Multimedia Tools and Applications
https://doi.org/10.1007/s11042-021-10700-x

Image encryption method based on improved ECC and modified AES algorithm

Amal Hafsa[1] · Anissa Sghaier[1] · Jihene Malek[1,2] · Mohsen Machhout[1]

Received: 10 January 2020 / Revised: 17 January 2021 / Accepted: 10 February 2021
Published online: 02 March 2021

Abstract
Currently, embedded systems can be found everywhere in quotidian life. In the development of embedded systems, information security is one of the important factors. Encryption is an efficient technique to protect information against attacks. However, because of constraints, existing encryption functions are not compatible and do not agree with real-time applications in embedded systems. In this paper, an improved cryptographic approach with a high level of security and high speed is put forward. Our work uses an efficient version of a hybrid scheme comprising an Advanced Encryption Standard (AES) - Elliptic Curve Cryptography (ECC) for medical image encryption, which combines the benefits of the symmetric AES to speed-up data encryption and asymmetric ECC in order to secure the interchange of a symmetric session key. The contribution of this paper consists of the following two main points: First, we put forward an optimized ECC hardware architecture to respect the compromise between area, power dissipation, and speed. Thus, we primarily utilize only two multipliers to develop the Point Addition (PA) block and the Point Doubling (PD) block, which reduces time complexity. Then, a 32-bit multiplier and a 32-bit inverter architecture based on shifts and XORs are proposed to reduce power consumption and area occupancy. Second, for image encryption, we primarily propose to modify the AES by eliminating the mix-columns transformation and replacing it with a permutation based on the shifts of columns, which decreases time complexity while maintaining the Shannon diffusion and the confusion principle. Then, an adjustment of the rearrangement of the general structure is given to enhance the entropy value. The global cryptosystem is implemented using a co-design approach where the modified AES runs on the NIOS II processor, and the scalar ECC multiplication is designed as a hardware accelerator. The suggested cryptographic system spends much less execution time, which is a significant factor for being applied in practice. Security analysis is successfully performed, and our experiments prove that our proposed technique provides the basics of cryptography with more simplicity and correctness. In fact, the results of the evaluation prove the effectiveness, rapidity and high security of the suggested algorithm.

✉ Amal Hafsa
amalhafsa12@gmail.com

Extended author information available on the last page of the article

Keywords Embedded system · MAES · ECC · Co-design · FPGA · NIOS II · Fast image encryption · High security

1 Introduction

Nowadays, the fast growth of the internet makes electronic healthcare (e-healthcare) feasible and popular. E-healthcare refers to an internet-based system where the patient can contact an expert doctor for diagnostic. Some medical images are stored and transmitted over the internet. These images may contain much privacy of patients and are very confidential and sensitive [15]. The best significant way to protect this privacy issue is data encryption. Medical images have some characteristics such as redundancy, a big data volume, and great pixel correlation compared to normal images. Therefore, strong encryption algorithms are required. Among encryption schemes, a hybrid scheme is the best solution to protect medical images, which mixes the advantages of symmetric and asymmetric algorithms. The symmetric scheme is effective for large volume data encryption because it is generally faster than the asymmetric scheme, but it suffers from secret key distribution. On the other hand, the asymmetric scheme is more secure than the symmetric one since it uses a pair of keys for encryption and decryption: a public key known by everyone used for encryption, and a private key known only by the recipient utilized for decryption. Although existing hybrid encryption systems have a great security level and reliability, the large computational time presents a serious obstacle, making it inappropriate to secure the multimedia transfer in embedded systems. In our proposed design, the original image is encrypted by a Modified Advanced Encryption Standard (MAES), which provides great encryption speed. Meanwhile, Elliptic Curve Cryptography (ECC) is applied to encrypt the MAES key to enhance security. However, the optimization of ECC in existing research has not been sufficient yet, which results in reduced operating frequency. In this paper, we suggest a high-speed hybrid algorithm on FPGA, in which the clear image is encrypted and decrypted by the proposed MAES, where the MAES key is considered as an ECC plaintext.

This paper makes the following contributions:

(i) Designing a hybrid framework based on the MAES for image encryption and the ECC for key security.
(ii) Designing an optimized ECC hardware architecture to respect the compromise between area, power dissipation, and speed. Thus, we primarily utilize only two multipliers to develop the PA block and the PD block, which reduces time complexity. Then, a 32-bit multiplier and a 32-bit inverter architecture based on shifts and XORs are put forward to reduce power consumption and area occupancy.
(iii) Designing a novel operation based on random permutation that replaces the mix-columns transformation in the Advanced Encryption Standard (AES). This operation is based on shifts of columns; it is simple in the calculation and it decreases time complexity. The novel MAES keeps the Shannon principle of diffusion and confusion.
(iv) Changing the rearrangement of the general structure in the AES. This change enhances the entropy value.
(v) Designing a System on Programmable Chip (SoPC) of the proposed hybrid scheme, featuring a NIOS II softcore processor, where the overall MAES and the control of

encryption/decryption in ECC run on the processor, while the scalar multiplication operation of ECC is implemented in hardware.

(vi) Undertaking in-depth experimental measurements in FPGA for several images with different contents and sizes to evaluate the strength of the suggested cryptosystem against the new generation of attacks.

(vii) Undertaking an in-depth evaluation study of the performance of the execution and comparing the results with other recent work.

The objective of this paper is to build an effective hybrid algorithm with great-level control, a simplified MAES and accelerated ECC to gain an overall powerful cryptosystem available for secure medical image transmission in embedded systems. In the following, a survey of existing work is given in Section 2. Proposed cryptographic algorithms are clearly explained in Section 3. In Section 4, the adopted approach to create the hybrid algorithm is detailed, followed by implementation results. Section 5 presents the complete security analysis over the propounded technique. Finally, the last section concludes and recommends some future work.

2 Related work

Several hybrid encryption schemes have been suggested in the literature. In [8], an AES-ECC hybrid encryption system was proposed. In this technique, the generation of the ECC pair keys was followed by the generation and encryption of the AES secret key. Then, when this token was transferred to the other part, another pair of the ECC key (Key 2) would be generated and encrypted using the symmetric algorithm. This token was transferred to the other part, and the data encryption process was performed using Key 2. Therefore, cipher ECC was achieved. Finally, the encryption of cipher-ECC was performed using the AES, and the ciphertext output was obtained to be transferred via the network. The main inconvenience of this paper consisted in employing the ECC asymmetric algorithm to encrypt all original data. This way was highly time-consuming and the system was not efficient in terms of energy consumption, particularly for weak sensors. In [24], Zhao. Z et al. proposed to encrypt images using ECC and code computing. In this method, binary bits were mapped to letters. Both subtraction and addition field arithmetics were defined over letters, and the image was encrypted using ECC, letters and operations. This way was vulnerable to the Chosen Plain text Attack (CPA), whereas the model was apparent to the adversary. Furthermore, security analysis was not carried out in [24] and the system was not effective in terms of speed. Hajajneh. T et al. put forward in [11] a cryptographic system that would secure multimedia applications in FPGA. The goal of this work was to perform authenticity and encryption using a cipher block chaining message authentication code protocol and a counter protocol. Though the results indicated an improvement in speed, the overall system risked to be attacked [11, 12] and these techniques did not provide any possibility of enhancement. Attaya A.M et al. [3] suggested to employ a hybrid system that would combine both chaos and AES algorithms. In the AES, both the substitution box and the add-round key were replaced by a chaos generator, which led to an increase in both diffusion and confusion and decreased the run time, compared with the standard AES. Yet, encryption could outcome a feeble code when compared with the traditional AES since there was only one step that performs the entropy, compared with the different steps in the AES. Nevertheless, the authors declared that the decryption process was impossible without

the key [3]. In [20], Shankar. K et al. proposed to encrypt images using an asymmetric encryption key. They utilized the genetic algorithm to get the ideal key. In this way, ECC was utilized to encrypt all pixels one by one. However, employing an asymmetric algorithm for every pixel and researching for the ideal keys were costly operations. In [7], Gafsi M et al. put forward a hybrid scheme for numerical image secure transmission. In this paper, the authors selected the Rivest–Shamir–Adleman (RSA) algorithm to encrypt the secret key. Nevertheless, when it came to performance at 128-bit security levels, the RSA would be reported to be 10 times slower than ECC for private key operations. In [16], Liu. H et al. proposed a cooperation between ECC and a chaotic system. In that paper, the authors tried to dissolve the issue of vulnerability against the Known Plaintext Attack (KPA), so they overcast the errors by using the Chirikov standard map for the diffusion and confusion of the image. Similarly, they employed a preceding stream of the encrypted images to encrypt the next stream. However, some problems could be produced with the generated logic map because there existed a correlation between the Xn values of the chaotic system [21]. In [5], the authors proposed to combine either a 2D Discrete Wavelet Transform 1 Level (2D-DWT-1 L) or 2D Discrete Wavelet Transform 2 Level (2D-DWT-2 L) steganography model with a hybrid cryptosystem. This latter was built utilizing an AES and RSA combination. In [21], Toughi. Sh et al. suggested a hybrid cryptosystem for color image protection. Their model was a combination of the ECC and the AES. Accordingly, the elliptic curves were employed for generating three random sequences. These sequences were utilized for generating three masks to encrypt the red, blue, and green components of the clear image. The drawback of this work was the use of a sequential method to encrypt the image, which caused the degradation of the performance of the system in terms of speed. Our motivation in this paper is to propound a technique that is secure while addressing the issues of preceding work. In particular, a fast and secure image encryption system that can be implemented in embedded systems is suggested. The proposed method focuses on the key sharing way, where the initial MAES-key encryption is performed by an accelerated ECC for more security in transmission. Then, a modified MAES with simple operations is designed to assure data encryption in short time. Thus, the use of enhanced algorithms leads to the amelioration of the overall hybrid design, which explains the good results found after implementation in terms of run-time, area occupation and power consumption. The use of the co-design approach between the two improved algorithms enhances the system. However, the utilized NIOS II CPU is a relatively powerful one amongst embedded processors.

3 Proposed cryptographic algorithms

The material constraints of embedded systems necessitates great amelioration in the cryptographic algorithm to reduce area occupation, power consumption and especially time complexity while increasing the security performances. In this section, we will give a novel presentation of the AES to speed up the run time on the NIOS II processor and enhance security performance efficiency. Then, an area-time efficient hardware implementation of ECC over GF(2^m) is presented. The improved design aims to gain an improved cooperative framework able to secure medical image transmission in embedded systems.

3.1 MAES

The AES is one of the most known encryption algorithms for data protection. It was invented in 1998 by Joan Daemen and Vincent Rijmen and proved in 2000 by the NIST. The AES has been widely deployed thanks to its high performance. It involves key sizes and block sizes. The size of the information block is 128 bits, and the length of the key can be 128, 192, or 256 bits [6]. The repetition and the size of the key determine the complexity of the algorithm. A higher repetition or elevated key size provokes higher CPU usage and complexity. In this work, reduced processing time is needed. Then, a 128-bit key size is sufficient.

For the encryption operation, round transformation is performed as a set of iterations, which includes sub-bytes, shift-rows, mix-columns, and add-round key operations. For the sub-bytes operation, an S-box table is utilized to substitute every block byte with a novel block which embodies Shannon's principle of confusion. For shift-rows, every row of the matrix is performed by a cyclic shift to the left according to its position. It guarantees the Shannon diffusion principle. The mix-columns transformation consists of multiplying a constant matrix with the state matrix. This operation also participates in the Shannon diffusion principle. Finally, an XOR between the round key and the matrix state is performed to obtain an intermediate matrix. In the AES, the mix-columns transformation requires large computation time to be processed since it is based on the modular multiplication of columns. The main idea is to decrease as much as possible the processing time by minimizing the arithmetic operations. The proposed way is to eliminate the mix-columns operation and replace it with a novel procedure simple in calculation, which helps speed gain while maintaining the Shannon principle of the diffusion and the confusion. The suggested method is developed on the AES with a novel transformation called 'shift-columns'. The shift-columns operation takes place on different cyclic columns with various offsets, which enables good state permutation. A second modification is given, which consists in changing the rearrangement of the general structure to enhance the entropy value. The MAES includes four blocks as follows: shift-columns, sub-bytes, shift-rows, and add-round keys, as illustrated in Fig. 1. By these modifications, the MAES is adjusted to resolve the issue of hard calculation. Thus, the best encryption speed is obtained.

3.2 Improved ECC

ECC has done a revolution in public keys since 1987 thanks to its small operand length compared to other asymmetric algorithms [13]. ECC has various benefits such as rapid calculation, reduced power, and memory consumption. ECC is utilized for key exchanging, authentication, digital signature, etc.

