Chaos blended cellular automata on fractals:
the effective way of reconfigurable hardware assisted
medical image privacy

R. Sivaraman 1 · Ajay Vijaykumar 1 · Prem Savarinathan 1 · Avila Jayapalan 1

Received: 11 August 2021 / Revised: 6 December 2021 / Accepted: 10 April 2022 /
Published online: 18 April 2022
© The Author(s), under exclusive licence to Springer Science+Business Media, LLC, part of Springer Nature 2022

Abstract
In more recent times data continues to be generated at a very unprecedented scale. This is a result of the pervasive nature of modern-day digitisation. As such, it is absolutely critical that this data only be accessed by the trusted parties concerned in an effort to maintain the privacy of individuals. One particular type data that could severely compromise the identity and privacy of an individual is ‘medical data’. With a focus on medical images, this work proposes a novel ‘fractalized’ chaos-cellular automata encryption scheme, implemented on Cyclone IV EP2C35F672C6 FPGA, resulting in a hardware-based concurrent security solution. The scheme entails three stages of diffusion, which arise from different mechanisms. In tandem with the diffusion process is the “On the Fly” process of confusion governed by a Linear feedback Shift Register (LFSR), all of which in implemented by applying the nature of fractals. The security architecture occupies 16,351 Logic Elements (LEs) with 230 registers on the target FPGA with the power dissipation of 133.39 mW. Further, the encryption achieves near zero correlation with the average entropy of 15.17156 that ensures the statistical properties. In addition, the security framework requires 12.13 ms to encrypt a 256 × 256 × 16 DICOM image which results in the throughput of 86.44 Mbps. The proposed encryption resists the brute force attack and chosen plain text attack by achieving a very large span of keyspace.

Keywords FPGA · Cellular automata · LFSR · Hardware security · Chaos · Fractals

1 Introduction

The collateral brought forth by rapid digitisation has had far-reaching and permeating consequences previously thought un-heard of. As such, it has led to the appropriation of technologies in many industries in an effort to modernise themselves, with the benefit of improving

✉ Avila Jayapalan
avila@ece.sastra.edu

1 SASASTRA Deemed University, Thanjavur, Tamil Nadu 613401, India
production efficiency and improving the quality of services rendered to their respective customers. In addition to revamping existing industries, it has also led to the rise and development of new industries, a prominent one being telemedicine. Owing to the multitude of restrictions that COVID-19 has brought forth, it has necessitated the rendering of services effectively while minimising physical contact. While initially presenting itself as an unavoidable requirement, it is slowly positioning itself as a service of convenience. As such, telemedicine has bloomed to meet these requirements. Telemedicine can be regarded as the remote dispensation of medical diagnosis [6]. Medical information usually condensed in the form of a patient record is exchanged between medical staff in order to ascertain a patient’s condition. This needs to be done in a secure manner so that the privacy and security of the patient is maintained. Cryptography is one mechanism that can be used to secure information during transmission and storage, to prevent un-authorised third parties from reading the data [13]. Cryptography prescribes a series of steps in forms of mathematical algorithms which encrypt the data. There are many such algorithms that rose to prominence and information security as whole was bolstered due to its inception. A few prominent ones are the Advanced Encryption Standard (AES) [24], Data Encryption Standard (DES) [17] and the Triple DES [25]. These algorithms have been widely used to encrypt data, particularly textual data with satisfactory results. In the scope of images, these algorithms perform if not equally, lesser in comparison to textual data, with several rounds of operation. However, in the context of medical images, the algorithms do not perform as admirably, due to their high prevalence of redundancy, general bulkiness and high correlation.

In order to combat these inadequacies presented by the more ‘traditional’ approaches, several other techniques have been developed to secure medical images. Another consequence of digitisation is the rise in the processing power of computers, which has led to development of more complex, efficient and powerful encryption schemes, extending support to processing bulky data such as Medical Images. A well-tested mechanism to impart security by encryption is by exploiting the pseudo-random nature of chaotic maps and attractors. Chaos maintains a robust degree of randomness, complimented by its high sensitivity to initial conditions. They also exhibit a certain degree of ergodicity and quasi randomness [2].

M. Benssalah et al., have proposed a chaos based elliptical curve cryptosystem for encryption of Medical images in particular. The elliptical curve cryptosystem was revealed to perform similarly to traditional systems RSA, with significantly smaller key sizes rendering it suitable utilisation in resource-constrained devices like FPGAs [5]. Wenting Yuan et al., proposed a digital hyper chaos system which forms part of a double encryption scheme, where an image is encrypted in both its time and frequency domains, resulting in a significantly high key space [27]. Y. Wang et al., were successful in implementing a modified RSA approach, through the use of semiconductor chaos circuit for a voice encryption system [26]. A combination of wavelet transforms and a chaos-based system was proposed by Xiao Feng Li et al., which resulted in minimum distortion during decryption, indicating a string response to noise based attacks. As such the good localisation characteristics of the wavelet combined with the random nature of the chaos map has yielded in a highly secure encryption scheme [14]. A combination of high and low dimensional chaos maps was designed by W.C Qiu et al., which resulted in a key generation technique with a particularly high key space and key sensitivity [18].

