Since January 2020 Elsevier has created a COVID-19 resource centre with free information in English and Mandarin on the novel coronavirus COVID-19. The COVID-19 resource centre is hosted on Elsevier Connect, the company's public news and information website.

Elsevier hereby grants permission to make all its COVID-19-related research that is available on the COVID-19 resource centre - including this research content - immediately available in PubMed Central and other publicly funded repositories, such as the WHO COVID database with rights for unrestricted research re-use and analyses in any form or by any means with acknowledgement of the original source. These permissions are granted for free by Elsevier for as long as the COVID-19 resource centre remains active.
Generating visually coherent encrypted images with reversible data hiding in wavelet domain by fusing chaos and pairing function

Farhan Musanna, Sanjeev Kumar *

Department of Mathematics, Indian Institute of Technology Roorkee, Roorkee 247667, India

A R T I C L E I N F O

Keywords:
3D-Arnold cat map
Logistic equation
Integer Wavelet Transform
Pixel-encoding
Coherent cipher image
Pairing Function

A B S T R A C T

For secure transmission of digital images, existing cryptographic algorithms transform coherent visual information into a noise-like appearance prompting an adversary of the presence of a possible cipher. This paper proposes an algorithm that produces a visually coherent and meaningful cipher image. The proposed algorithm consists of a permutation-substitution subroutine to obtain a partial cipher. The Arnold-3D map does the permutation, and a delayed logistic map performs the substitution in this subroutine. The hiding of this partial cipher is done in the reference image using an integer wavelet transform. The pixels of the partial cipher are embedded in the four sub-bands of the decomposed reference image as 4 to 1-pixel encoding using Cantor-like pairing function. In addition to the lossless encryption scheme, the integer nature of all the sub-bands in the wavelet decomposition and the invertible pairing function facilitates the perfect reconstruction of the reference image. One of the significant novelty of this work lies in the subtle use of simple pairing functions, which prohibits the unnecessary increase in the size of the cipher, thereby reducing the storage and transmission costs.

1. Introduction

In the recent past, technology has become an integral part of our lives. The internet-based communications involve transmitting digital information in terms of images, videos, financial transactions by digital signatures, a classified military database map, the blueprint of a city, and many more. The obvious question that arises is that are these digital transmissions secure? The answer that caters to this question falls under the realm of information security. It emphasizes the importance of the integrity of data as a significant source of concern for individuals and firms, which in turn makes data encryption, a prominent research area. The three basic approaches under the gamut of cryptography are encryption, steganography, and watermarking. Encryption aims at preventing any malicious party to manipulate the data, but at the same time hints at a possible communication between parties. Steganography tries to embed the data into some other text so that we have a concealed original text to inhibit any clues of communication. Watermarking deals with authentication of the data rather than its specific security. Some of the popular algorithms used in cryptography for these purposes are the Improves Proposed Encryption Standards [1], renamed in 1992 to (IDEA), Advanced Encryption Standard (AES) [2], Rivest Shamir Adleman (RSA) [3]. The aforementioned cryptographic algorithms process each bit of the data by performing multiple rounds of s-box substitutions and permutations to produce the final cipher. The design of these algorithms exploit a block cipher based approach. The algorithms are used for multimedia data encryption is used apart from the AES-Electronic Code Book (ECB) mode which has various visual redundancies. Therefore, encryption by this traditional algorithm is quite inefficient and lackluster. The AES-ECB mode works as a block cipher and encrypts the text in blocks. The major limitation involved with this is that similar blocks of plaintexts get encrypted into similar ciphertexts. Fig. 1 clearly shows the security issues with the AES-ECB mode, whereas the stream cipher AES-CTR mode encrypts the image into a noise-like structure. Therefore, one may not prefer this traditional ECB mode for securing digital image communication and rather opt for CTR mode. However, we implement an alternative encryption scheme based on chaotic map that is sufficiently secure based on the comparative analysis of our scheme with the CTR mode on the information entropy of the cipher image. The results based on the entropy values of the partial cipher obtained by the AES-CTR mode and the proposed method is given in Table 1, whereas the comparative analysis of the histogram analysis is done in Fig. 2. Inferring from the results, we would like to assume that, though the CTR mode has a slight advantage over the proposed method, but nevertheless the proposed method is a viable option in terms of cipher generation when it comes to designing an intermediate step in a steganographic algorithm.