The elliptical curve equation is as follows:

$$y^2 = x^3 + ax + b \tag{1}$$

where both a and b parameters are fixed, and x and y belong to the finite field (binary or prime field). The most processing time-consuming operation of ECC is point multiplication where a point P is multiplied by an integer k giving a novel point Q that belongs to the curve. Scalar multiplication is the base of ECC. There exist various algorithms to calculate it. Montgomery scalar multiplication is the most popular algorithm resistant against side channel attacks. To calculate the point multiplication $Q = k.P$, two basic operations should be computed: PA and

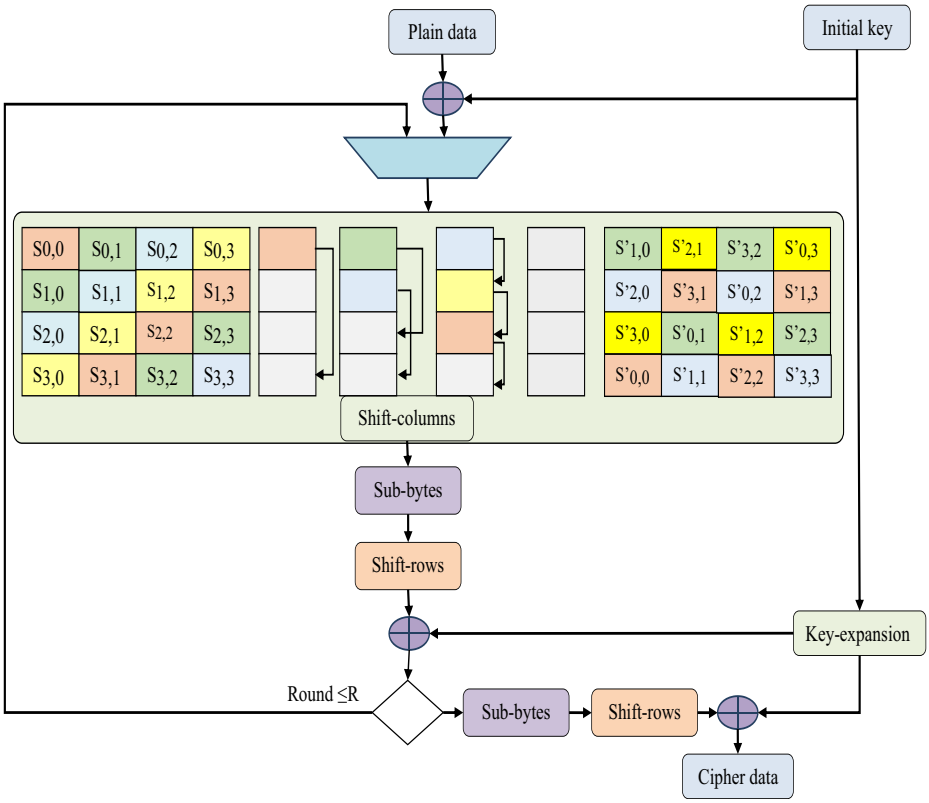


Fig. 1 Flow design of proposed MAES algorithm

PD. PA calculates a third point on the curve, taking two different input points, while the PD computes a third point on the curve when the two inputs are the same point. Both PA and PD are based on arithmetic operations (addition, multiplication, squaring, subtraction and inversion). To avert modular inversion in both PD and PA, Lopez and Dahab projective operations are utilized [10]. The security of the ECC is linked to the capability of computing the scalar (point) multiplication. Therefore, to be performed, the Montgomery algorithm is employed to exploit the parallelism of PA and PD which are computed independently. In this paper, these two operations are calculated at the same time, as shown in Fig. 2. In fact, PA necessitates five multiplications and PD requires six multiplications, but only two multiplications are used in every transformation in this paper.

This method is founded on the full-time-function components. Therefore, both the point conversion and the point operation use only two multipliers. They are activated and reactivated in the following step. The reuse component and the full-time-function components are the main optimization in the ECC architecture to reduce time complexity. The target of our conception is to accelerate the scalar multiplication process by parallelizing both PA and PD Montgomery algorithms. PA and PD use only two multipliers in every step. The reuse of these two blocks minimizes the area occupancy and the power conception of the overall system [19].

Finally, to improve the performance of the ECC cryptosystem, field multiplication and inversion are simplified and the corresponding block number is decreased.

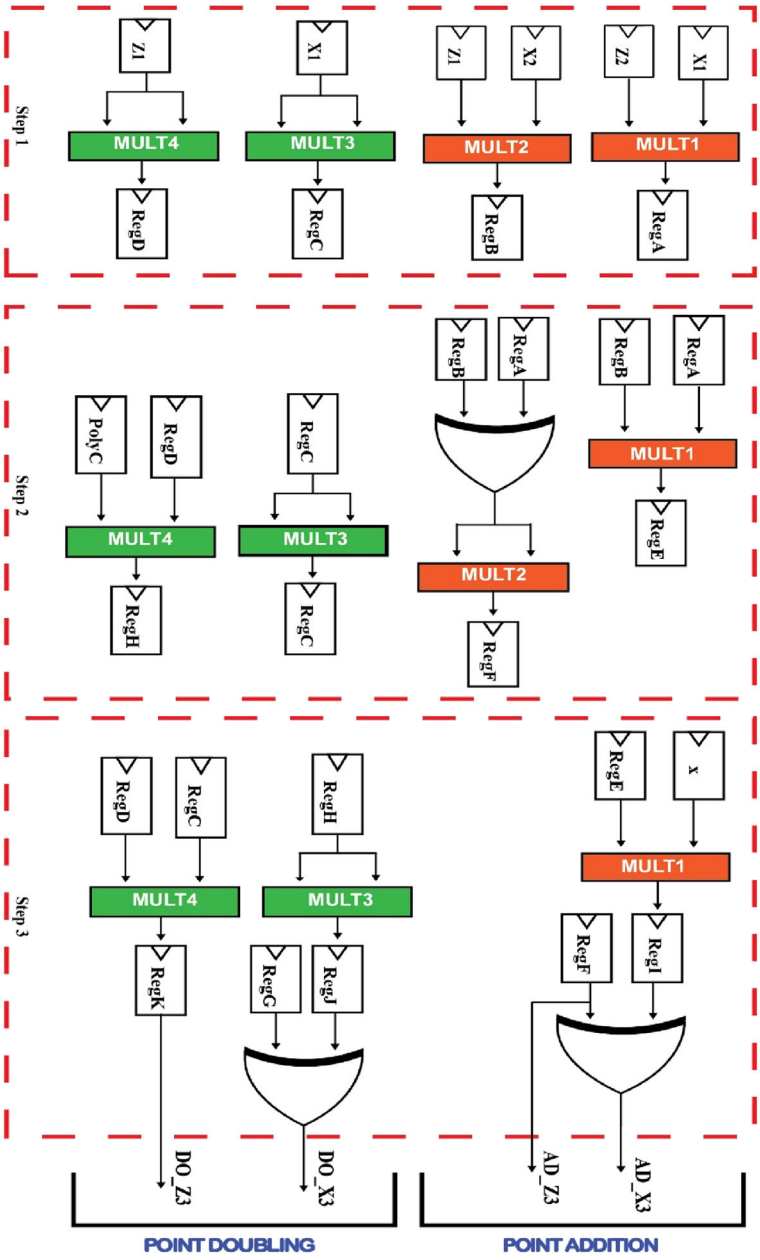


Fig. 2 Parallelism between PA and PD

3.2.1 Proposed field multiplication

In the ECC hardware design, the main objective is to efficiently implement field multiplication. Algorithm 1 provides a 32-bit multiplier. A(x), B(x) and I(x) are three input polynomials. They are given to calculate A(x) × B(x) mod I(x). At the beginning, we have to split the whole

polynomials into N 32-bit parts. The registers are equal to 32 bits. The multiplication is based on only offsets ("shifts") and XORs. This decreases the cost and power consumption of the multiplier. The architecture of the suggested multiplier is presented in Fig.3. It includes a controller that represents the fundamental element performing the entire computation. It is based on a finite state machine that handles the algorithm loops and synchronizes the findings and their updates in the RAMR update, which is a random-access memory. In practice, our design is very easy to implement in embedded systems for gaining consumption, surface, and speed.

***Algorithm 1**: Multiplication series in F_2m*

Input: *A(x), B(x)* $\in F_2^m$, *I(x) irreducible polynomial of degree m*
Output: *R(x) = A(x). B(x) mod I(x)*
1: *Set R(x) = 0*
2: ***For*** *i from N to 0* ***do***
3: ***For*** *i from 31 to 0* ***do***
4: *Offset of R(x) (SHL)*
5: *R(x) = R(x) +A(x) bi*
6: *R(x) = R(x) +r_m I(x)*
7: ***End for***
8: ***End for***
9: ***Return R (x)***

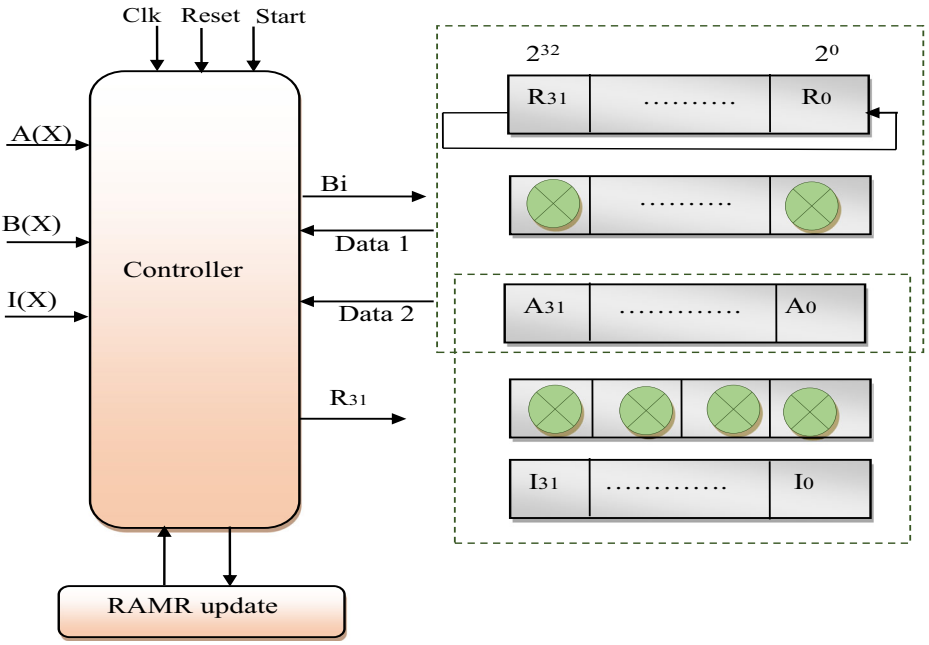


Fig. 3 Proposed field multiplier architecture

3.2.2 Proposed field inversion

The inversion operation consists in finding $S^{-1}(x)$ with $S^{-1}(x) \times S(x) = 1 \bmod I(x)$. Algorithm 2 presents a novel technique to calculate the N-bit modular inversion over GF (2^m). The calculation is performed with a 32-bit datapath. It is based on shifts and XORs, which decreases area occupation.

Algorithm 2: *Inversion series in F_2m*

Input: *$A(x) \in F_2m$, $I(x)$ irreducible polynomial of degree m*

Output: *$A^{-1}(x) = A(x) \bmod I(x)$*

1: *$Y(x) = A(x)$, $X(x) = 1$, $B(x) = 0$, $D(x) = I(x)$*
2: *For i from N to 0 do*
3: *For i from N-1 to 0 do*
4: *If Y (0) = 0 then*
5: *Shift Y(x)*
6: *If X (0) ≠ 0 then*
7: *X(x) = X(x) XORI(X), shift X(x)*
8: *Else*
9: *Shift X(x)*
10: *End if*
11: *End if*
12: *If Y (0) = 1 then*
13: *D(x) = Y(x), B(x) =X(x),*
14: *Y(x) = Y(x)XOR D(x), X(x) = X(x)XOR B(x)*
15: *Else*
16: *R(x) = X(x)*
17: ***End if***
18: ***End for***
19: ***End for***
20: ***Return R(x)***

The architectural design of the proposed field inversion is depicted in Fig.4. A controller is needed to handle shifts, computation, and storage in RAMs. Because inversion is the most expensive operation in terms of area and time complexity, it is essential to select the right technique to calculate it. The suggested way is practical to be implemented in embedded systems since it is based on shifts and XORs.

4 Cryptographic system design

4.1 Hybrid model

The hybrid encryption model provides the benefits of both asymmetric-key and symmetric-key algorithms. In this paper, the symmetric-key cryptosystem is employed to encrypt data transmitted in an insecure channel, whereas the asymmetric-key algorithm is utilized to share the key with another part so that they can cipher these data. As seen in Fig. 5, the MAES is used to encrypt/decrypt data, whereas the proposed ECC is utilized to encrypt/decrypt the MAES key. The primary MAES key is generated by a pseudo random number generator (W7) developed in C language and run on the NIOS II CPU [9].