Linear Feedback Shift registers have also seen applications in security mechanisms owing to their randomised bit shifting properties. They are high speed binary sequence generators and when implemented on FPGAs, which present its own set of advantages, and is a very efficient
random number generation mechanism [8]. S. Saha et al., have proposed randomised LFSR generator whose output is further randomised to enhance the security afforded by the LFSR. This is said to yield in a cryptographically secure LFSR PRNG which greatly boost the statistical randomness and the resistance toward attack [23]. A combination of chaos and LFSR was proposed by R. K. Koppanati et al with a focus on efficient computational algorithm meant for resource-limited devices [11]. W. Mao et al. proposed a hardware RNG in tandem with a LFSR based scrambler to generate a robust secure key. This unique combination is meant to reduce bias to zero by offsetting the noise sources to generate a very low bias bitstream [15]. S. T. Allawi have developed a LFSR based diffusion scheme for colour images. This is done on a row-by-row basis followed by a column-by-column basis [1]. Following theme of the ‘light weight-ness’, M. Garcia-Bosque et al have proposed bi-chaotic systems constructed from different maps, followed by a LFSR implementation in order to further randomise the random sequences generated [10].

Another prominent way for the generation of random numbers lies with the properties of Cellular Automatas (CAs). The inherent nature of the CA to describe a complex system with a minimally set of relatively simple rules is exploited here. CAs, in comparison to LFSR posses more internal feedback circuits, and as such can be regarded as more complex leading to potentially better random number generation capabilities. A. Banerjee et al., in their work demonstrate the capability of the CA to enhance data security in lesser time as compared to traditional Schemes such as AES. This would benefit resource constrained devices as well [4]. Following this theme is their implementation in the securing of data pertaining to robotic movement developed by M. A. Al-Jabbar et al. This is done by simulating and shuffling keyframes points as defined by a cellular automaton, yielding sufficient strength against static and brute-force attacks [16]. CA based schemes have also seen use as a reliable security mechanism in niche fields such as IoT. A light weight CA is used in medical IoT devices to encrypt data as it does not generate redundant data, thereby lifting the additional constrain places on memory and resource limited devices [12]. Similar to proposed work but with a varying is degree of complexity a hardware-based implementation on the hybridised cellular automata scheme, as an alternative to the traditional AES with an emphasis on the comparison of operational frequency, logic element utilization and overall power consumption [9].

Software implementation of the various encryption schemes afore mentioned to easier to implement, considering the wide array of software development kits available. However, their performance in terms execution complexity algorithms may require higherpower and processing time as opposed to their hardware-based counterparts. A worthy alternative to purely software-based solution lies in Application Specific Integrated Circuit (ASIC). These devices possess a decisive advantage over their software implementation due to their inherent nature of parallel processing. The main disadvantage to this lies in their inflexibility to alter and improve upon existing design as seamlessly as you would have in software. As such, to overcome the roadblocks posed by these existing infrastructures, the proposed scheme for medical image encryption has been implemented on Field Programmable Gate Array (FPGA). FPGAs are flexible in terms of their reconfigurability generally consumes lesser design time. It also features access to certain proprietary IP cores which could significantly bolster the tools available on hand. The inherent nature of the hardware mechanism is that it serves as additional security mechanism by itself by limiting access and reducing the chance of tamperment. There are some of the reasons that a hardware-based FPGA deployment of an encryption scheme would prove to be advantageous over other existing methods.
2 Proposed methodology

This section illustrates the proposed encryption scheme in the block diagram in Fig. 1. The scheme proposes three stages of diffusion across different mechanisms followed by a confusion which occurs simultaneously. This constitutes the “on the fly process” process of confusion. The scheme is proposed for 256 × 256 16-bit grayscale DICOM image and may be scaled up as and when the requirements arise in tandem with more sophisticated hardware execution environments. The fractal structure and whose intrinsic randomness that was incorporated in the scheme is illustrated in Fig. 2. A detailed overview in Fig. 3 is presented along the Finite State Machine (FSM) representation of the overall flow.

The encryption flow as illustrated in Fig. 1 is defined below:

![Block diagram of the proposed encryption scheme](image-url)
Data Assignment

- Step 1: Load 4 consecutive pixels from the pixel array into 4 variables. Each pixel is of length 16 bit.
  - Split each pixel into 4 quartets and shuffle them. The shuffled pixels are now ready for processing.

Diffusion

- Step 2: The chaos Attractor is generated. The three planes are used for the diffusion process.
  - The lower nibble of x-plane is used to generate 4 keys, for the primary diffusion.
  - The lower nibble of z-plane is used as seed for Rule 42 based CA generation.

![Detailed overview of the fractal structure](image)

Fig. 2  Detailed overview of the fractal structure
Step 3: Once the primary diffusion is executed, the secondary diffusion is performed after ca generation.

Step 4: The tertiary diffusion is done by using the lower nibbles of the y-plane and the z-plane respectively.

Rotation

Step 5: The arms themselves are now rotated in direction ad amount determined by a certain control variable in place.

On the fly Confusion

Step 6: As the FPGA allows for parallel processing, the encrypted image is written to the BRAM segment simultaneously, without having to instantiate another stage, which makes up the ‘on the fly’ confusion.

The fractalized approach to the encryption scheme is illustrated in the square fractal structure. The intrinsic structural properties of the Fractal are exploited in implementing the scheme, which adds another layer of complexity, with a particular emphasis toward quantum-based attacks. The fractalized structure boasts a significant structural key sensitivity as elaborated in detail in a later section.