Harnessing the intrinsic properties of digital images like inter-pixel redundancy, psycho-visual redundancy, coding redundancy, specialized algorithms were proposed for image encryption in the past. The basic
The idea of any image encryption algorithm is to transform a plain image into a noise-like image. In recent years, quite a few algorithms were developed for image encryption. The encryption is done in spatial as well as in the frequency domain. In the first case, the image is transformed in the existing domain, i.e., spatial domain, whereas the second involves transforming the image into a frequency domain. Some of the spatial domain-based image encryption algorithms composed of the permutation and substitution architecture can be seen in [4–10]. These algorithms utilize tools such as the bit-plane decomposition, the chaotic properties of dynamical systems. A new spatial domain encryption algorithm presented in [11] uses DNA computing and a hybrid genetic algorithm to convert the original image into the cipher image. The frequency-domain encryption methods can be found in [12], which encrypts the data in the Fourier domain. A novel Fractional Fourier
transform method and finite field cosine transform can be seen in [13, 14]. Other wavelet-based and Fourier transform-based encryption algorithms have been introduced in [15,16]. [17] presented a recent development in image security by harnessing modular chaotic maps. The end product of either approach is to generate an incoherent cipher image that conceals any information about the original image. One obvious inference that an adversary can make upon intercepting the noise-like or texture-like images is that it is a cipher and not the original one. Further, a noisy look can prompt the intruders to apply some cryptanalysis on the cipher and can gain information about the original image. As a result, some significant loopholes were observed in various existing algorithms by performing proper cryptanalysis [18–20].

A partial answer to bypass the first step of visual cryptanalysis can be in the form of steganography. Standard data hiding and steganographic algorithms include [21–23], whereas the SVD and wavelet-based watermarking techniques include [24–26]. However, steganography generally imposes restrictions on the size of the embedded data and the size of the cover medium, where the former is way smaller than the latter. This condition is to ensure that each pixel in the cover image contains at most one bit of the secret. A more generalized way of communicating secret image can be given in the form of generating visually coherent cipher images (VCCI) instead of the random noise like cipher images. The initial work done in this direction was proposed in [27], which generated a visually meaningful cipher image (VMCI). It exploits an existing encryption algorithm and then embedding the partial cipher into the vertical, horizontal, and diagonal subbands of the host image in discrete wavelet transform [28] (DWT) domain of the host image. The final coherent image is transmitted over the public channel/cloud with the minimal chance of suspicion from any adversary. The work presented in [29] is a refinement of the previous scheme that aims at minimizing the texture like appearance in the final cipher in addition to the lossless decryption of the original image. The work reported in [30] aims to encrypt only the salient regions of the image to produce the VMCI, thereby reducing the computational effort. A different approach to generate VCCI’s was put forward by Chai et al. in [31], by combining compressive sensing to their encryption mechanism. The approach did manage to reduce the transmission cost. However, the compression involved at the time of transmission did affect the decryption quality of the original image since the reconstruction involved optimization techniques. The two critical points that need to be improved in the above schemes are as follows:

1. The final VMCI size increases by 4 times, i.e., the encrypted version of an image $I$ of size $n \times n$ becomes $2n \times 2n$. This increment in the size of the cipher image further increases the transmission cost by requiring larger bandwidth for transmission and also adds to the storage cost over the cloud.

| Image                                | Entropy (in bits) | Entropy (in bits) |
|--------------------------------------|-------------------|-------------------|
| Sierpinsky                          | 7.9993            | 7.9992            |
| Tux-Linux mascot                    | 7.9993            | 7.9993            |
| Image with text                     | 7.9993            | 7.9993            |
| Hilbert space filling curve         | 7.9993            | 7.9992            |
| Goldhill                            | 7.9993            | 7.9992            |
| Lena                                | 7.9993            | 7.9992            |
| Barbara                             | 7.9993            | 7.9993            |
| Man & Woman                         | 7.9994            | 7.9992            |