4.1.1 Image encryption

The color image is encrypted using the modified MAES symmetric scheme. The encryption steps are as the following:

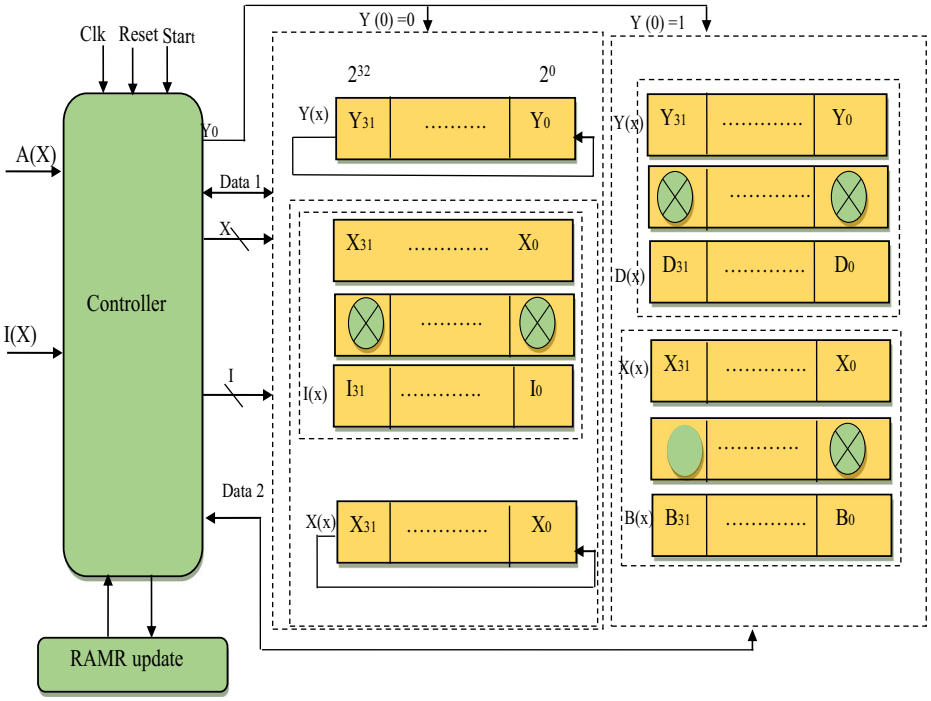


Fig. 4 Proposed fiel inversion architecture

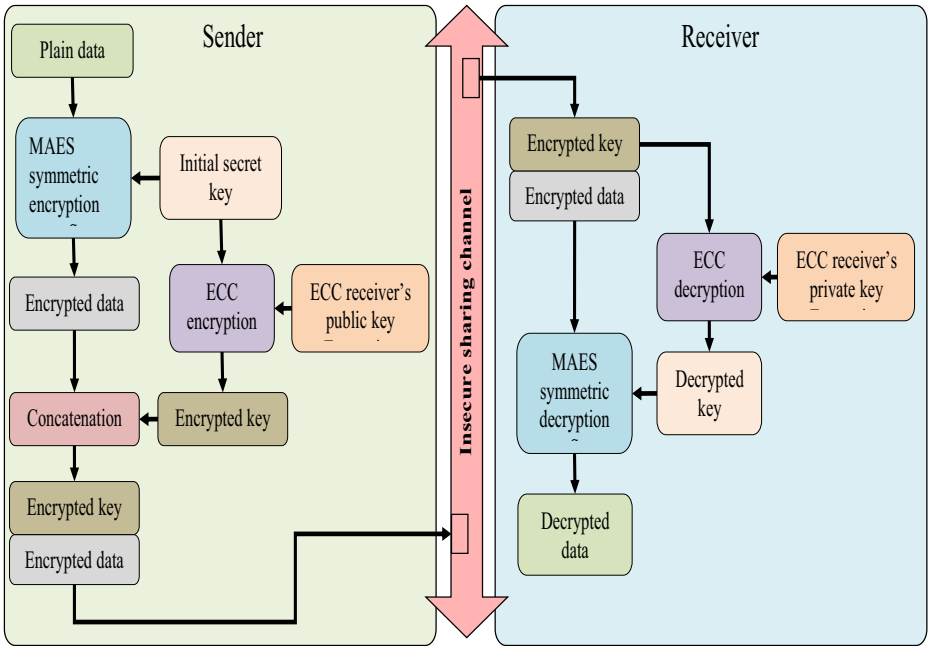


Fig. 5 General hybrid cryptography synoptic scheme

Step 1: Read a color image of size M × N × 3. M and N are the image dimensions.
Step 2: Split the color image into three components: red, green and blue.
Step 3: Generate three initial keys fully related to each channel by a W7 random number generator.
Step 4: Encrypt each channel undependably using the MAES and the corresponding initial secret key.
Step 5: Combine the three ciphered channels to obtain the encrypted color image.

Figure 6 illustrates the flow design of the proposed image encryption process.

4.1.2 Secret key sharing

The generated keys are concatenated and considered as one secret key to be encrypted by ECC. The chosen ECC-based cryptosystem is the "elliptic curve analog of the basic ElGamal

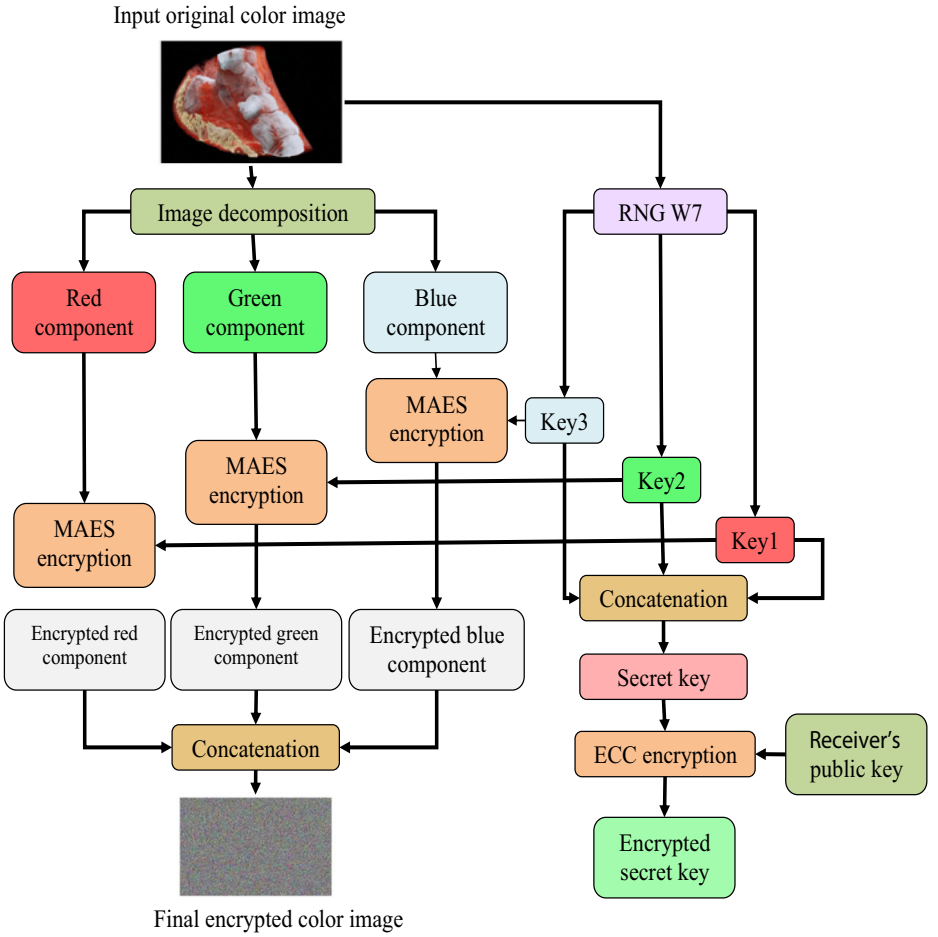


Fig. 6 Flow design of image encryption procedure

cryptographic scheme". More details are given in Algorithm 3. A clear text m, presented as a point M, is ciphered by adding it to $k.Q$, where k is a random integer and Q is the recipient's public key. The sender sends points $C1 = kP$ and $C2 = M + kQ$ to the recipient who utilizes their private key d to perform dC1 = $d(kP) = k(dP) = kQ$, and then recovers $M = C2 - kQ$, as presented in Algorithm 4. In the proposed method, the original text m represents the MAES key to be ciphered.

Algorithm 3: *Basic ElGamal elliptic curve encryption*

Input*: Elliptic curve domain parameters (p, E, P, n) public Key Q, plain text*

Output: *Ciphertext (C1, C2)*

1: *Represent the message m as point M in* $E(F_p)$
2: *Select k € R [1,n-1].*
3: *Calculate C1= k.P*
4: *Calculate C2= M+ k.Q*
5: Return (C1, C2)

Algorithm 4: *Basic ElGamal elliptic curve decryption*

Input*: Elliptic curve domain parameters (p, E, P, n) private Key d, Ciphertext (C1, C2)*

Output: *plain text m*

1: *Calculate M= C2-d C1, then extract m from M*
2: Return *Plain text m*

4.1.3 Image decryption

The decryption system to seek the plain image is the reverse approach according to the encryption steps. The private key of the ECC destination is used in a first step to decrypt the MAES encrypted key. Then, image decryption can be performed.

4.2 NIOS II system design

In this paper, the FPGA-based DE2–115 development board including Cyclone IV.E (Altera) is required. The suggested system SoPC includes the NIOS II processor, which is a 32-bit embedded processor specialized for the Altera family of FPGAs, input/outputs GPIOs, internal memory controllers, a timer to perform the run time, and a JTAG UART cable for the debug. With the help of Qsys (System Integration tool of Quartus II), the CPU is connected with all modules via an Avalon bus, as shown in Fig. 7.

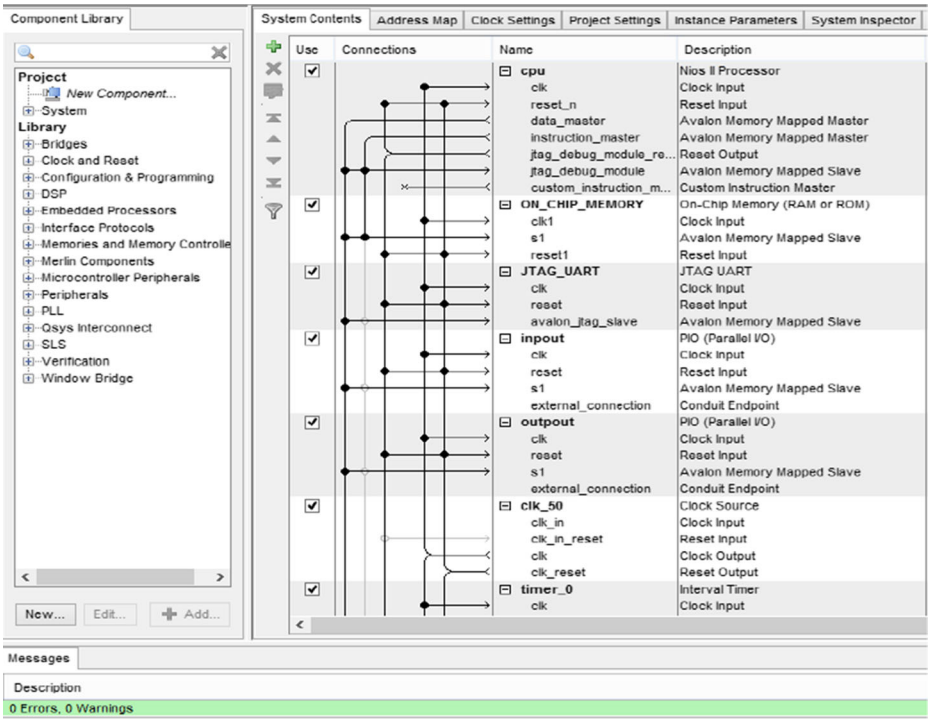


Fig. 7 Connection in Qsys GUI

4.3 MAES implementation

The MAES cryptosystem is developed in embedded C language and is executed on the NIOS II CPU. The implementation results of the MAES in the SoPC at 50 MHz, in terms of run time for digital images and 3 D medical scanner ankle image, are provided in Table 1.

The results indicate that the processing time of the suggested MAES running on the NIOS II processor is much less than the standard AES. Our proposed method can speed up the system compared to the existing AES. The modification applied to the AES aims to decrease the maximal execution time while maintaining the Shannon principle of confusion and diffusion.

4.4 ECC implementation and interfacing

The ECC hardware IP is interfaced to the system via GPIOs. In order to adapt the input/output GPIOs (32-bit) of the system designed to the hardware block, two interfaces are employed, as depicted in Fig. 8, to wrap ECC with control signals from the CPU. The ElGamal encryption

Table 1 Run-time results of MAES on DE2–115 and comparison with the standard AES

Execution time of encryption (s)	Lena (512×512×3)	Pepper (512×512×3)	Wish-ir (2250×2250×3)	3D scanner ankle (1080×1920×3)
AES algorithm	68.2218	67.2761	269.1044	190.1458
MAES algorithm	0.23254	0.22897	0.63015	0.57935

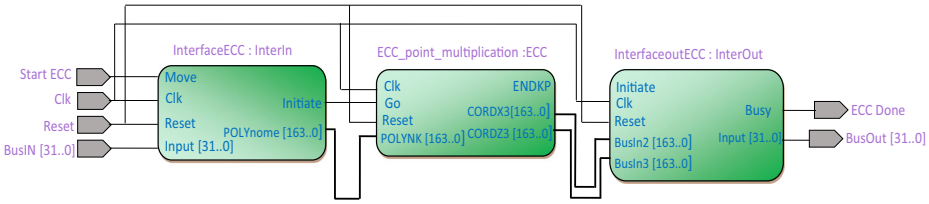


Fig. 8 Flow design of ECC and input/output interfaces

scheme is developed in C language and is executed on the NIOS II processor. The MAES key (m) and its ciphers (C1 and C2) are transferred between the CPU and the ECC hardware block in a 32-bit datapath.

The implementation findings on Cyclone IV FPGA shows that the k.P design requires 11% of logic elements, 9% of combinational functions and 7% of logic registers, and it achieves the maximal time of 143.512 ns.