The proposed encryption scheme for medical images takes advantage of the inherent sensitive nature of the fractal structure. The scheme is a tri-layer scheme, comprising of two distinct diffusions stages and followed by an ‘on the fly’ confusion as and when the encrypted image is written into the BRAM of the FPGA. The encryption architecture has been constructed for a 256 × 256 DICOM medical image, which can scale up to meet other image size requirements by approaching the necessary changes at the hardware level.

The proposed encryption architecture is to perform the encryption of 16-bit DICOM images is comprised of several sub modules. These sub modules are discussed in detail and were
accessed and operated and accessed via a FSM. The sub-modules are ordered as follows with their usage:

- Fractal framework
- Cellular Automata for diffusion
- Lorenz attractor for diffusion
- LFSR for confusion

2.1 Fractal framework

Though the square fractal structure is not a physical component in a strict sense, it heavily influences the way in which the encryption scheme is implemented, particularly on how physically the pixels are arranged and encrypted. Fractals can be regarded as self-replicating structures. As such from a certain simple base shape, complex structures may arise, when arranged and stacked in a certain manner. Examples of more complex fractals include the sierpinski triangle \[\]. For the purposes of this work, a relatively simple square fractal structure was used. The advantage in lies in the fact the structure is highly sensitive to change. Even if the attacker was to work out the base structure of the overall fractal, without the nature and direction of evolution, only a brute force approach could suffice to overcome the scheme. This is the novelty of the proposed work, and this kind of approach is regarded to be resistant to even quantum-based attacks.

2.2 Cellular automata rule 42

\[
b(n) = (\sim(b(n))\& b(n + 1)) \oplus (b(n)\& b(n + 1))
\]

(1)

This unit is responsible for the key generation with which the secondary diffusion is performed. The CAs are set of relatively simple expression which may be used to model more complex situations [7, 19]. As such, to formally put it, CAs are set of expressions whose state transition functions are dependent on their neighbouring cells/state. As such they exist as cells. The nature of these dependencies is defined by a set of rules, which mathematically specify the relationships between the cellular automaton unit and its neighbouring cells. For the purposes of this experiment the Rule 42 attractor was employed owing to its excellent random number generation capabilities. Rule 42 based CA is capable of generating a maximal length sequence used to perform the secondary diffusion [22]. The structure of 8-bit CA with rule 42 is depicted in the Fig. 4 and the functional simulation of CA has been presented in Fig. 5.

2.3 Lorenz attractor

Attractors have the capacity to describe physical systems in a deterministic fashion and yet give rise to inherently unpredictable solutions. The Lorenz attractor is most famously attributed to the “Butterfly effect”, owing to butterfly like shape that the attractor generates [3]. This chaotic attractor is derived from the Lorenz System of ordinary differential equations. The attractor as any other attractors has chaotic solutions when initial conditions met and parameters are used. The attractor is essentially a set of chaotic solutions for the Lorenz systems.
chaotic nature of these solutions is exploited in generation of Pseudorandom sequences for the purposes image encryption. The solutions exist in three planes, x, y and z which originally corresponded to change in convection, vertical temperature and horizontal temperature. In the context of the proposed scheme, each of these planes results in an 64 bit output in each of the planes. Different subsections are derived as input and control for other sections of the architecture. Each of the planes are sub-selected for this purpose. The attractor is described through the following equations (Eq. 2, 3 and 4). In addition it was verified with a simulator whose simulation has been shown in Fig. 6.

\[
\frac{dx}{dt} = \sigma(y-x),
\]  
\[
\frac{dy}{dt} = x(\rho-z)-y
\]  
\[
\frac{dz}{dt} = xy - \beta z.
\]

2.4 Linear feedback shift register

The LFSR is capable of generating pseudorandom sequences for which is incorporated into the encryption architecture. It is capable of producing the maximal length sequences of length $2^{n-1}$. 

![Fig. 4 Illustration of 8-bit CA attractor with rule 42](image1)

![Fig. 5 Functional simulation of CA attractor with rule 42](image2)
The LFSR is a mathematically defined circuit specified by a Fibonacci polynomial of degree 16, as is the requirement for a 256 × 256 dimensioned DICOM image. The LFSR pseudo-random sequence contributes to the “On the Fly” confusion, which makes up the third stage in the encryption scheme. The “On the fly” aspect refers to the advantage of this particular confusion approach, is the diffused pixel has been confused and written in to BRAM of the FPGA in a single clock cycle which improves the execution speed. The functionality verification of LFSR is depicted Fig. 7. Since four pixels have been encrypted at a time with the fractal structure, a 14 bit LFSR is sufficient to confuse 16,384 pixels ([256 × 256] / 4 = 16,384).

The first stage of the encryption process is to shuffle the quartet of image pixel bits to resemble the fractal arm like structure, which is the first step in the fractal implementation. Four pixels are processed at tandem and as such the pixels are shuffled between them to increase the overall pixel-independency. As processing is done for four pixels, 4 distinct 8-bit keys are constructed from the 64-bit chaos sequence generated. These keys are concatenated in a certain order to create 4, 16-bit encryption keys to perform the primary diffusion for each of the 16-bit pixels. Another control is sourced from the chaos map to control the nature of the diffusion process, i.e., being a XOR based or a XNOR based diffusion.

The secondary diffusion is made possible with rule 42 based 16-bit cellular automata generator, which yields in 16 unique keys, of which four are extracted to perform the secondary diffusion of the diffused pixels. The seed for the CA is extracted from a subsection of the chaos map generator. The mechanism for assigning a 16-bit CA key from the list of 16 unique values is also sourced from a section of the chaos map key. Once the respective 16-bit CA key as been marked, the secondary diffusion occurs.