Table 1: Entropy analysis of AES-CTR mode vs Proposed method (GBDA).
2. In the process of designing these meaningful ciphers, the host/reference image reconstruction is not performed. Even though the steps involved in the reference image reconstruction are straightforward, the architecture is such that perfect reconstruction is not possible. The main reason for this flaw is that the detailed sub-bands obtained after the host image decomposition are discarded to embed the partial cipher, and once these pieces of information are discarded, there is no way of getting them back.

2. Related work

As mentioned earlier, the pioneering work in this area of generating VMCI was proposed in [27]. In the VMCI generation method, two major phases were considered:

1. Pre-Encryption Phase: This phase involves the encryption of the original image with the permutation-substitution framework by performing pre-existing algorithms. This phase is the traditional phase used in almost all image encryption algorithms which converts all original images into similar-looking noise-like images.

2. Embedding Phase: The major emphasis of this work lies in this phase, as it involves generating the VMCI by implementing an embedding algorithm. The embedding is done by taking a reference image of four times the size of the original image and performing the DWT with a proper filter $K$ on it. This decomposes the image into 4 sub-bands namely: Approximation Detail band $(C_A)$, Horizontal Detail band $(C_H)$, Vertical Detail band $(C_V)$, and Diagonal Detail band $(C_D)$, each sub-band resulting in dimension $n \times n$. The band $C_V$, $C_D$ are modified to embed the pre-encrypted image into them, leaving $C_H$ unaltered. The embedding is done by sequentially extracting the digits of each pixel of the pre-encrypted image. Mathematically, it can be written as

$$C_V(i,j) = \left[ \left( P(i,j) \right) mod 10 \right] \tag{1}$$

$$C_D(i,j) = P(i,j) \mod 10 \tag{2}$$

The work proposed in [29] is an extension of [27] in the sense that it does not override the values in the $C_V, C_D$ part, but presents an expression where the bit-plane representation of the encrypted image is utilized to embed the secret image into the host image. For any $P \in [0,255]$, $P$ can be written in the binary format as $P = \sum b_i 2^i$.

The algorithm extracts 2 sets of three bits each and a tuple of two bits, i.e., $b_h b_h b_h$, $b_h b_h b_h$ and $b_h b_h$ and embeds them in the least significant bits of $C_H, C_V, C_P$. The inverse-IWT is performed to combine the new bands resulting in the final VMCI of size $2n \times 2n$.

Restricting our attention to design a better version of these schemes, we present a novel encryption-data hiding algorithm that not only safeguards the original image from any adversary but also reduces the transmission and storage costs of the final VMCI.

3. Proposed encryption scheme

3.1. Pre-encryption phase

In addition to the generation of the same size VMCI, we also present an image encryption scheme, which is the intermediate step in our algorithm. This step is essential because in case the eavesdropper can decode the VMCI, it should not be able to decipher the cipher image. Our encryption algorithm reads an image $I$ of size $n \times n$ and produces the partial cipher $C$ of the same size at the end of this phase. There are two subroutines involved in this phase: permutation subroutine and diffusion subroutine. These two phases are explained in detail in this section.

3.1.1. Permutation subroutine:

This phase involves a scrambling of the pixel locations in the original image $I$, so as to de-correlate the pixels. This phase is driven by the chaotic Arnold 3-D cat map [32]. The 3-D map is a generalization of its 2-D counterpart

$$\begin{bmatrix} x_{j+1} \\ y_{j+1} \\ z_{j+1} \end{bmatrix} = \left[ \begin{array}{ccc} 1 & p & 0 \\ q & 1 + pq & 0 \\ 0 & 0 & 1 \end{array} \right] \begin{bmatrix} x_j \\ y_j \\ z_j \end{bmatrix} \mod n$$