4.5 Complete hybrid system implementation on DE2–115 board

The diagram depicted in Fig. 9 presents the communication between ECC integrated as IP accelerator and the MAES implemented in the CPU. The proposed hybrid algorithm is implemented on the DE2–115 board featuring Cyclone IV.E FPGA. Table 2 illustrates the utilization of resources extracted from Quartus II. The system needs 11% of logic elements, 9% of combinational functions, 7% of logic registers, and 7% of memory. Finally, it runs at a 157.53-MHz clock frequency, consumes 166.67 mW of power and can achieve a great throughput of 1.72 Gbit/s. Concluding the obtained results, the proposed cryptosystem hardware design occupies a small hardware area, has reduced power and reaches a high throughput.

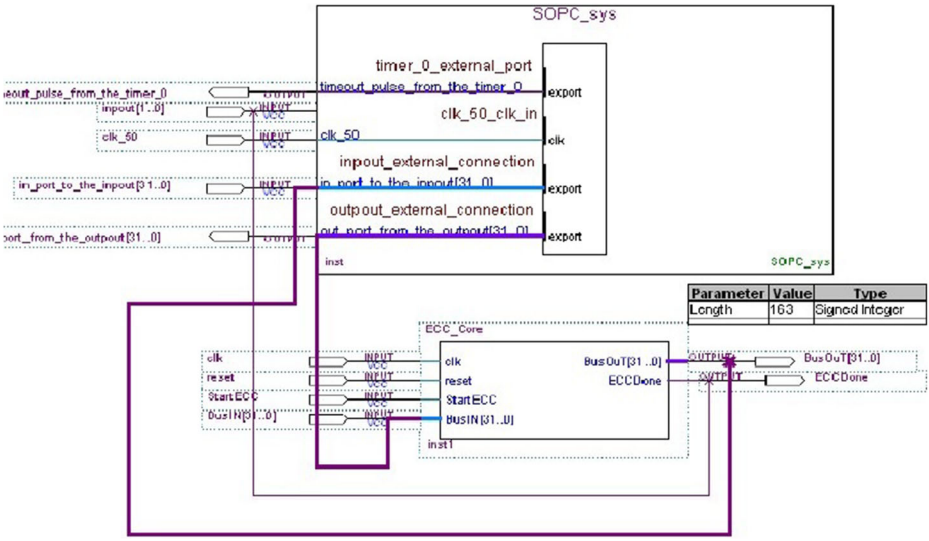


Fig. 9 Block diagram of overall system

Table 2 Hardware implementation results of proposed algorithm

Hardware performances	Cyclone IV. E FPGA
Total logic elements	11.638/114.480 (11%)
Total combinational function	9.441/114.480 (9%)
Dedicated logic registers	8.326/114.480 (7%)
F MAX	157.63 MHz.
Total thermal power dissipation	166.67 Mw
Memory	273.408/3.981312 (7%)
Throughput	1.72 Gbit/s.

5 Security analysis

As usual, in order to satisfy a novel security approach, an evaluation study is performed. Ordinary and medical images with different sizes and types are used for the test. For ordinary color images, we use the standard Lena, Peppers, Baboon images of size ($512 \times 512 \times 3$) and the Wish-ir image of size ($2250 \times 2250 \times 3$). For medical images, a 3D scanner ankle of size ($1080 \times 1920 \times 3$) and a 3D X-ray chest image ($2048 \times 1520 \times 3$) are chosen. We discuss the security analysis performances of the proposed method at several levels: visibility test, histogram, entropy, correlation of adjacent pixels, Unified Average Changing Intensity (UACI), Normal Correlation (NC) of adjacent pixels, Number of Pixels Change Rate (NPCR), keyspace, and robustness against noise attack, KPA, and CPA. In addition, a comparison is performed with the existing work, involving the run time and the security performance.

5.1 Visibility test

The implementation of the novel hybrid encryption model is applied to the standard color Lena image, the Wish-ir image, the 3D scanner ankle image and the 3D Chest X-ray image, as depicted in Fig. 10. The results of encrypted images show the perfect invisibility of information. This proves the strength of the proposed method. The results of encrypted images are also illustrated in this section.

5.2 Statistical analysis

The histogram analysis and the correlation of adjacent pixels are the fundamental parameters required to prove that the proposed cryptographic model is resistant to statistical attacks.

5.2.1 Histogram analysis

The histogram of one image shows the value of every pixel. An improved cryptographic system must create a uniform color distributed pixel [22]. Pictures (a), (b) and (c) in Fig. 11 present a histogram of red, green and blue components for Lena's color plain image. Pictures (d), (e) and (f) are the corresponding histograms after encryption. Similarly, Fig. 12 presents the corresponding histograms of the Wish-ir image before and after encryption in each channel, Fig. 13 presents histograms of the original 3D medical scanner ankle and its corresponding histogram after encryption, and Fig. 14 presents the corresponding histograms of the 3D X-ray chest image before and after encryption in each channel. As shown in these

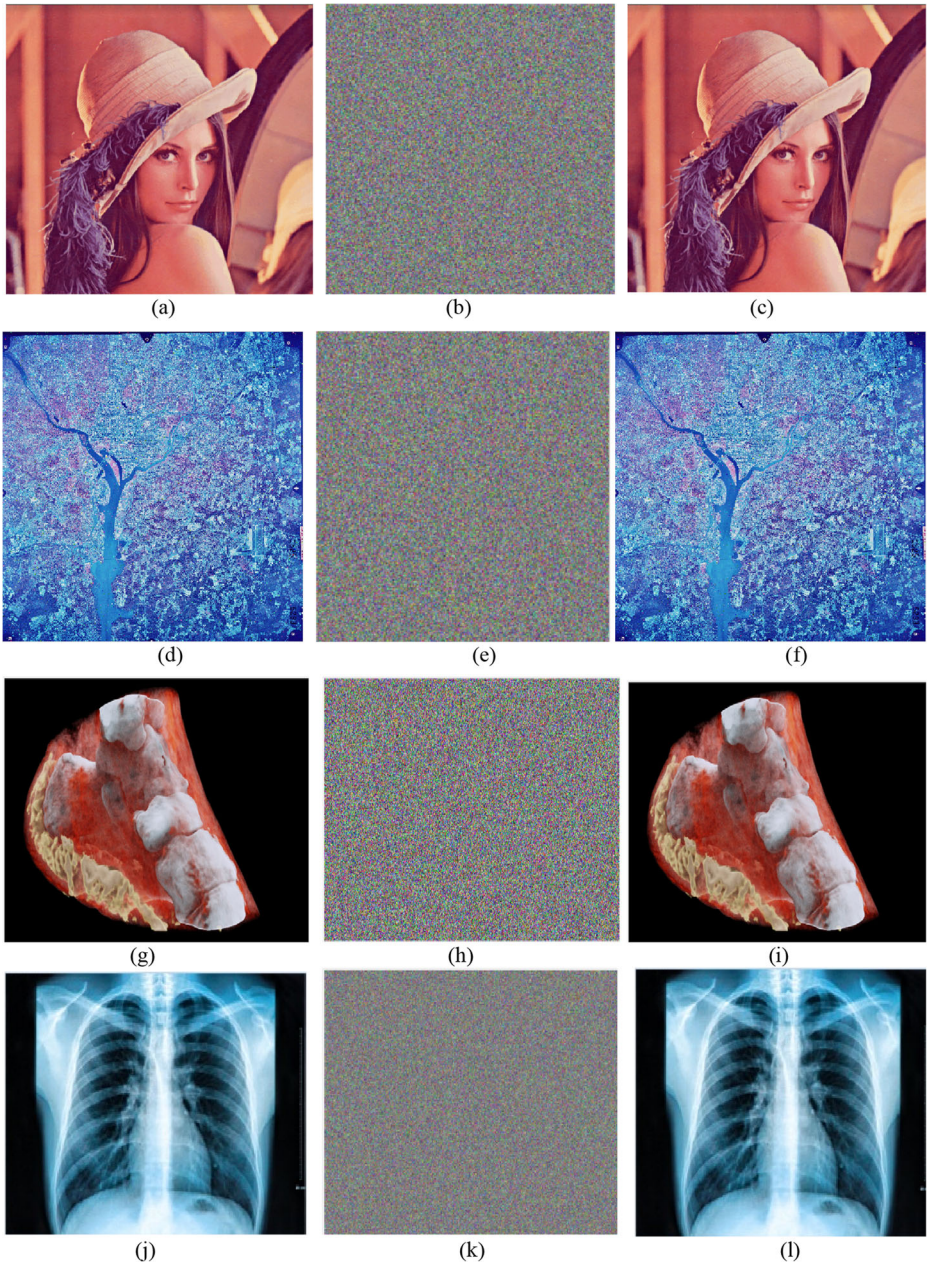


Fig. 10 Visibility test: **a d g j**: Original images; **b e h k**: Encrypted images; **c f i l**: Decrypted images

figures, the histograms of the ciphered images are uniform. Therefore, any information can be detected from the ciphered image, which proves that the proposed model is resistant to statistical attacks.

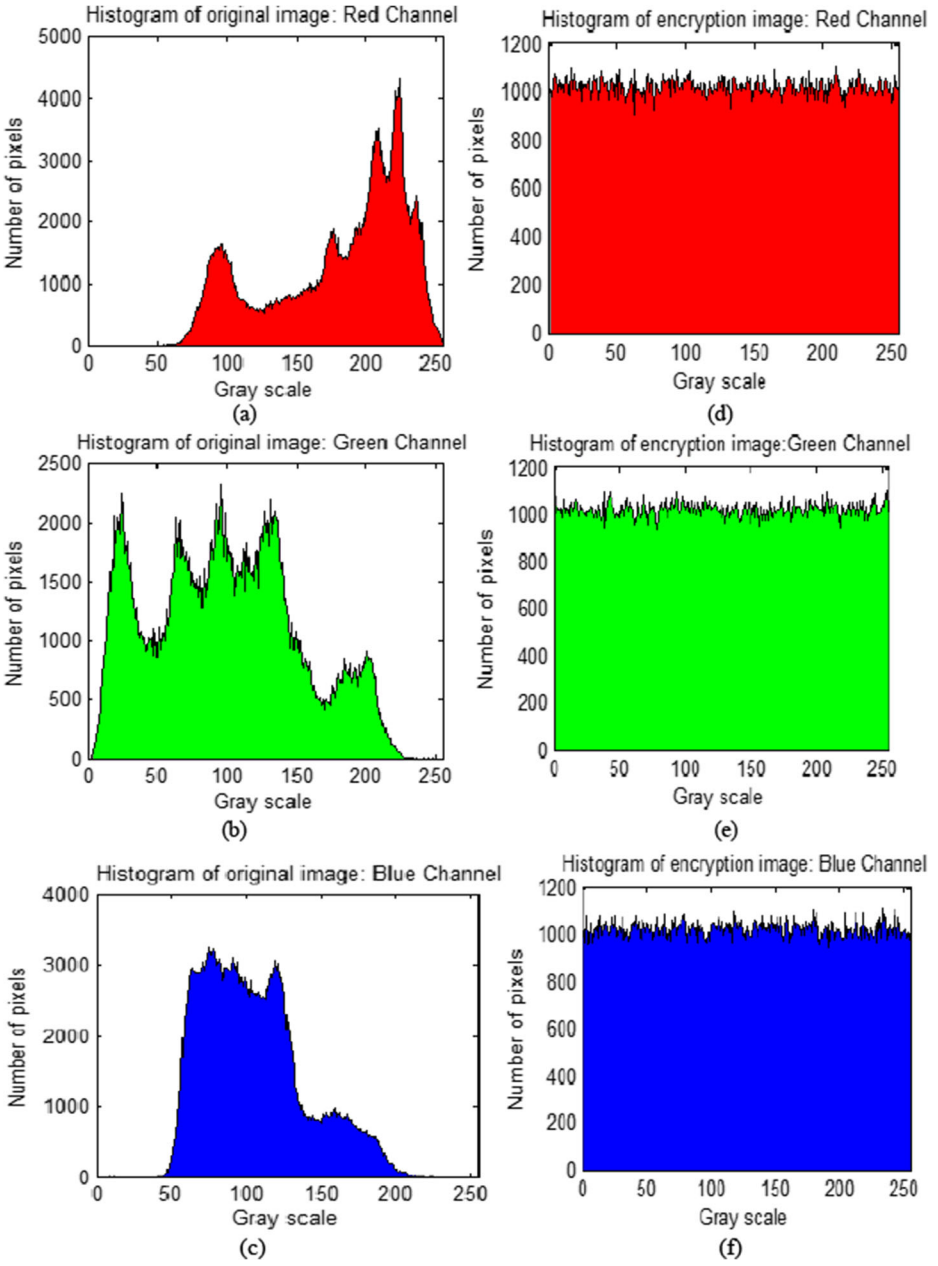


Fig. 11 Histograms illustration of: **a**: Red original image of 'Lena', **b**: Green original image of 'Lena', **c**: Blue original image of 'Lena', **d**: Red ciphered image of 'Lena', **e**: Green ciphered image of 'Lena', **f**: Blue ciphered image of 'Lena'

5.2.2 Correlation coefficient analysis

In the original image, every pixel is correlated with its adjacent pixels in horizontal, vertical and diagonal directions. In a great encrypted image, every pixel has the least correlation with