A secondary shuffling takes places, where the order of the entire pixels is shuffled among one another, which is a kin to rotating the arms of the fractal in either direction by one stem. The control for the direction of rotation is sourced from the seed of the CA used in the secondary diffusion. A reduction operation is performed on the 16-bit seed value for the CA, the outcome of which is used to determine the direction of fractal arm rotation. The third and final step in the encryption framework consists of the rather unique ‘on the fly’ confusion. A 16-bit Fibonacci LFSR with a non-zero seed value is used. The ‘on the fly’ aspect of the confusion process is deemed as such as the confused information is written directly on the BRAM of the FPGA without having to confuse the image separately and load it subsequently into the memory.
2.5 FSM state operations

- **Initial State**
  - All the variables are initialised:
    - pixel count i = 0.
    - Lorenz attractor initial conditions defined.
    - Linear Feed Back Shift register initiated and feedback defined.
    - The DICOM array DA, containing the entire DICOM image as pixels in array form is loaded.
    - next_state = State 1

- **State 1: Data Assignment**
  - Pixels are loaded 4 at a time, each representing one arm of the fractal.
    - \[ \text{dicom1} = \text{dicom\_array}[i]; \]
    - \[ \text{dicom2} = \text{dicom\_array}[i + 1]; \]
    - \[ \text{dicom3} = \text{dicom\_array}[i + 2]; \]
    - \[ \text{dicom4} = \text{dicom\_array}[i + 3]; \]
  - Pixels are shuffled amongst the arms and stored on pN
    - \[ p1 = \{\text{dicom1[3:0]}, \text{dicom2[3:0]}, \text{dicom3[3:0]}, \text{dicom4[3:0]}\}; \]
    - \[ p2 = \{\text{dicom1[7:4]}, \text{dicom2[7:4]}, \text{dicom3[7:4]}, \text{dicom4[7:4]}\}; \]
    - \[ p3 = \{\text{dicom1[11:8]}, \text{dicom2[11:8]}, \text{dicom3[11:8]}, \text{dicom4[11:8]}\}; \]
    - \[ p4 = \{\text{dicom1[15:12]}, \text{dicom2[15:12]}, \text{dicom3[15:12]}, \text{dicom4[15:12]}\}; \]
  - next_state = State 2

- **State 2: Chaos generation and primary Diffusion**
  - The Lorenz attractor is generated across the x, z and y planes.
    - X-plane is used to generate four 8-bit keys. Pixels are diffused with the keys.
    - \[ dN = pN \oplus (k_n, k_{n + 1}) \text{ or } pN \otimes (k_n, k_{n + 1}) \text{ based on control} \]
  - Z-plane is used as seed for Cellular Automata generation.
  - Z-plane is used for CA address selection
  - Y-plane is reserved for an alternate operation.
  - next_state = State 3

- **State 3: Cellular Automata Generation and secondary Diffusion**
  - Seed from Z-plane is used for Rule 42 based CA generation and is stored in array
    - \[ \text{reg[15:0]} \text{ ca\_array[0:15]} \]
  - Address from Z-plane is used to select the specific 16-bit component from ca\_array
\[ ddN = dN \oplus (ca_{array}[z - plane]) \]

- next_state = State 4

**State 4: Arm rotation**

- The arms (pixels) are rotated, i.e., they are made to shift and take up the position of the adjacent pixel.
- Rotation is done once in either left or right direction, based on control value.

\[
\text{temp_reg} = dd1; \\
dd1 = dd2; \\
dd2 = dd3; \\
dd3 = dd4; \\
dd4 = \text{temp_reg};
\]

- next_state = State 5

**State 5: Chaos based tertiary Diffusion**

- Sections of Z-plane and Y-plane serve as keys for the tertiary diffusion of pixels.

- The results are stored in ans_array

\[
\text{ans_array}[N] = ddN \oplus y - plane \oplus z - plane
\]

- This constitutes the final diffusion operation.

- next_state = State 6

**State 6: Linear Feed Shift Register Confusion**

- This Process occurs simultaneously as the Diffusion, which constitutes the ‘On the fly’ process of confusion. The 16.bit LFSR generates \(2^{16} - 1\) random values, used as address while writing the image into the BRAM of the FPGA.

\[
\text{BRAM_address} = \text{lfsr_output}
\]

- Each time the pixel is written to a random location as specified by \text{BRAM_address}.

- This concludes the Encryption operation.

- If pixel count is lesser the 65,536, the operation is repeated for the next quartet of pixels.

\[
\text{if(pixel\_count < 65,536)}
\]

- next_state = Data Assignment

\[
\text{else if(pixel\_count } \geq 65,536)
\]

- next_state = Final State

**State = Final State**
As the process on a Verilog coded hardware environment cannot be physically halted, it is programmatically halted. This is achieved by the utilization of a self-loop, which can be interpreted as the end of the operation.

Next State = Final State

The decryption flow may be regarded as simple the reverse of the encryption flow process, and as such has been omitted for preventing redundancy.

3 Results and experimental validation

This section outlines the performance analysis of the encryption scheme, conducted over a suite of statistical and hardware tests. Five test DICOM images designated DICOM1, DICOM2, DICOM3, DICOM4, and DICOM5 were used to implement the proposed algorithm. The sample test images are depicted in Fig. 8(a − c) wherein the encrypted images have been presented in Fig. 9(a − c). The images are first converted into their hexadecimal equivalent values, which were used as such in the Verilog script. The strength of the scheme is elaborated further in the section. The proposed scheme was implemented on a Cyclone IV EP2C35F672C6 FPGA, through Quartus II Electronic Design Automation (EDA) tool.