where $n$ is the dimension of the square image, $(x_j, y_j)$ and $(x_{j+1}, y_{j+1})$ are the original and transformed intensity respectively, $p$ and $q$ are the chaotic parameters. The map is generalized to 3-D by a series of transformations. The first transformation applies the 2-D map on the $xy$ plane, keeping the $z$ plane unchanged by the following transformations:

$$\begin{bmatrix} x_{j+1} \\ y_{j+1} \\ z_{j+1} \end{bmatrix} = \left[ \begin{array}{ccc} 1 & \gamma_c & 0 \\ \gamma_h & 1 + \gamma_c \gamma_h & 0 \\ 0 & 0 & 1 \end{array} \right] \begin{bmatrix} x_j \\ y_j \\ z_j \end{bmatrix} \mod n$$

The second and third transformation applies on the $yz$ and $xz$ plane by the following transformations:

$$\begin{bmatrix} x_{j+1} \\ y_{j+1} \\ z_{j+1} \end{bmatrix} = \left[ \begin{array}{ccc} 1 & 0 & \gamma_y \\ 0 & 1 & \gamma_x \\ \gamma_z & \gamma_y & 1 \end{array} \right] \begin{bmatrix} x_j \\ y_j \\ z_j \end{bmatrix} \mod n$$

To construct the 3-D map, we combine all the three transformations to get the equation in Box I. Since, $\text{det}(ABC) = \text{det}(A)\text{det}(B)\text{det}(C)$, hence, for all parameters $p_c, q_c, q_s, q_z$, the map remains unimodular. Setting $p_c = 1, q_s = 1 \forall a, b \in [x, y, z]$. Setting up these values maintains the symmetry of each individual transformation to get

$$\begin{bmatrix} x_{j+1} \\ y_{j+1} \\ z_{j+1} \end{bmatrix} = \left[ \begin{array}{ccc} 2 & 1 & 3 \\ 3 & 2 & 5 \\ 2 & 4 & 1 \end{array} \right] \begin{bmatrix} x_j \\ y_j \\ z_j \end{bmatrix} \mod n$$

The numerical computations validate the chaotic properties of this 3-D map by calculating the Lyapunov exponents of this map which come out to be $\alpha_1 = 7.1842, \alpha_2 = 0.2430, \alpha_3 = 0.5728$. Since all three Lyapunov exponents are positive, implying the extended map is indeed chaotic.

The main motivation to use this instead of the 2-D map is due to the presence of the leading exponent being larger than its 2-D counterpart’s exponent of $\beta_1 = 2.618033988$. We use only one parameter $q_z$ as a key in our algorithm with its values ranging in the set $\{1, 2, ..., 16\}$. We implement this transform with the series of steps given below:

P1: Input the grayscale image $I$. If the image is not square, then perform a padding to make it a square image. After the padding, we get an image of size $n \times n$.

P2: Extract $d$ blocks of size $d \times d$ from the plane $I$. These blocks are chosen by traversing the plane $I$ from the top left pixel. Reshape these blocks obtained into a cube $B_i$ of size $d \times d \times d$. The original image which had the representation as $I(x, y)$ now has the representation $B_i(x, y, z)$. The schematic of cube generation is given in Fig. 3.
After the permutation we get a shuffled cube \( x \) as follows:

\[
\begin{bmatrix}
x_{j+1} \\
y_{j+1} \\
z_{j+1}
\end{bmatrix} = \begin{bmatrix}
1 + p_1 p_1 q_2 \\
q_1 + p_1 p_2 q_2 + p_2 + p_1 p_1 p_2 q_3 \\
p_1 q_1 q_2 q_2 + q_4
\end{bmatrix} \begin{bmatrix}
x_j \\
y_j \\
z_j
\end{bmatrix} \mod n
\]

Each round of the permutation is carried out by a different Arnold matrix to enhance the security aspects of the cryptosystem.

**P4:** After the permutation we get a shuffled cube \( \tilde{B}_1 \). The phase ends with reshaping \( \tilde{B}_1(x, y, z) \) into \( \tilde{I}(x, y) \), which is done using the following equation:

\[
\tilde{I}(q, r) = \tilde{B}_1(x, y, z) \mod 255
\]

The output \( I' \) of size \( n \times n \) is passed on as an input to the diffusion subroutine for further processing.