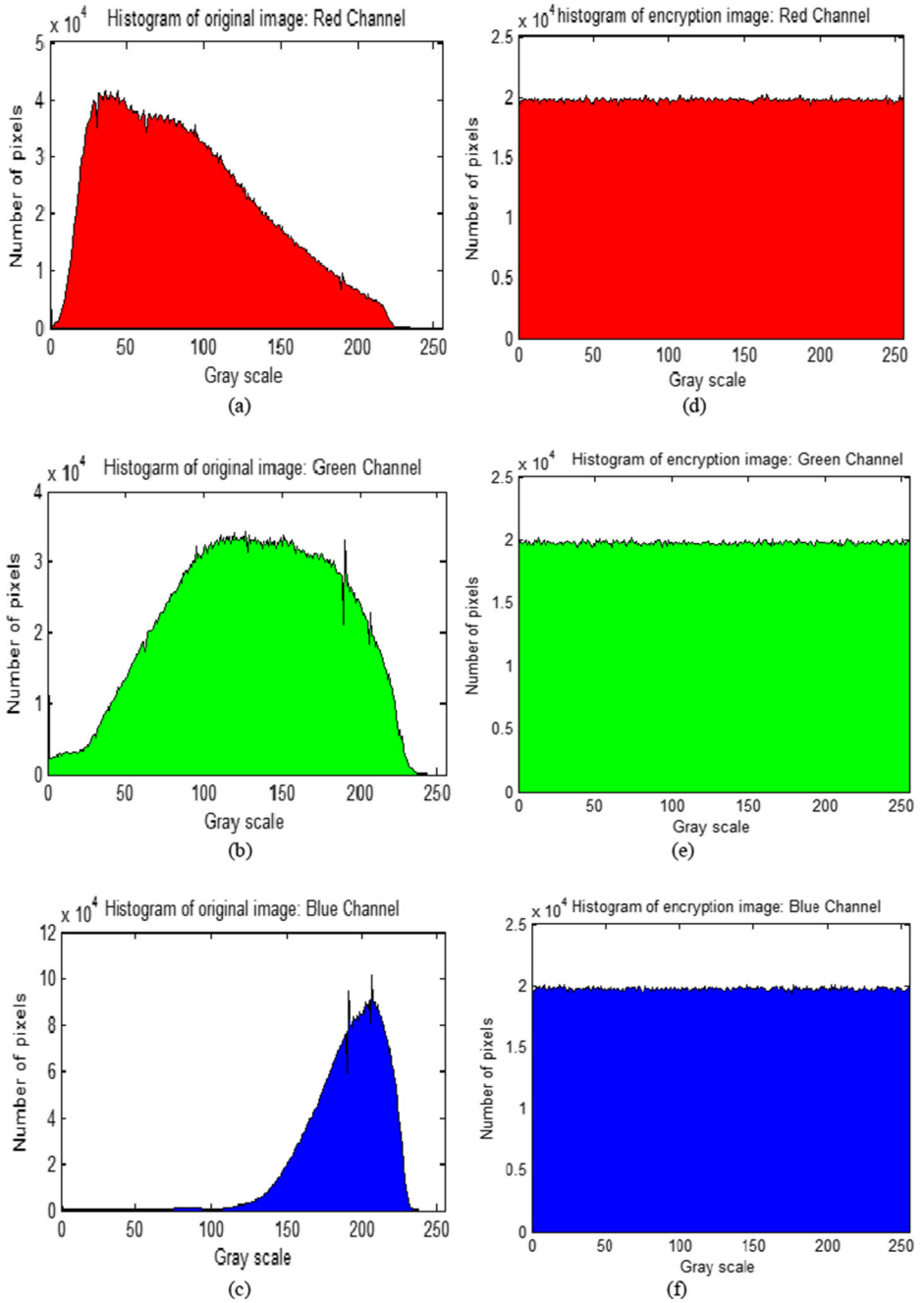


Fig. 12 Histograms illustration of: **a**: Red original image of 'Wash-ir', **b**: Green original image of 'Wash-ir', **c**: Blue original image of 'Wash-ir', **d**: Red ciphered image of 'Wash-ir', **e**: Green ciphered image of 'Wash-ir', **f**: Blue ciphered image of 'Wash-ir'

its adjacent pixels [21]. To visually illustrate the correlation of adjacent pixels, samples of both Lena's plain and encrypted images are taken. Then, the distribution of the adjacent pixels is

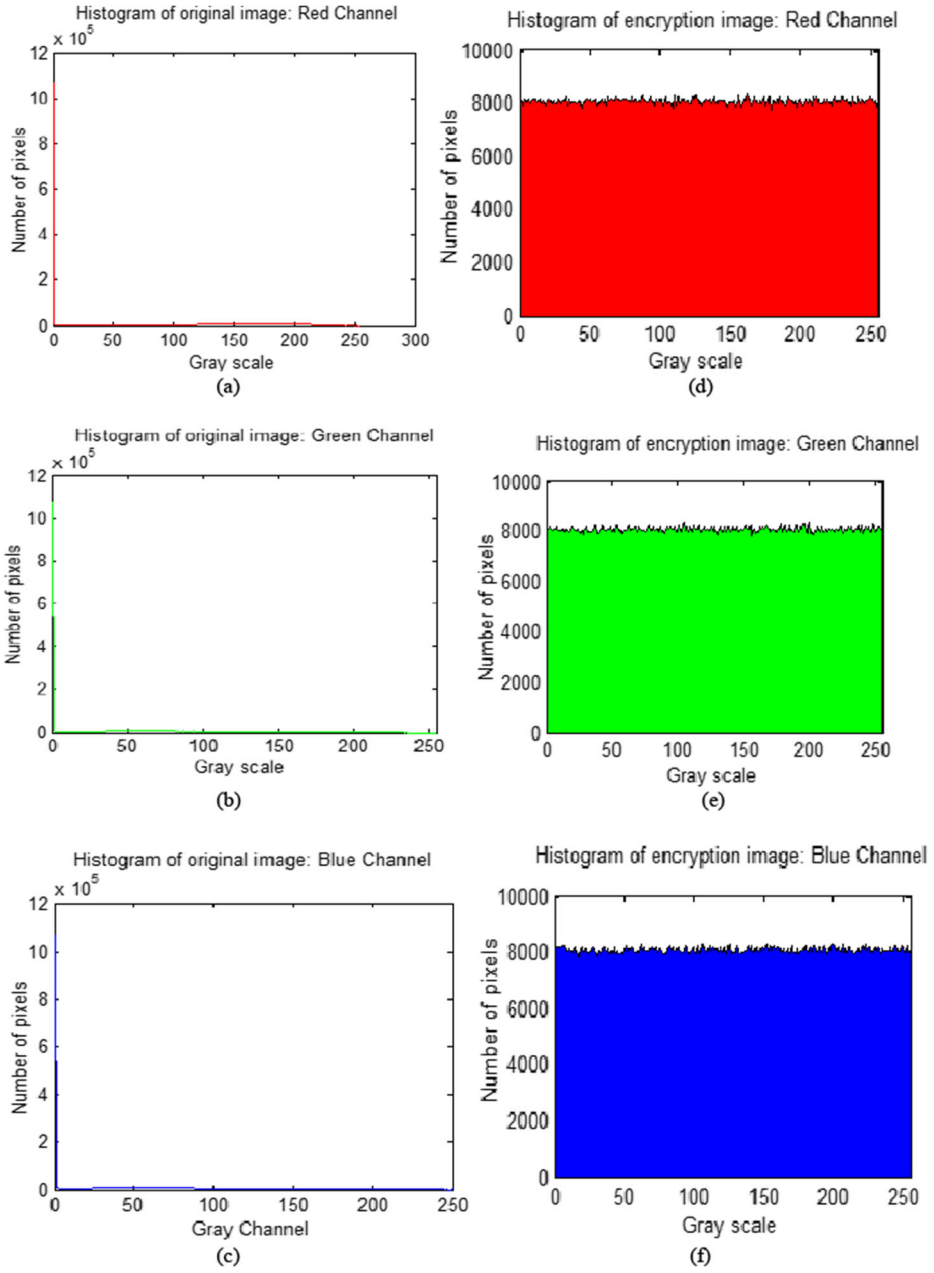


Fig. 13 Histograms illustration of: **a**: Red original image of 'Scanner Ankle', **b**: Green original image of 'Scanner Ankle', **c**: Blue original image of 'Scanner Ankle', **d**: Red ciphered image of 'Scanner Ankle', **e**: Green ciphered image of 'Scanner Ankle', **f**: Blue ciphered image of 'Scanner Ankle'

plotted. Figure 15 shows the distributions of 2000 pairs of randomly selected adjacent pixels of the original and encrypted Lena image, respectively in each channel.

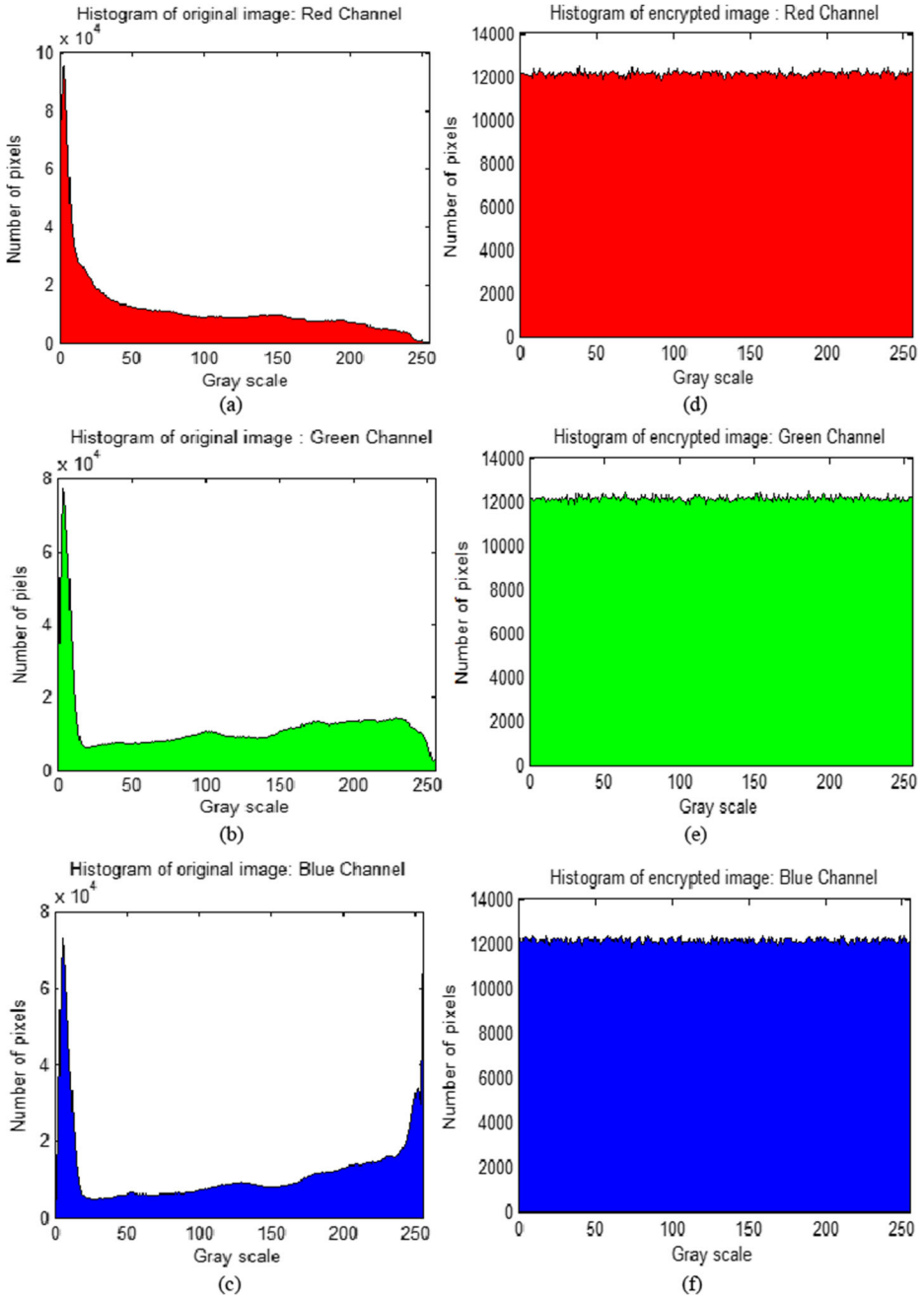


Fig. 14 Histograms illustration of: **a**: Red original image of 'Chest X-ray', **b** Green original image of 'Chest X-ray', **c**: Blue original image of 'Chest X-ray', **d**: Red ciphered image of 'Chest X-ray', **e**: Green ciphered image of 'Chest X-ray', **f**: Blue ciphered image of 'Chest X-ray'

The following equations are utilized for the study of the correlation between two adjacent pixels in the horizontal, vertical, and diagonal orientations for both clear and ciphered images