3.1 Histogram analysis

This analysis allows one to perceive the distribution of the intensities of the pixel weights of the image. Each image has a unique pattern, with which the image may be reconstructed. The flat and even distribution indicates the there exists an equal distribution of pixel weights amongst the images [19]. This scattering of the pixel depths in a uniform pattern across the

---

Fig. 8 Test DICOM images

(a) (b) (c)

Fig. 9 Encrypted DICOM images

(a) (b) (c)
entire range of the histogram, a good indicator of performance of the encryption scheme. This is outlined in the Table 1.

Histogram analysis ensures that the encrypted images yielded an approximate flat response which is close to the ideality. This results in the equi-distribution of the pixel values which confirms the adequate diffusion has been employed.

| Original Images | Histogram of Original Images | Histogram of Encrypted Images |
|-----------------|------------------------------|------------------------------|
| DICOM 1         | ![Histogram of DICOM 1](image1) | ![Histogram of Encrypted DICOM 1](image2) |
| DICOM 2         | ![Histogram of DICOM 2](image3) | ![Histogram of Encrypted DICOM 2](image4) |
| DICOM 3         | ![Histogram of DICOM 3](image5) | ![Histogram of Encrypted DICOM 3](image6) |
| DICOM 4         | ![Histogram of DICOM 4](image7) | ![Histogram of Encrypted DICOM 4](image8) |
| DICOM 5         | ![Histogram of DICOM 5](image9) | ![Histogram of Encrypted DICOM 5](image10) |
3.2 Correlation analysis

This is another statistical metric used to ascertain the performance of the encryption algorithm. Correlation can be broadly defined as the relationship between two variables, or in other words, it may be used to compute the degree of similarity between two variables [22]. A cryptosystem can be regarded as good if the encryption scheme is able to efficiently mask all the distinctive such the encrypted image is randomized with high uncorrelation. The correlation coefficient describes the relationship between the neighbouring pixels along the horizontal, vertical and the diagonal directions. For an encrypted image, should it be different from the original plaintext image, then these values should ideally be zero or practically should be as close to zero as possible. The correlation values are normalised from −1 to 1. Should the coefficient be 1, the algorithm is said to have failed with the plaintext and encrypted image being perfectly correlated. In the case of 0, the images are perfectly uncorrelated. When the correlation coefficient is −1, then the encrypted image is said to be the negative of the original image. This information is also presented in the form of correlation diagrams. As expected, the correlation responses along the afore mentioned three directions is expected to be sparse. These values have been computed with the following equations:

\[
C.C = \frac{Cov(x, y)}{\sigma_x \cdot \sigma_y}
\]

\[
\sigma_x = \sqrt{VAR(x)}
\]

\[
\sigma_y = \sqrt{VAR(y)}
\]

\[
VAR(x) = \frac{1}{N} \sum_{i=1}^{N} (x_i - E(x))^2
\]

\[
Cov(x, y) = \frac{1}{N} \sum_{i=1}^{N} (x_i - E(x))(y_i - E(y))
\]

Where:

* x and y are the pixel values of the same pixel location in the encrypted and plaintext images
* C.C is the correlation coefficient
* Cov is covariance of x and y
* Var(x) is the variance at pixel x

The correlation analysis has been conducted for the original and encrypted image. Table 2 depicts the correlation in Horizontal, Vertical and Diagonal directions and Table 3 presents the correlation coefficients.
3.3 Information entropy analysis

Entropy of any random process yields in information about itself. Information entropy is a primary identifier of uncertainty. This metric helps us to determine the inherent entropy or randomness of the algorithm by observing an encrypted image [3]. For a 16-bit depth grayscale DICOM image, the ideal value is 16, but in general, the entropy value of a source is smaller than its ideal value and as such in practicality is not achievable. In any case, if the entropy is lesser than ideal, there always exists a certain degree of predictability. Hence value as closer to the ideal can be regarded as good encryption scheme which minimize the chances of predictability. The entropy values for the 5 test images were calculated and are illustrated in Table 4. From Table 4 it may be observed that the entropy values manage to reach a value as high as 15.1726, which is close to the ideal value of 16, suggesting that the proposed encryption scheme is sound in nature.

3.4 Chosen plain text attack analysis

The proposed encryption scheme utilized the Modulo 2 addition operation for the diffusion in the primary and secondary encryption stage. This necessitates the test to validate the resistance toward chosen plain text attack [21]. This resistance may be verified by applying the following equation:

\[ d_1 \oplus d_2 = E(d_1) \oplus E(d_2) \]  

(10)

| Image      | Horizontal Correlation | Vertical Correlation | Diagonal Correlation |
|------------|------------------------|----------------------|----------------------|
| DICOM 1    | −0.3199                | −0.3392              | 0.2590               |
| DICOM 2    | −0.0374                | −0.0344              | 0.0248               |
| DICOM 3    | −0.0187                | −0.0056              | 0.0037               |
| DICOM 4    | −0.0394                | −0.0240              | 0.0203               |
| DICOM 5    | −0.5291                | −0.5402              | 0.4191               |
If the equation holds true, then this suggests that the encrypted image is vulnerable to attack. However, the equation when applied to the test and the encrypted test images show that the relation does not hold true, implying that the algorithm displays a significant resistance toward chosen plain text attacks. This experimentation has been conducted between the original and its corresponding encrypted images. It has been identified that the proposed encryption is resistant to chosen plain text attack.

