.8585512

[16] R. Chaudhary, A. Jindal, G. S. Aujla, et al., "LSCSH: Lattice-based becure cryptosystem for smart healthcare in smart cities environment," IEEE Communications Magazine, vol. 56, no. 4, pp. 24-32, 2018. doi:10.1109/MCOM.2018.1700787

[17] H. Liu, B. Zhao and L. Huang, "A remote-sensing image encryption scheme using DNA bases probability and two-dimensional logistic map," IEEE Access, vol. 7, pp. 65450-65459, 2019. doi:10.1109/ACCESS.2019.2917498

[18] A. Krogh, M. Brown, I. S. Mian, K. Sjolander and D. Haussler, "Hidden Markov Models in computational biology: Applications to protein modeling," Journal of Molecular Biology, vol. 235, no. 5, pp. 1501-1531, 1994. doi:10.1006/jmbi.1994.1104

[19] D. Mount, Bioinformatics: Sequence and Genome Analysis. Cold Spring Harbor Laboratory Press, pp. 409-494, 2004

[20] F. K. Gulagız, "Estimation of synchronization time in content delivery networks with profile hidden Markov Model," PhD. Thesis, Kocaeli University, 2018

[21] R. Durbin, S. R. Eddy, A. Krogh, G. J. Mitchison, Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids. Cambridge University Press, pp. 101-134, 1998. doi:10.1017/CBO9780511790492

[22] T. Quach and M. Farooq, "Maximum likelihood track formation with the Viterbi algorithm," in Proc. Proceedings of 1994 33rd IEEE Conference on Decision and Control, Lake Buena Vista, FL, USA, 1994, pp. 271-276. doi:10.1109/CDC.1994.410918

[23] A. Abidi, Q. Wang, B. Bouallegue, M. Machhout and C. Guyeux, "Quantitative evaluation of chaotic CBC mode of operation," in Proc. 2016 2nd International Conference on Advanced Technologies for Signal and Image Processing (ATSIP), Monastir, Tunisia, 2016, pp. 88-92. doi:10.1109/ATSIP.2016.7523053

[24] K. T. Huang, J. H. Chiu and S. S. Shen, "A novel structure with dynamic operation mode for symmetric-key block ciphers," International Journal of Network Security and Its Applications, vol. 5, no. 1, pp. 17, 2013. doi:10.5121/ijnsa.2013.5102

[25] J. K. Wolf, A. M. Viterbi and G. S. Dixon, "Finding the best set of k paths through a trellis with application to multitarget tracking," IEEE Transactions on Aerospace and Electronic Systems, vol. 25, no. 2, pp. 287-296, 1989. doi:10.1109/7.18692

[26] N. C. F. Codella, V. Rotemberg, P. Tschandl, et al., "Skin lesion analysis toward melanoma detection," in Proc. 2018: A Challenge Hosted by the International Skin Imaging Collaboration (ISIC), Granada, Spain, 2019, pp. 1-12

[27] Y. Wu, J. P. Noonan, G. Yang, H. Jin, "Image encryption using the two-dimensional logistic chaotic map," Journal of Electronic Imaging, vol. 21, no.1, pp. 1-16, 2012. doi:10.1117/1.JEI.21.1.013014

[28] J. Alireza and M. Abdolrasoul, "Image encryption using chaos and block cipher," Computer and Information Science, vol. 4, no. 1, pp. 172-185, 2010. doi:10.5539/cis.v4n1p172

[29] T. Aydogan and C. Bayilmis, "A new efficient block matching data hiding method based on scanning order selection in medical images," Turkish Journal of Electrical Engineering and Computer Sciences, vol. 25, pp. 461-473, 2017. doi:10.3906/elk-1506-189

[30] A. Hore and D. Ziou, "Image quality metrics: PSNR vs. SSIM," in Proc. 2010 20th International Conference on Pattern Recognition, Istanbul, Turkey, 2010, pp. 2366-2369. doi:10.1109/ICPR.2010.579

[31] Z. Wang, A. C. Bovik, H. R. Sheikh, E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," IEEE Transactions on Image Processing, vol. 13, no. 4, pp. 600-612, 2004. doi:10.1109/TIP.2003.819861

[32] Z. Wang, E. P. Simoncelli and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in Proc. The Thrity-Seventh Asilomar Conference on Signals, Systems and Computers, Pacific Grove, CA, USA, 2003, pp. 1398-1402. doi:10.1109/ACSSC.2003.1292216

[33] C. Zhu, G. Wang and K. Sun, "Cryptanalysis and improvement on an image encryption algorithm design using a novel chaos based S-Box," Symmetry, vol. 10, pp. 399-414, 2018. doi:10.3390/sym10090399

[34] P. Narendra, "Design and analysis of a novel digital image encryption scheme," International Journal of Network Security and Its Applications, vol. 4, pp. 95-108, 2012. doi:10.5121/ijnsa.2012.4207

[35] J. Wu, X. Liao and B. Yang, "Color image encryption based on chaotic systems and elliptic curve ElGamal scheme," Signal Processing, vol. 141, pp. 109-124, 2017. doi:10.1016/j.sigpro.2017.04.006

[36] Y. Wu, J. P. Noonan and S. S. Agaian, "NPCR and UACI randomness tests for image encryption," Cyber Journals: Journal of Selected Areas in Telecommunications, vol. 2, pp. 31-38, 2011

[37] Y. Huang, L. Cao, J. Zhang, L. Pan and Y. Liu, "Exploring feature coupling and model coupling for image source identification," IEEE Transactions on Information Forensics and Security, vol. 13, no. 12, pp. 3108-3121, 2018. doi:10.1109/TIFS.2018.2838079

[38] N. S. Atalay, S. Dogan, T. Tuncer, E. Akbal, "Image encryption methods and algorithms," Dicle University Journal of Engineering, vol. 10, no. 3, pp. 815-831, 2019. doi:10.24012/dumf.478877

[39] Y. Song, Z. Zhu, W. Zhang, H. Yu and Y. Zhao, "Efficient and secure image encryption algorithm using a novel key-substitution architecture," IEEE Access, vol. 7, pp. 84386-84400, 2019. doi:10.1109/ACCESS.2019.2923018