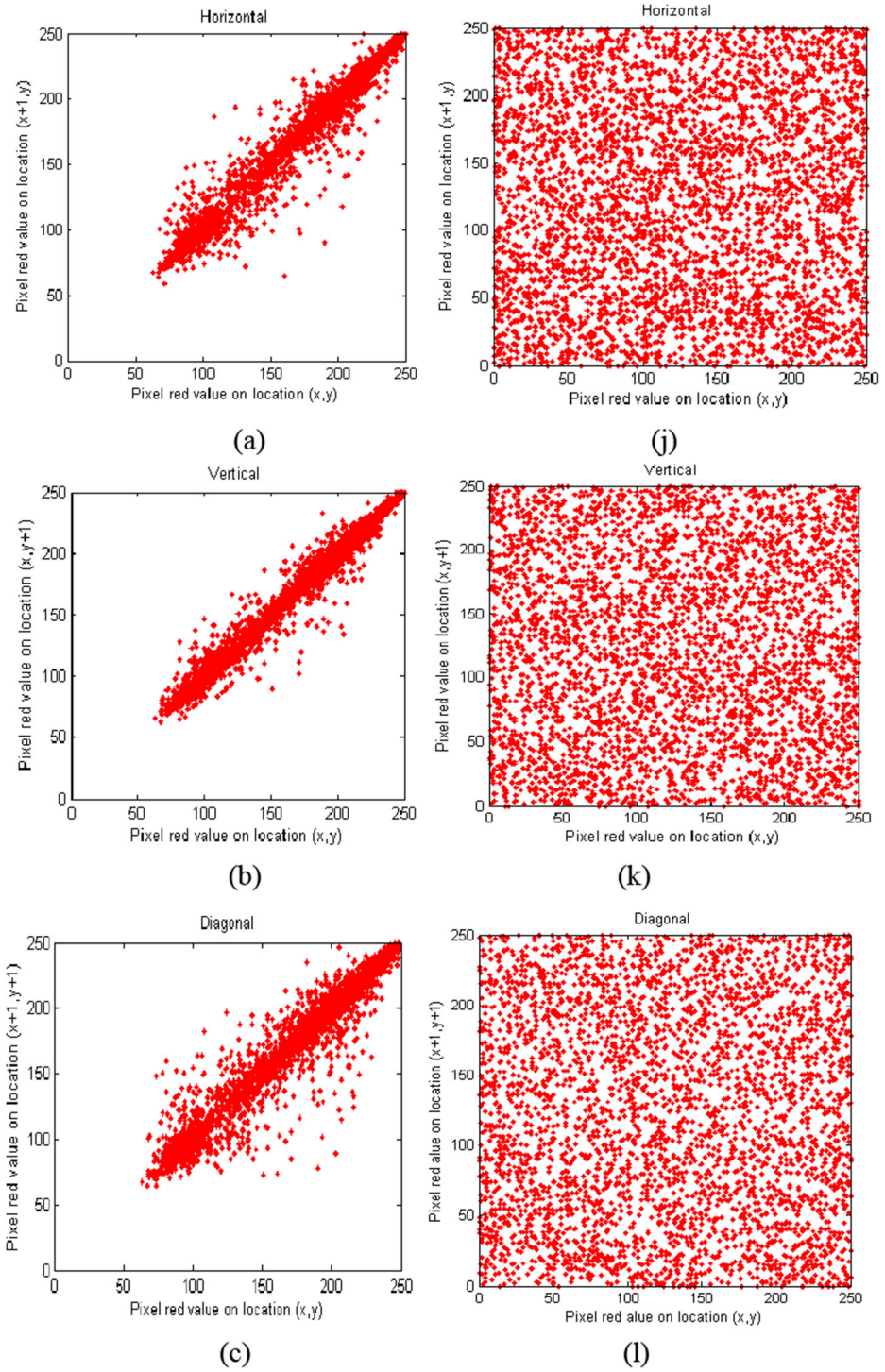


Fig. 15 Correlation distribution of original and cipher 'Lena' color image in horizontal, vertical and diagonal directions: **a-i**: Correlation distribution of original images; **j-r**: Correlation distribution of cipher images

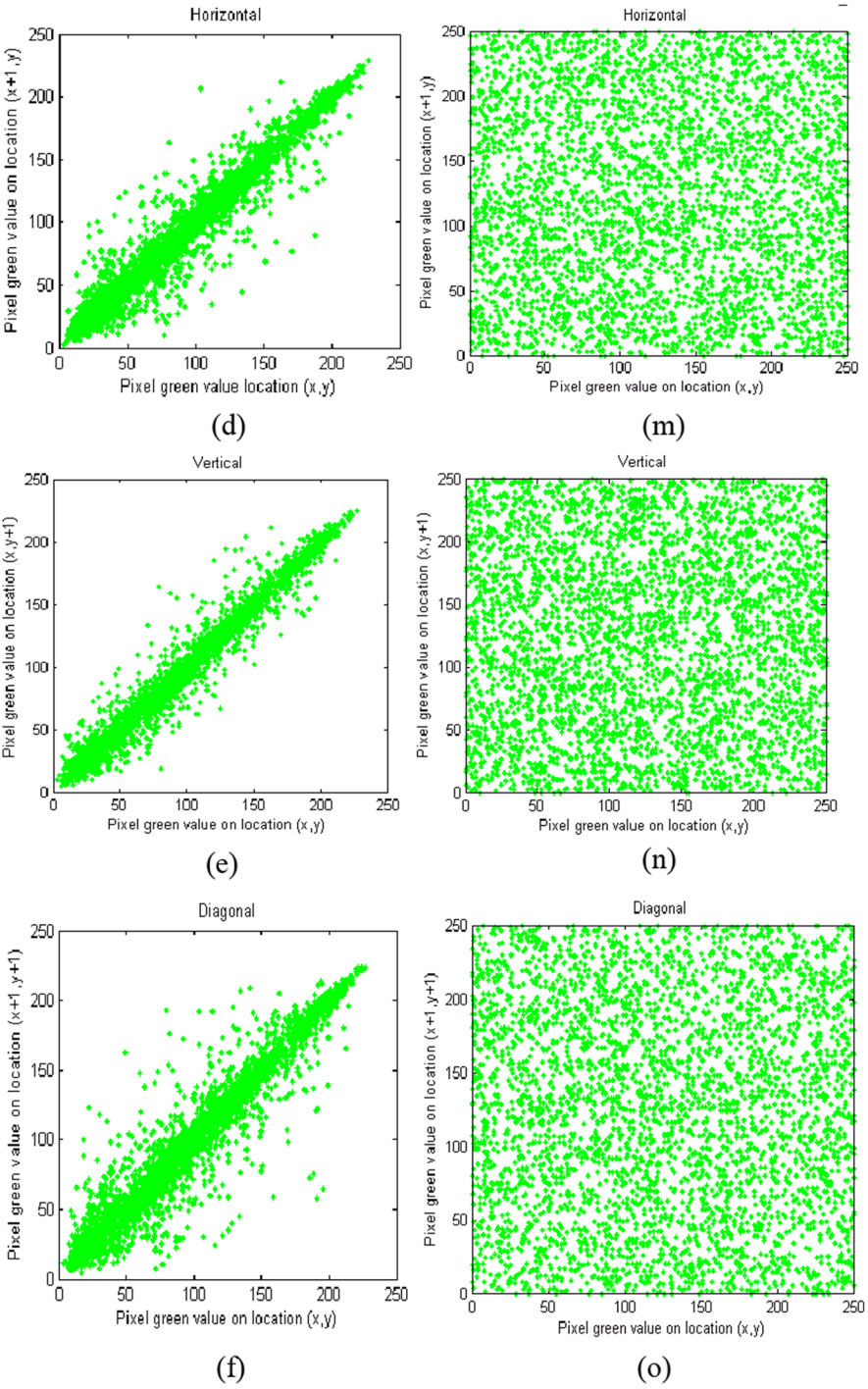
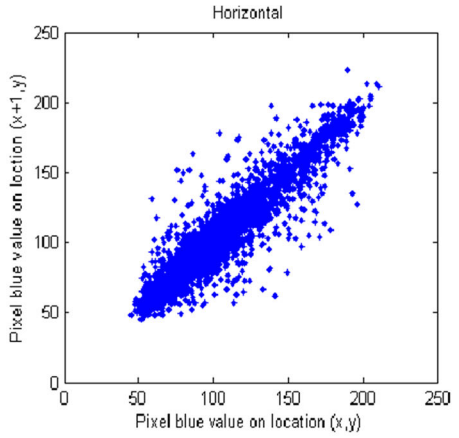
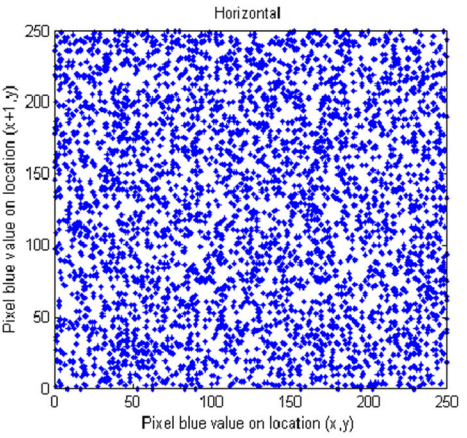


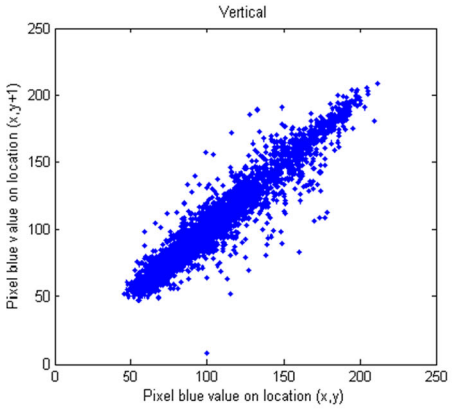
Fig. 15 (continued)



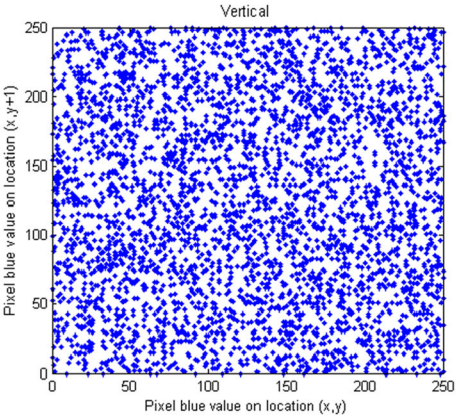
(g)



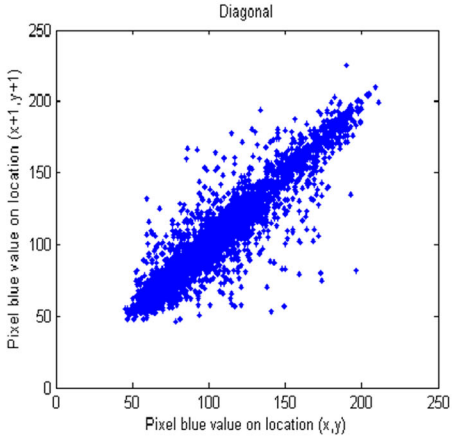
(p)



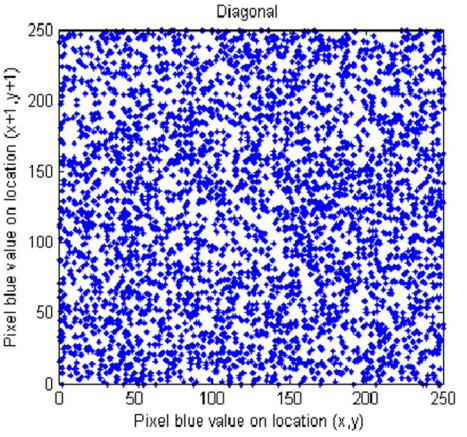
(h)



(q)



(i)



(r)

Fig. 15 (continued)

in each channel.

$$E(x) = \frac{1}{N} \sum_{i=1}^{N} xi \tag{2}$$

$$D(x) = \frac{1}{N} \sum_{i=1}^{N} (xi - E(x))^2 \tag{3}$$

$$cov(x,y) = \frac{1}{N} \sum_{i=1}^{N} (xi - E(x))(yi - E(y)) \tag{4}$$

$$rx,y = \frac{cov(x,y)}{\sqrt{D(x)} \times \sqrt{D(y)}} \tag{5}$$

where x and y are the intensity values of the two adjacent pixels in one image, and N is the number of adjacent pixels chosen to compute the correlation.

We measure the correlation factor between clear and ciphered images in each channel, and the findings are given in Table 3. The findings show that the coefficients are very reduced in the ciphered images in all directions and near to 0. For the 3D medical images, the results clearly demonstrate that the correlation coefficients in the three directions of the original medical images are close to 1. They are characterized by great correlation, while the correlation coefficients of the encrypted images are close to 0. This indicates that there is no correlation between the two images, which proves the strength of the proposed model. On the other hand, the suggested cryptosystem is compared with other methods existing in the literature, and the

Table 3 Correlation coefficient of original and cipher images in every channel

Image	Direction	Original image			Cipher image		
		Red	Blue	Green	Red	Blue	Green
Lena	Horizontal	0.98907	0.98421	0.95700	0.00083	−0.01072	−0.00784
	Vertical	0.97941	0.96681	0.93531	−0.01548	−0.00830	0.01941
	Diagonal	0.96747	0.95264	0.91894	−0.02549	−0.01593	0.01054
Peppers	Horizontal	0.96913	0.97601	0.94882	−0.03619	0.00307	−0.03475
	Vertical	0.96307	0.97335	0.95228	0.00195	−0.06850	−0.00743
	Diagonal	0.95079	0.96504	0.92125	0.01308	0.00242	0.00644
Baboon	Horizontal	0.91453	0.96714	0.98365	0.00796	0.00237	−0.08246
	Vertical	0.90268	0.98403	0.99139	−0.01509	−0.00286	−0.04293
	Diagonal	0.95074	0.95014	0.91396	0.00196	−0.03261	−0.03576
Wash-ir	Horizontal	0.74535	0.76729	0.78036	−0.00308	0.01244x	0.02278
	Vertical	0.78962	0.78908	0.81230	0.01382	−0.00514	−0.01080
	Diagonal	0.67670	0.69995	0.71790	0.01746	0.01104	0.02693
Scanner Ankle	Horizontal	0.9993	0.9993	0.9985	−0.0219	−0.0013	−0.0037
	Vertical	0.9956	0.9984	0.9975	−0.0137	−0.00261	−0.0320
	Diagonal	0.9957	0.9981	0.9976	−0.0259	−0.03441	−0.01093
Chest X-ray	Horizontal	0.9981	0.9971	0.9955	−0.0241	−0.0051	−0.0164
	Vertical	0.9995	0.9990	0.9986	−0.0071	−0.0034	−0.0180
	Diagonal	0.9995	0.9990	0.9987	−0.0043	−0.0336	−0.0108

results in Table 4 prove that the propounded cryptosystem has a better correlation with the smallest coefficients in all directions, hence prove the effectiveness of the algorithm for encrypting large medical images and its capability of resisting statistical attacks.