3.5 Encryption quality analysis

An encryption scheme is considered effective if in its capacity is able to mask the distinctive features of an image. Prior tests relied on the visual inspection of images, but it is not a sufficient indicator to determine exactly the number of features hidden. As such, to evaluate the scheme in more detail the deviation of pixels values between the plaintext and the encrypted ones are examined to get a better idea of the quality of encryption. To the effect, Maximum deviation, Irregular deviation and Deviation from Uniform Histogram are used deviation parameters [20].

Maximum deviation is calculated by measuring the maximum deviation between plaintext image and its encrypted counterpart. This is calculated using the histogram responses of the plaintext image and its encrypted counterpart. This process is listed as:

- Let $p$ be absolute difference between the plaintext and encrypted histogram responses
- Let $p_i$ be the amplitude of the histogram at point $i$. Maximum Deviation can be calculated as:

$$M_d = \frac{P_0 + P_{65535}}{2} + \sum_{i=0}^{65536} P_i$$ (11)

The higher the value of $M_d$, the more is the deviation of the encrypted image with respect to the plaintext image. Using Eq. (11) the Maximum deviation can be estimated.

Irregular deviation measures how close the statistical distribution of the encrypted image pixels is close to the uniform distribution. An encryption algorithm is expected to randomise all the pixels in an image, which would ideally result in a uniform distribution. As such, of the Irregular deviation is close to the uniform distribution, and then the encryption scheme can be regarded as performing as intended. The irregular deviation may be calculated as:

$$I_D = \sum_{i=0}^{65536} H_{D_i}$$ (12)

Where:

| Table 4  | Results of entropy analysis |
|----------|-----------------------------|
| Encrypted Image | Entropy  |
| DICOM 1    | 15.1721 |
| DICOM 2    | 15.1699 |
| DICOM 3    | 15.1726 |
| DICOM 4    | 15.1711 |
| DICOM 5    | 15.1721 |

Table 4 Results of entropy analysis
• $I_D$ is the irregular deviation
• $H_{D_i}$ is the absolute histogram deviation from the mean value

As a rule of thumb, the smaller the value of irregular deviation, the better is the strength of the encryption scheme. This value indicates how close is the histogram distribution of the encrypted image is to the ideal uniform distribution. The histogram response of an encrypted image must ideally be uniform. Deviation from Histogram gives insight into the deviation from the ideal uniform histogram distribution. Deviation from histogram is given by the equation:

$$D_h = \sum_{i=0}^{65536} |H_{C_i} - H_C| \frac{M \times N}{M \times N}$$

Where:
• $H_C$ is the histogram of the ciphertext image
• $H_{C_i}$ is the value of the frequency of occurrence at index $i$

The encryption quality metrics are applied on the encrypted images and the obtained values have been listed in the Table 5. From the results, it has been evidenced that the proposed encryption scheme has achieved adequate values in the evaluation.

### 3.6 Keyspace analysis

Keyspace is a vital parameter in encryption which decides the strength of the algorithm towards brute force attack. The size of the key to the power of 2 constitutes the keyspace. In general, it has been recommended to have $2^{128}$ bit of keyspace to overcome the key oriented attacks. In this proposed work, the keyspace is not only dependent on the key generators but also depends on the position of pixel bits located in the square tree fractals. Since the square tree fractal has many self-similar structures on evolution, it helps to keep the inputs in any of the squares which bring the complexity for the attacker. It would be difficult for the attacker to guess the position of the input and to try all the possible input combinations on the fractals structure, which requires more time.

In this work, 80 squares have been used to formulate the algorithm in which 64 bits (4 pixels of 64 bits) of input can be stored for encryption. These 80 bits can be kept in 80 squares of the fractals in $80! \times 79 \times 78 \times \ldots \times 1$ ways which generate huge input combinations. Hence, the keyspace has been calculated as, Keyspace = Cellular Automata (16 bits) + Lorenz chaotic system (3 initial conditions & 3 control parameters in IEEE 754 single precision) + LFSR (14 bits) + Fractals positions for input pixels (80 squares) = $2^{16} \times (2^{32} \times 2^{32} \times 2^{32} \times 2^{32} \times 2^{32} \times 2^{32}) \times 2^{14} \times 80!$ which gives the large span of keyspace to resist the brute force attack. Besides, this fractal-based encryption is resistant towards the quantum computing attacks because of the growth.

| Image    | $M_d$  | $I_D$  | $D_h$   |
|----------|--------|--------|---------|
| DICOM 1  | 70,166 | 11,593 | 1.1954  |
| DICOM 2  | 63,435 | 11,927 | 1.0952  |
| DICOM 3  | 68,187 | 12,124 | 1.1812  |
| DICOM 4  | 71,606 | 14,979 | 1.2277  |
| DICOM 5  | 74,928 | 15,149 | 1.2789  |
of square tree fractals in terms of input positions. This work outperforms in key space than other earlier works of [3, 7, 19–22].