### 3.1.2. Diffusion subroutine:

This phase aims to modify the pixel values in such a way that the resulting cipher has a uniformly distributed histogram and maximum entropy, thereby giving a noise-like look. The diffusion algorithm presented here implements a delayed logistic map [33]. This equation is an archetypical example of simple mathematical equations giving rise to a complex mathematical structure, when the dissipative parameter \( \mu \in [3.57, 4] \). The following equations define the delayed logistic map as follows:

\[
x_{i+1} = \mu x_i (1 - x_i)
\]

For \( \mu = 4 \), the equivalent form studied for its statistical properties like auto-correlation, limit cycles etc., is

\[
x_i = \sin^2(2\theta x_i) = \frac{(1 - \cos(2\theta x_i + 1))}{2}
\]

where \( \theta \in [0, \pi] \) satisfies \( x_0 = \frac{1}{2}(1 - \cos(2\pi \theta)) \). It was shown in [34] that points from this chaotic sequence at a sampling distance of \( s = 15 \) are statistically approximately independent and identically distributed. This property is necessary for randomness and incoherence. Fig. 4 presents 3 subfigures illustrating the following three properties:

- The orbit of the chaotic points generated by the map with \( \mu = 3.9 \), can be seen to haphazardly, indicating chaos.
- Cobweb diagram of the logistic map showing a filled out plane, which implies that there is an infinite number of non-repeating values dense in the unit interval [0, 1].
- The bifurcation diagram suggests that period-doubling cascade starts beyond \( \mu = 3.57 \) (\( \mu \) is the Bifurcation parameter against which we plot to check the chaotic nature of the map) and which implies the onset of chaos.

The steps involved in the diffusion subroutine are as follows:

**D1:** The logistic map (4) is iterated \( n \times s + k \) times, where \( s = 15 \) is the sampling distance, and \( k \) is the threshold limit to discard the initial iterates of the chaotic map. The iterates are properly digitized using the equation:

\[
X_i = (\lfloor 10^4 x_i \rfloor) \mod 255
\]

The vector is reshaped to form a matrix \( X \) of dimensions \( n \times n \).
D2: An intermediate cipher $T$ is generated by the equation

$$T = (\mathcal{I} \oplus X)$$

where $\oplus$ denotes the bit-wise XOR operation.

D3: The partial cipher $C$ is obtained by implementing a Grid-based Diffusion Approach (GbDA) [34] on the temporary cipher given by the following equation:

$$C(x, y) = \begin{cases} 
\frac{\text{sum}(I) \mod 256}{256} \oplus T(x, y); & \text{for } x = 1, \ y = 1 \ldots n. \\
C(x - 1, y) \oplus C(x - 1, y + 1) \oplus T(x, y); & \text{for } x = 2 \ldots n, \ y = 1. \\
C(x, y - 1) \oplus C(x, y) \oplus C(x, y + 1) \oplus T(x, y); & \text{for } x = 2 \ldots n, \ y = 2 \ldots n - 1. \\
C(x - 1, y - 1) \oplus C(x - 1, y) \oplus C(x, y - 1) \oplus T(x, y); & \text{for } x = 2 \ldots n, \ y = n. 
\end{cases}$$

where $\oplus$ denotes bit-wise XOR, and $\text{sum}(I)$ is inserted in the head data to be transmitted. The set of equations in (8), are imperative to the algorithm, since these are the equations that implement the diffusion of the intensities in the cipher and generate the uniform looking histogram of the partial cipher. The diffusion phase is set for a maximum of $\text{ite}_{17} = 17$ rounds, in which the value of the pair $(x_{17}, \mu)$ is changed until the second round. Fig. 5 gives the encryption results on the four images used in Fig. 1.