5.3 Entropy analysis and NC

The entropy parameter is considered as the standard to test randomness. The entropy coefficient is utilized to obtain the incertitude performed in the ciphered image. If the entropy is elevated, confidentiality is higher. It is noted that the utmost entropy value for a greyscale image is eight bits/pixel. The average value for H(m) for numerous preceding studies has been between 7.90 and 7.99. This value depends on the image, the size of the key, and the cryptographic model. Entropy is computed as:

$$H(m) = \sum_{i}^{2N-1} P(mi)\, Log2 \left(\frac{1}{P(mi)}\right) \tag{6}$$

where H(m) is the entropy of the image, $P(m_i)$ presents the probability mass function, and 2 N-1 presents the number of gray levels.

Table 5 denotes the entropy value of different color images with various sizes before and after encryption. The results prove that the entropy values of the suggested cryptographic system are much closer to the ideal case. NC is a performance that evaluates the grade of similitude between two objects. The reduced value of NC indicates that there is no relation between plain and ciphered images. The results show that the NC values are very reduced, which proves the strength of the cryptosystem.

Table 6 compares the entropy value with other encryption algorithms. Our results are more successful than other work, which proves the efficacity of the proposed cryptographic model. The efficient result of the entropy value is due to the modification of the rearrangement of the general structure of the AES.

5.4 Robustness against noise attack

During transmission via the network, the ciphered image can lose information or can be influenced by noise. Various cryptographic systems are sensitive to noise, where a small change in the ciphered image can produce a strong distortion within the deciphered picture. Figs. 16, 17, 18 and 19 show that the deciphered pictures keep the global clear image

Table 4 Correlation coefficient comparison with different encryption methods

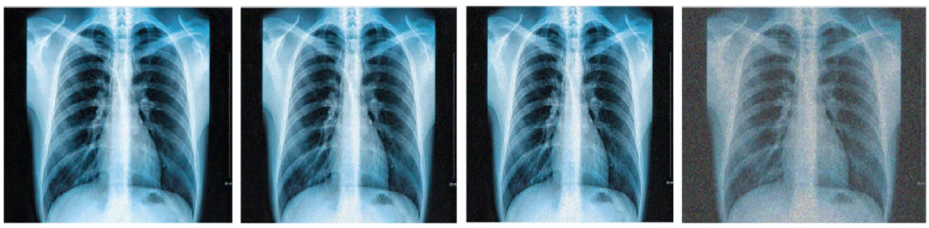
Algorithm	Correlation coefficient		
	Horizontal	Vertical	Diagonal
Proposed method (Lena 512 × 512 ×3)	−0,00591	−0,00145	- 0,01029
[15]	0.004639	0.006763	0.010818
[21]	−0.000400	−0.00180	0.000100
[1]	0.001000	0.001700	0.012500
[17]	0.002500	0.006207	0.003041
[14]	−0.0008	0.0016	0.0043
[23]	0.000751	0.00113300	0.001253
[4]	0.000101	0.00000958	0.000131

Table 5 NC and Entropy values of encrypted images

Image	NC			Entropy		
	Red	Blue	Green	Red	Blue	Green
Lena	−0.00411	−0.00238	0.00065	7.99985	7.99989	7.99985
Peppers	−0.00253	−0.00018	−0.00079	7.99971	7.99979	7.99985
Baboon	0.00274	−0.001912	−0.01348	7.99981	7.99979	7.99977
Wash-ir	−0.00087	−0.00037	−0.00036	7.99998	7.99996	7.99997
Scanner Ankle	−0.00091	−0.00087	−0.00047	7.99999	7.99999	7.99999
Chest X-ray	−0.00041	−0.00042	−0.0015	7.99999	7.99999	7.99999

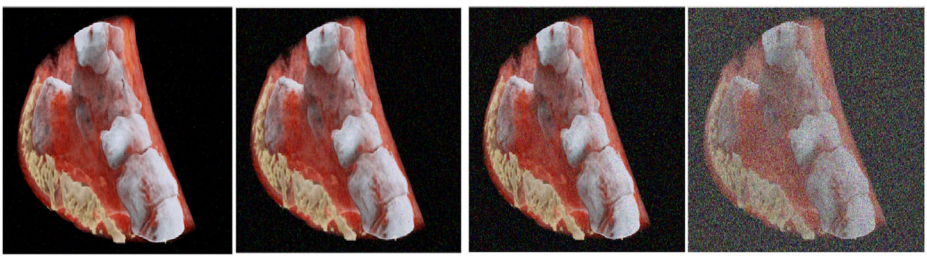
Table 6 Entropy value comparison with the standard AES and existing encryption methods

Algorithm	Cipher image
Proposed Method (Lena 512*512 ×3)	7.9998633
Standard AES	7.8693
[15]	7.9989
[21]	7.9998
[1]	7.9973
[17]	7.9969
[14]	7.9993
[23]	7.999329
[4]	7.9994



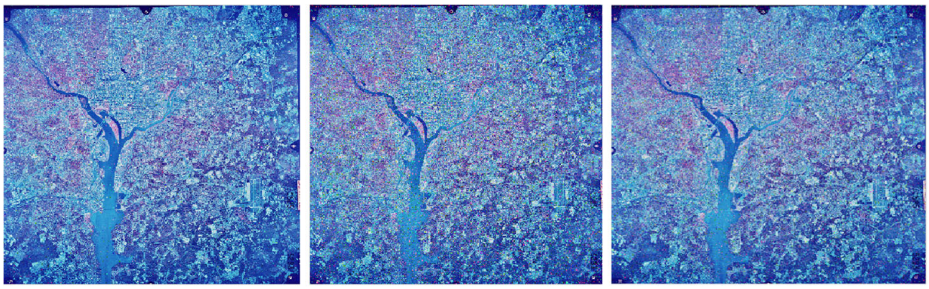
(a) (b) (c) (d)

Fig. 16 Decrypted 3D 'Chest X-ray' image with Salt&Pepper noise: (**a**): d = 0.005, (**b**): d = 0.01, (**c**): d = 0.1, and (**d**): d = 0.5



(a) (b) (c) (d)

Fig. 17 Decrypted 3D 'Scanner Ankle' image with Salt&Pepper noise: (**a**): d = 0.005, (**b**): d = 0.01, (**c**): d = 0.1 and (**d**): d = 0.5



(a) (b) (c)

Fig. 18 Decrypted 'Wash-ir' with Salt&Pepper noise: (**a**): d = 0.01, (**b**): d = 0.1 and (**c**): d = 0.5

information for the human eye when the ciphered picture is affected by Salt&Pepper noise with various percentages. Thus, the suggested method is robust.

5.5 Differential attack analysis

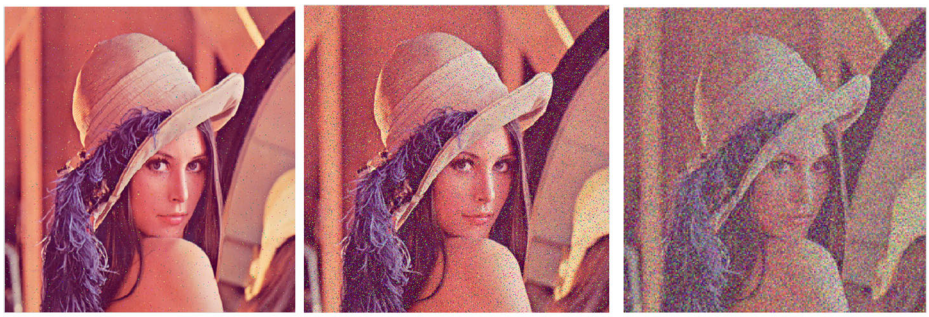
One of the major exigencies to assure the security of one cryptosystem is that a slight change in an original image should result in a prominent change in the encrypted image. Both the NPCR and the UACI are used to measure these changes. Only a one-bit modification over the clear image can result in a considerable modification in the encrypted picture. The NPCR and UACI parameters are presented in eqs. (7) and (8).

$$\mathrm{NPCR} : \mathrm{N}(\mathrm{C1}, \mathrm{C2}) = \sum_{\mathrm{i,j}} \frac{\mathrm{D(i,\,j)}}{\mathrm{W}*\mathrm{H}} *100\% \tag{7}$$

$$\mathrm{UACI} = \mathrm{U}(\mathrm{C1}, \mathrm{C2}) = \frac{1}{\mathrm{W}*\mathrm{H}} \sum_{\mathrm{i,j}} \frac{|\mathrm{C1(i,\,j)} - \mathrm{C2(i,\,j)}|}{225} *100\% \tag{8}$$

where C1 and C2 are the ciphered images**,** M is the size of images, and D is the bipolar matrix determined from C1 and C2.

The NPCR measures the number of pixels that modifies the value in differential attacks. The elevated value is considered better. The UACI computes the average variance between two paired encrypted images, where a minimal value is the best. Table 7 presents the $\mathrm{NPCR}_{\mathrm{R,}}$



(a) (b) (c)

Fig. 19 Decrypted 'Lena' with Salt&Pepper noise: (**a**): d = 0.01, (**b**): d = 0.1 and (**c**): d = 0.5

Table 7 Results of $NPCR_{R, G, B}$ and $UACI_{R, G, B}$ for various color images

Image	$NPCR_{R, G, B}$ (%)			$UACI_{R, G, B}$ (%)		
	Red	Green	Blue	Red	Green	Blue
Lena	99.68109	99.64426	99.70664	33.16517	33.45629	33.80929
Peppers	99.75318	99.75471	99.70664	33.28726	33.39463	33.83002
Baboon	99.64226	99.65950	99.60441	34.03204	33.83002	33.42789
Wish-ir	99.66370	99.75318	99.50981	33.49820	33.64322	33.68199
Scanner Ankle	99.77370	99.8291	99.8370	33.7921	33.79344	33.74291
Chest X-ray	99.87732	99.8946	99.8949	34.0984	34.09613	34.09556

Table 8 Comparison of NPCR and UACI results with existing methods

Algorithm	NPCR (%)	UACI (%)
Proposed method (Lena 512×512 ×3)	99.6773	33.4769
[15]	99.6162	33.3979
[21]	99.60	33.48
[1]	99.50	33.30
[17]	99.6140	33.4805
[14]	99.62	33.41
[23]	99.6112	33.3743
[4]	99.61	33.48

$_{G, B}$ and $NPCR_{R, G, B}$ values for various digital color images and 3D medical images using the propounded cryptographic method. The results prove that the encryption model has a great performance and it is characterized by high sensitivity to any small modification in the clear image. Table 8 compares the NPCR and UACI results using the suggested algorithm with some existing work. The results prove that the proposed cryptographic technique meets the desired objective for resisting differential attacks.

5.6 KPA and CPA

The KPA and the CPA have been utilized to crack some of the cryptographic models including [16]. In general, an adversary utilizes whole black or whole white to discover the possible patterns in the cryptographic model. Thus, whole white and whole dark images are encrypted utilizing the suggested method. Figure 20 presents the ciphered images, and no pattern is apparent. The entropy value of images is similar to other images, and the correlation coefficients are ideal. Table 9 illustrates the correlation of the adjacent pixels and the entropy values of the two images. Because the suggested model utilizes efficient ECC for the key security and

Fig. 20 (**a**) All black, (**b**) All white dark, (**c**) Ciphered all black, (**d**) Ciphered all white

Table 9 Entropy and correlation values

Image	Correlation coefficients			
	Entropy	Horizontal	Vertical	Diagonal
All black	0	–	–	–
All black ciphered	7.9998	−0.0023	−0.0072	0.00204
All white	0	–	–	–
All white ciphered	7.9998	0.0088	−0.020	0.0032

MAES, including XOR operations, permutation and substitution, it guarantees both diffusion and confusion processes, and therefore it is greatly secure against these kinds of attacks.

5.7 Key sensitivity

The key sensitivity analysis warrants the safety of one cryptographic algorithm. An enhanced encryption model should be greatly sensitive to key changes. Similarly, the suggested model must be resistant to the brute-force attack obtained by a large key space. For the MAES, we opt for using a128-bit key size generated by W7 PRNG. It is big enough to resist attacks. To check the encryption process, the plain image is encrypted by three different keys: The first is the main key, the second is the same key with a small change in one bit, and the third is a variance between the two keys. The findings of the three different ciphered images are presented in Fig. 21. Similarly, the ciphered image is decrypted by two keys: One key is original, and the other is modified. The changed key does not allow the retrieval of the clear image, as depicted in Fig. 22. As a result, the suggested model is greatly sensitive to key changes.