3.7 Hardware analysis

Hardware analysis is a prime concern of analysis to estimate the efficiency of the algorithm with area, power and speed requirements. It has three parameters namely hardware utility in terms of LEs and registers, power dissipation with static, dynamic and Input / Output (I/O) power and throughput. Hardware utility comprises of Look Up Tables (LUTs) otherwise called as combinational functions which creates the replica of the encryption algorithm inside FPGA. The collection of LUTs termed as LE. The consumption of LEs exhibit the utilization of FPGA area for the specified algorithm through which the area calculation has been carried out. In addition, power dissipation analysis provides the amount of power dissipated when the algorithm is executed on the FPGA. Among the three power dissipation metrics, dynamic power dissipation is directly deals the power dissipation of the encryption algorithm wherein static and I/Os are FPGA dependent measures. Further, throughput of the algorithm has been calculated through timing analysis which was determined with the Zero Plus digital logic analyzer interfaced with target FPGA. The following Table 6 depicts the hardware analysis of the proposed encryption architecture which in terms compared with the existing FPGA based medical image encryption work [21].

On analysing the results, the proposed encryption architecture consumes very less time than the existing work [21] due to the adoption of fractal four arm structures which encrypts 4 pixels of DICOM at one clock cycle of 50 MHz. Hence, the throughput has been very much increased. Conversely, LE consumption would be high (approximately 7% extra) than the existing work because of the concurrent operations on pixels. Since the throughput has been concerned, it a common fact that area would be increased on FPGA when attempting the hardware concurrency. The highlights of the proposed encryption architecture have given below:

- Fractal structure adopted encryption method to resist brute force attack
- FSM based architecture that suits for any type of FPGA (Device independent)
- Dissipates very low dynamic power dissipation of 2.85 mW on Cyclone EP4CE115F29C7 FPGA
- Inclusion of “on the fly” confusion helps in faster execution time

Table 6 Hardware analysis for the proposed encryption architecture

| Reference        | Ref. [21]                          | Proposed                      |
|------------------|------------------------------------|-------------------------------|
| Target FPGA      | Intel Cyclone EP2C35F672C6          | Intel Cyclone EP4CE115F29C7   |
| Hardware Utility | Combinational Functions            |                               |
|                  | 2480 / 33, 216 (7.4% Utilization)  | 16,351 / 1, 14, 480 (14.28% Utilization) |
|                  | Logic Registers                    |                               |
|                  | 708 / 33, 216 (< 3% Utilization)   | 230 / 1, 14, 480 (< 1% Utilization) |
| Power Dissipation| Static                             |                               |
|                  | 80.49                              | 99.64                         |
|                  | Dynamic                            |                               |
|                  | 153.45                             | 2.85                          |
|                  | I/O                                |                               |
|                  | 44.71                              | 30.90                         |
| Time taken for   |                                    |                               |
| 256 × 256 × 16 bits (ms) | 215.92 | 12.13 |
| Throughput (Mbps)| 4.85                               | 86.44                         |
• Requires only 12.13 ms to encrypt a $256 \times 256 \times 16$ bit DICOM image
• Resistant towards side channel attacks with the adoption of on-chip memory (BRAM)
• Achieves the average entropy of 15.17156 with near zero correlation

4 Conclusion

A fractalized tri-layer encryption scheme for 16-bit grayscale DICOM images was implemented successfully on Cyclone IV EP2C35F672C6 FPGA. The novel fractalized approach has yielded in a formidable encryption scheme, which provides a decisive advantage over other similar implementations. In addition to this, a hardware implementation of the same is an additional layer of security as a wider response to attack vectors. The fractalized approach gives rise to scopes of structural evaluation, thus presenting as an additional blanket of security. Even in the worst-case scenario of the primitive fractal element being compromised, the random evolution pattern retains the security of the algorithm. To that effect, the various performance metrics calculated have revealed the robust nature of the encryption architecture. The hardware analysis has revealed that the architecture has consumed a rather 14.28% of logical elements on the FPGA board with a maximum throughput of 86.44 Mbps. The comparative analysis of image cryptosystem with different fractals structures would be a continuation of this work.

Declarations

Conflict of interest The authors confirm that this work is original and has not been published elsewhere, nor is it currently under consideration for publication elsewhere. The authors believe this manuscript is appropriate for publication and have no conflicts of interest to disclose.

References

1. Allawi ST, Al-A’meri JH (2016) "image encryption based on linear feedback shift register method," 2016 Al-Sadeq international conference on multidisciplinary in IT and communication science and applications (AIC-MITCSA). Baghdad, Iraq, pp 1–4. https://doi.org/10.1109/AIC-MITCSA.2016.7759903
2. Anandkumar R, Kalpana R (2018) "analyzing of chaos based encryption with Lorenz and Henon map,” 2018 2nd international conference on I-SMAC (IoT in social, Mobile, analytics and cloud) (I-SMAC)I-SMAC (IoT in social, Mobile, analytics and cloud) (I-SMAC), 2018 2nd international conference on. Palladam, India, pp 204–208. https://doi.org/10.1109/I-SMAC.2018.8653652
3. Arumugham S, Rajagopalan S, Rayappan JBB, Amirbarajan R (2018) Networked medical data sharing on secure medium – a web publishing mode for DICOM viewer with three layer authentication. J Biomed Inform 86:90–105
4. Banerjee A, Kundu A (2020) Performance analysis of multilingual encryption for enhancing data security using cellular automata based state transition mapping: a linear approach,” 2020 IEEE 1st international conference for convergence in engineering (ICCE). Kolkata, India, pp 458–463. https://doi.org/10.1109/ICCE50343.2020.9290694
5. Bennallah M, Rhaskali Y, Azzaz MS (2018) Medical Images Encryption Based on Elliptic Curve Cryptography and Chaos Theory. International conference on smart Communications in Network Technologies (SaCoNeT). El Oued, Algeria 2018:222–226. https://doi.org/10.1109/SaCoNeT.2018.8585512
6. Bhushan K, Kulkampi P, Sharma ML (2021) Role of telemedicine for Facio-odontogenic pain management during COVID-19 pandemic. J Maxillofac Oral Surg 20(3):409–413
7. Das M, Das KR, Sahu M and Dash R (2019) "Application of Cellular Automata for an Efficient Symmetric Key Cryptosystem," International conference on applied machine learning (ICAML), Bhubaneswar, India, 2019, pp. 21–26. https://doi.org/10.1109/ICAML48257.2019.00012.
8. Datta D, Datta B, Dutta HS (2017) "Design and implementation of multibit LFSR on FPGA to generate pseudorandom sequence number." 2017 devices for integrated circuit (DevIC). Kalyani, India, pp 346–349. https://doi.org/10.1109/DEVIC.2017.8073966