3.2. Encoding phase

In this phase, VMCI is generated by applying the Integer Wavelet Transform (IWT) and Pairing function. This phase has two sub-routines in it, a) Encoding sub-routine, b) Embedding Phase. The encoding phase uses the Cantor-like pairing function to encode 4 pixels to a single-pixel intensity in a reversible manner. A pairing function [35, 36] in mathematics is an invertible function constructed to show the equinumerosity of the sets $\mathbb{N}^2$ and $\mathbb{N}$. A modified and elegant pairing function [36] that pairs the two numbers depending on the magnitude of each number is defined as

$$\sigma : \mathbb{N} \times \mathbb{N} \rightarrow \mathbb{N}, \text{such that} \quad \sigma(n_1, n_2) = \begin{cases} 
 n_1 + n_2^2, & \text{if } n_1 < n_2 \\
 n_2 + n_1 + n_2, & \text{otherwise.}
\end{cases}$$

Fig. 6 gives the schematic description of Eq. (9). A justification for the invertibility of the pairing function is given in the Appendix section. The steps involved in the encoding are given as:

E1: Reshape the partial cipher $C$ into a vector $e$ of length $n^2$, and divide it into two sub-vectors $e_1, e_2$ of length $n^2/2$ each, in the following manner: $e_1$: all odd indexed elements and $e_2$: all even indexed elements.

E2: Join these two vectors $e_1$ and $e_2$ into a single vector $e_1$ element-wise as follows:

$$e_1(k) = \begin{cases} 
e_1(k) + e_2(k)^2, & \text{if } e_1(k) < e_2(k) \\
e_1(k)^2 + e_1(k) + e_2(k), & \text{otherwise.}
\end{cases}$$

E3: Input a host image $H$ of size $n \times n$, apply the Integer Wavelet Transform (IWT) to get the 4 sub-bands namely: Approximation part (AH), Horizontal Detail part (HH), Vertical Detail part (VH), Diagonal Detail part (DH). To get an integer to integer mapping, we use the lifting scheme [28] of the wavelet transform.

E4: The HH, VH, DH subbands contain values from $\mathbb{Z}$, before applying the Pairing function we introduce a pre-processing module that involves mapping of $\mathbb{Z}$ to $\mathbb{Z}^+ \cup \{0\}$. For this we deploy a bijection function between $\mathbb{Z}$ and $\mathbb{Z}^+ \cup \{0\}$ as

$$f : \mathbb{Z} \rightarrow \mathbb{Z}^+ \cup \{0\}$$

$$f(x) = \begin{cases} 
-2x, & \text{if } x \leq 0 \\
2x - 1, & \text{otherwise.}
\end{cases}$$

Remark Since, the maximum value of a pixel in the partial cipher can be 255 so the maximum outcome of the first pairing function can be $2^{16} - 1$ which is the highest value that an unsigned integer 16-bit can hold. Similarly, the maximum from the second pairing function can be $2^{32} - 1$ which is the highest value that an unsigned int 32-bit can hold. Since dealing with such high magnitude numbers would slow the algorithm, therefore we decided to divide these values by $10^8$ which gives 42.99467295. Further this factor of $10^8$ serves as a key for our algorithm as we need to extract the digits from this as 4,2 and the fractional part.
Fig. 5. Pre-encryption algorithm results: Original images are shown (left to right) in the first row, and the second row illustrates the corresponding cipher images by the proposed pre-encryption subroutine.

Fig. 6. Schematic of the pairing function defined in Eqs. (9) and (10).

0.94967295. A factor of $10^8$ would make the digits as 0.4 and the fractional part as 0.94967295, which would substantially affect the further embedding process. Now, without this factor of $10^8$ we would not be able to divide the numbers and distribute it to the sub-bands of the reference image. For instance, if we were to divide $2^{32} - 1$ say using the scheme 429,496, and 7295 and tried to embed them in the sub-bands, then this would distort the image. Fig. 7 illustrates the effect of $DF$ in the final output and the choice of taking $DF = 10^8$.