5.8 Key Space

An efficient encryption model should have a big key space size. To assure the safety of the proposed algorithm, a 128-bit MAES is employed. This results in a key space large enough against the brute force attack (it requires 2^{128} states to crack the key).

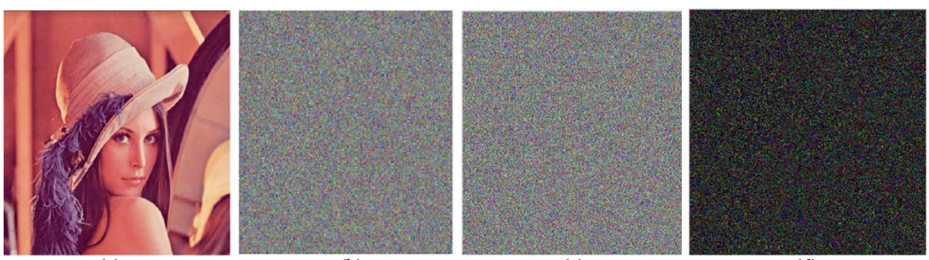


Fig. 21 Test of key encryption sensitivity: (**a**) 'Lena' plain image; (**b**) Cipher image by main key; (**c**) Cipher image by modified key; (**d**) Encryption with key difference between two keys

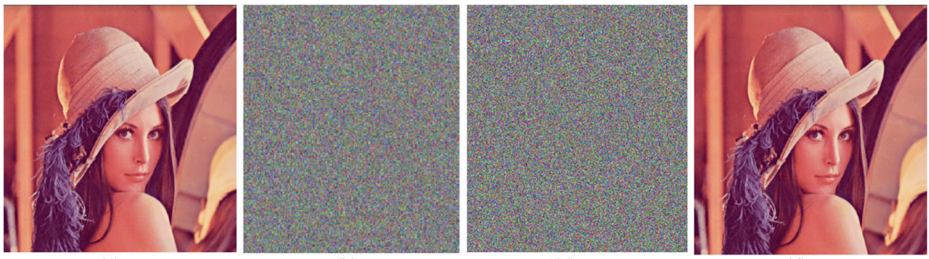


Fig. 22 Test of key decryption sensitivity: (**a**) 'Lena' original image; (**b**) Cipher image by right key; (**c**) Decryption by 1-bit key change; (**d**) Decryption with right key

5.9 Processing time in embedded system

One of the main concerns in image encryption is the encryption time. The processing time of one cryptosystem depends on different factors such as the hardware specs and the optimization of algorithms. The suggested model is implemented on the DE2–115 board featuring Cyclone IV.E FPGA, where the overall MAES and the control of encryption/decryption in ECC run on the NIOS II CPU, while the scalar multiplication operation of ECC is implemented as a hardware accelerator. To reduce the run time, the optimized ECC hardware architecture is only based on two multipliers to develop both the PA and the PD, and the proposed 32-bit multiplier and inverter architectures are based on shifts and XORs. Similarly, by minimizing the arithmetic operations in the MAES, a fast image cryptosystem is gained. To prove the efficiency of our method, the suggested cryptosystem is firstly compared to the standard AES-ECC algorithm implemented on DE2–115 board. According to the results given in Table 10, it is observable that our model is much faster than the AES-ECC cryptosystem.

Secondly, our algorithm is compared to some methods put forward in the literature as [4, 5, 18]. For a reliable comparison, we implement all algorithms on the DE2–115 board and the implementation results are tabulated in Table 11. In [5], an encryption algorithm for medical image protection was propounded. The scheme was a combination of RSA and AES algorithms. In [4], the method consisted in combining the AES and a 2D LAS chaotic map, whereases the authors in [18] used the AES and a logistic map chaotic system for image encryption. Similarly, our algorithm is compared to [2], already implemented on the DE2–115 board. The comparison results prove that the processing time of the suggested model is much less than the existing work. Thus, it is suitable for practical use with secure transmission.

6 Discussion

Through experimental findings and security analysis, it is shown that the histogram of the ciphered image has a uniform distribution and the correlation between pixels is decreased. The entropy value of Lena's standard image is 7. 9,998,633 (close to the ideal value). The variance

Table 10 Comparison of proposed image cryptosystem running time with AES-ECC

Execution time (s)	Proposed method	AES-ECC
(512×512×3) Image size	0.2731	85.29

Table 11 Comparison of proposed image cryptosystem running time with existing methods implemented on Cyclone IV.E FPGA

Execution time (s)	Proposed method	[5]	[4]	[18]	[2]
(512×512×3) Image size	0.2731	102.51	3.35	3.87	0.37

of entropy between the proposed model and the existing AES is 0.13056. Our method augments the entropy by about 17%, compared to the standard one. Thus, more randomness can be created. The suggested cryptographic model has an efficient encryption effect and a big secret keyspace: It can resist to noise attacks with various intensities, to KPAs, to CPAs and to differential attacks. The run time of the proposed scheme is also executed and the findings prove that the suggested algorithm requires much less calculation time than the standard AES-ECC hybrid and other existing work and that it can be implemented in embedded systems. The proposed method is compared with [1, 4, 14, 15, 17, 21, 23]. The technique put forward in [1] had some issues of vulnerability against KPAs. [4, 15, 17, 18, 23] proposed chaos-based algorithms. However, the chaos system suffered from correlation, as mentioned in [21]. The use of the ECC and the standard AES without optimization as well as the use of a sequential way to encrypt the image in [21] caused the degradation of the performance of the system in terms of speed.

7 Conclusion and future work

An improved image encryption scheme based on MAES-ECC for embedded systems is suggested in this paper. The proposed algorithm encrypts large medical images with great security and time efficiency. By minimizing the arithmetic operations in the AES, the propounded MAES is adjusted to resolve the issue of hard calculation, so the best encryption speed is obtained. Then, to improve the entropy value, an adjustment in the rearrangement of the general structure is put forward. In addition, in order to enhance the security of the key, improved ECC is used for symmetric key sharing. Here, the two basic operations in the Montgomery algorithm are optimized to use only two reused multipliers, and field arithmetics are simplified in the calculation. Thus, it can be implemented in practice. After these optimizations, the global cryptosystem is implemented in the DE2–115 board using a Co-design approach, and good results are gained in terms of execution time, area occupation, power consumption, and throughput. However, the utilized NIOS II CPU is a relatively powerful one amongst embedded processors. The security analysis over our method proves that it is resistant to attacks. The entropy, the correlation of adjacent pixels and the histogram of encrypted images are successfully performed, and the findings are promising. In some future work, we aim to propose more methods to keep on minimizing the computational complexity while maintaining a high level of security.

Acknowledgments This work is backed through EuE Laboratory.

Authors' Contributions All the authors have helped to conceive these simulation experiments. Amal Hafsa and Anissa Sghaier have designed and performed such experiments and have written the main part of the

manuscript with the help of Mohamed Gafsi. Amal Hafsa, Anissa Sghaier, Jihene Malek and Mohsen Machhout have contributed to the interpretation of the results, as well as the revision and writing of the paper.

References

1. Abd El-Latif AA, Niu X (2013) A hybrid chaotic system and cyclic elliptic curve for image encryption. Elsevier AEU – Int J Electron Commun 67(2):136–143. https://doi.org/10.1016/j.aeue.2012.07.004
2. Ansarmohammadi S. A, Shahinfar S, Nejatollahi H (2015) Fast and area efficient implementation for chaotic image encryption algorithms. 18th CSI International Symposium on Computer Architecture and Digital Systems (CADS), Tehran, pp. 1–4. https://doi.org/10.1109/CADS.2015.7377788.
3. Atteya A M, Madian A H (2014) A hybrid Chaos-AES encryption algorithm and its implementation based on FPGA. International new circuits and systems conference (NEWCAS), Trois-Rivieres, QC, 2014, pp. 217–220. https://doi.org/10.1109/NEWCAS.2014.6934022.
4. Bentoutou Y, Bensikaddour El H, Taleb N, Bounoua N (2020) An improved image encryption algorithm for satellite applications. Adv Space Res 66(1):176–192. https://doi.org/10.1016/j.asr.2019.09.027
5. Elhoseny M, Ramírez-González G, Abu-Elnasr O M, Shawkat S A, Arunkumar N, Farouk A (2018) Secure Medical Data Transmission Model for IoT-Based Healthcare Systems. IEEE Access, vol. 6, pp. 20596–20608. https://doi.org/10.1109/ACCESS.2018.2817615.
6. FIPS PUB 197 (2001) Advanced encryption standard (AES). Information Technology Laboratory. Computer Security Resource Center
7. Gafsi M, Hajjaji M A, Malek J, Mtibaa A (2020) Efficient Encryption System for Numerical Image Safe Transmission. J Electric Comput Eng, vol. 2020, Article ID 8937676. https://doi.org/10.1155/2020/8937676
8. Ganesh A R, Manikandan P N, Sethu SP, Sundararajan R, Pargunarajan K (2011) An improved AES-ECC hybrid encryption scheme for secure communication in cooperative diversity based wireless sensor networks. International conference on recent trends in information technology (ICRTIT). Chennai, Tamil Nadu. 1209-1214. https://doi.org/10.1109/ICRTIT.2011.5972351.
9. Hafsa A, Alimi N, Sghaier N, Zeghid M, Machhout M (2017) A hardware-software co-designed AES-ECC cryptosystem. International conference on advanced systems and electric technologies (IC_ASET), Hammamet, pp. 50-54, https://doi.org/10.1109/ASET.2017.7983665.
10. Hafsa A, Sghaier A, Zeghid M, Malek J, Machhout M (2020) An improved co-designed AES-ECC cryptosystem for secure data transmission. Int J Inf Comput Secur 13(1):118–140. https://doi.org/10.1504/IJICS.2020.108145
11. Hajajneh T, Mohd B J, Itradat A, Quttoum A N (2013) Performance and information security evaluation with firewalls. International Journal of Security and Its Applications Vol.7, No.6 (2013), pp.355–372. https://doi.org/10.14257/ijsia.2013.7.6.36.
12. Hayajneh T, Ullah S, Mohd BJ, Balagani KS (2017) An enhanced WLAN security system with FPGA implementation for multimedia applications. IEEE Syst J 11(4):2536–2545. https://doi.org/10.1109/JSYST.2015.2424702
13. Koblitz N (1987) Elliptic curve cryptosystems. Math Comput 48(177):203–203
14. Laiphrakpam DS, Khumanthem MS (2017) Medical image encryption based on improved ElGamal encryption technique. Optik 147:88–102. https://doi.org/10.1016/j.ijleo.2017.08.028
15. Lin Z, Liu J, Lian J, Ma Y, Zhang X (2019) A novel fast image encryption algorithm for embedded systems. Multimed Tools Appl 78:20511–20531. https://doi.org/10.1007/s11042-018-6824-5
16. Liu H, Liu Y (2014) Cryptanalyzing an image encryption scheme based on hybrid chaotic system and cyclic elliptic curve. Opt Laser Technol 56:15–19. https://doi.org/10.1016/j.optlastec.2013.07.009
17. Liu J, Ma Y, Li S, Lian J, Zhang X (2018) A new simple chaotic system and its application in medical image encryption. Multimed Tools Appl 77:22787–22808. https://doi.org/10.1007/s11042-017-5534-8
18. Sbiaa F, Kotel S, Zeghid M, Tourki R, Machhout M, Baganne A (2017) High-Level Implementation of a Chaotic and AES Based Crypto-System. J Circuits Syst Comput 26(07):1750122. https://doi.org/10.1142/S0218126617501225
19. Sghaier A, Zeghid M, Masoud CH, Machhout M (2017) Design and implementation of low area/power elliptic curve digital signature hardware Core. Electron 6(2):46. https://doi.org/10.3390/electronics6020046
20. Shankar K., Eswaran P (2016) An efficient image encryption technique based on optimized key generation in ECC using genetic algorithm. Artificial intelligence and evolutionary computations in engineering systems. Adv intell syst comput, vol 394, pp 705-714. Springer, New Delhi. https://doi.org/10.1007/978-81-322-2656-7_64.

21. Toughi S, Fathi MH, Sekhavat YA (2017) An image encryption scheme based on elliptic curve Pseudo random and advanced encryption system. Signal Process 141:217–227. https://doi.org/10.1016/j.sigpro.2017.06.010
22. Wang X, Zhao Y, Zhang H, Guo K (2016) A novel color image encryption scheme using alternate chaotic mapping structure. Opt Lasers Eng 82:79–86. https://doi.org/10.1016/j.optlaseng.2015.12.006
23. Yang CH, Wu HC, Su FS (2019) Implementation of encryption algorithm and wireless image transmission system on FPGA. IEEE Access 7:50513–50523. https://doi.org/10.1109/ACCESS.2019.2910859
24. Zhao Z, Zhang X (2013) ECC-based image encryption using code computing. International conference on communication, electronics and automation engineering. Advances in intelligent systems and computing, vol 181. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-31698-2_121.

Affiliations

Amal Hafsa[1] · Anissa Sghaier[1] · Jihene Malek[1,2] · Mohsen Machhout[1]

Anissa Sghaier
sghaier.anissa@gmail.com

Jihene Malek
Jihenemalek.14@gmail.com

Mohsen Machhout
machhoutt@yahoo.fr

[1] Electronics and Micro-Electronics Laboratory, University of Monastir, Monastir, Tunisia

[2] Department of Electronics, , Higher Institute of Applied Sciences and Technology, Sousse University, LR99ES30 E_E Lab, Sousse, Tunisia