9. Gangadari BR, Ahamed SR (2017) "FPGA implementation of hybrid linear cellular automata based encryption algorithm." 2017 international conference on communication and signal processing (ICCS). Chennai, India, pp 0281–0285. https://doi.org/10.1109/ICCS.2017.8286361

10. Garcia-Bosque M, Sánchez-Azqueta C, Royo G and Celma S, "Lightweight ciphers based on chaotic Map-LFSR architectures," 2016 12th Conference on Ph.D. Research in microelectronics and electronics (PRIME). Lisbon, Portugal, 2016, pp. 1–4, https://doi.org/10.1109/PRIME.2016.7519519.

11. Koppanati RK, Qamar S, Kumar K (2018) SMALL: secure multimedia technique using logistic and LFSR. Second Int Conf Intell Comput Control Systems (ICICCS) Madurai, India 2018:1820–1825. https://doi.org/10.1109/ICCONS.2018.8662840

12. Kumar R, Khan P, Kumar S (2019) "A cellular automata-based healthcare data encryption technique for IoT networks," 2019 IEEE 16th India council international conference (INDICON). Rajkot, India, pp 1–4. https://doi.org/10.1109/47234.2019.9030349

13. Kumar Sharma D, Chidananda Singh N, Noolu DA, Nirmal Doss A, Sivakumar J (2021) A review on various cryptographic techniques & algorithms. Mater. Today Proc.

14. Li X and Zhang Y (2016) "Digital image encryption and decryption algorithm based on wavelet transform and chaos system," 2016 IEEE Advanced Information Management, Communicates, Electronic and automation control conference (IMCEC), Xi’an, China, pp. 253–257, https://doi.org/10.1109/IMCEC.2016.7867211.

15. Mao W, Li Y, Heng C, Lian Y (2017) "Zero-bias true random number generator using LFSR-based scrambler," 2017 IEEE international symposium on circuits and systems (ISCAS). Baltimore, MD, USA, pp 1–4. https://doi.org/10.1109/ISCAS.2017.8050474

16. Mizher MAA-JA, Sulaiman R (2018) Robotic Movement Encryption Using Guaranteed Cellular Automata. Cyber Resilience Conf (CRC). Putrajaya, Malaysia 2018:1–3. https://doi.org/10.1109/CR.2018.8626820

17. Phan RC-W (2007) Reducing the exhaustive key search of the data encryption standard (DES). Comput Stand Interfaces 29(5):528–530

18. QUO WC (2019) "an image encryption algorithm based on the combination of low-dimensional chaos and high-dimensional chaos," 2019 3rd international conference on electronic information technology and computer engineering (EITCE). Xiamen, China, pp 684–687. https://doi.org/10.1109/EITCE47263.2019.9094882

19. Rajagopalan S, Sivaraman R, Upadhyay HN, Rayappan JBB, Amirtharajan R (2018) ONChip peripherals are ON for chaos – an image fused encryption. Microprocess Microsyst 61:257–278

20. Rajagopalan S, Sharma S, Arumugham S, Upadhyay HN, Rayappan JBB, Amirthanjan R (2019) YRBS coding with logistic map – a novel Sanskrit aphorism and chaos for image encryption. Multimied Tools Appl 78(8):10513–10541

21. Ravichandran D, Rajagopalan S, Upadhyay HN, Rayappan JBB, Amirtharajan R (2019) Encrypted biography of biomedical image - a Pentalayer cryptosystem on FPGA. J Signal Process Syst 91(5):475–501

22. Rengarajan A (2020) Chua diode and strange attractor: a three-layer hardware–software co-design for medical image confidentiality. IET Image Process 14(7):1354–1365(11)

23. Saha S, Karsh RK, Amrohi M (2018) Encryption and decryption of images using secure linear feedback shift registers. Int Comm Conf Signal Processing (ICSSP) Chennai, India 2018:0295–0298. https://doi.org/10.1109/ICSSP.2018.8523833

24. Sheikhpour S, Mahani A, Bagheri N (2021) Reliable advanced encryption standard hardware implementation: 32-bit and 64-bit data-paths. Microprocess Microsyst 81:103740

25. Vuppala A, Roshan RS, Nawaz S, Ravindra JVR (2020) An efficient optimization and secured triple data encryption standard using enhanced key scheduling algorithm. Procedia Comput Sci 171:1054–1063

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.