3.3. Embedding phase

E5: Generate matrices $G_1, G_2, G_3, G_4, G_5$ in the following manner:

\[ G_1 = \lfloor W_1 \rfloor \]  \hspace{1cm} (12)
\[ G_2 = W_1 - G_1 \]  \hspace{1cm} (13)

\[ G_3 = \lfloor G_1 / 10 \rfloor \]  \hspace{1cm} (14)
\[ G_4 = G_1 \mod 10 \]  \hspace{1cm} (15)
\[ G_5 = H_H + G_2 \]  \hspace{1cm} (16)

where $\lfloor . \rfloor$ denotes the floor of the values taken elementwise.

E6: Apply the pairing function to the pairs $(V_H, G_3)$ and $(D_H, G_4)$ obtained in step E3 and E5 to obtain $V_{H_{new}}, D_{H_{new}}$.

E7: Apply the inverse-IWT on $A_H, G_5, V_{H_{new}}, D_{H_{new}}$ to get the final VMCI $C$.

The schematic representation of the proposed encryption scheme is illustrated in Fig. 8, and the execution of the proposed scheme on a $2 \times 2$ matrix is shown in Fig. 9.
Fig. 7. The effect of choosing a proper digitizing factor: First row (Left to right) VMCI with $DF = 10^2$, VMCI with $DF = 10^3$, VMCI with $DF = 10^4$. Second row (Left to right) VMCI with $DF = 10^5$, VMCI with $DF = 10^6$, VMCI with $DF = 10^7$, VMCI with $DF = 10^8$.

Fig. 8. Schematic of the Encryption Algorithm.

Table 2

| Parameter symbol | Parameter definition |
|------------------|----------------------|
| $\tilde{C}$     | VMCI                 |
| $x_0$            | initial state of the logistic map with values in (0, 1) |
| $\mu$            | Chaotic parameter $\in (3.57, 4)$ |
| $q_i$            | Parameter of the Arnold 3-D map, with range {1, 2, …, 16} |
| $s$              | Delay parameter in the Logistic equation with range {15, 16, …, 30} |
| $\sum(I(i)) \mod 256$ | Sum of all the pixels in the original image modulo 256, with range {0, 1, …, 255} |
| $i_{e_A}$        | Iterations of the Arnold 3-D map, with range {2, 3, …, 9} |
| $i_{e_D}$        | Iterations of the Diffusion scheme {2, 3, …, 17} |
| $DF$             | Digitizing factor used in the pairing process with value $10^8$ |
| $\lambda$        | Wavelet used in the decomposition of the reference image |
3.4. Decryption algorithm

As mentioned earlier, the proposed scheme guarantees for the perfect reconstruction of the original image and the cover image. The entire encryption-decryption process can be mathematically expressed as inverse processes to each other given by the following functions:

\[ E : O \times R \times K \to C \quad \text{and} \quad D : C \times K \to O \times R \]

(17)

where,

\[ E(O, R, K) = C \]

(18)

\[ D(C, K) = (I, O) \]

(19)

**E**: Encryption function, **O**: Set of all plain text/Original Images to be encrypted, **R**: Set of all reference images that serve as cover images, **K**: Set of all possible keys for encryption, **C**: Set of all possible cipher images, **D**: Decryption function.

For a perfect reconstruction scheme \( I = \hat{I} \), to verify the perfect reconstruction claim, we give Algorithm 1, a schematic is given in Fig. 10 that supports our claim that \( I = \hat{I} \). Moreover, we have \( R = \hat{R} \), i.e., the decrypted original and the reference image are recovered perfectly. An important step in the Algorithm is Step 49, which partially reconstructs the original image and acts as an inverse of step E5 of the Embedding Phase. The list of functions and symbols used in the algorithm are given in Table 2.

4. Key-space analysis

The security of the proposed cryptosystem is based on: (1) large key space to resist brute force attacks; (2) strong permutation property to decorrelate pixels; (3) large amount of diffusion to give the partial cipher a random appearance. The key space of the proposed encryption scheme, as calculated below, is \( 2^{277} \). To provide sufficient security against brute-force attacks the key-space should be \( > 2^{128} \) as given by Alvarez et al. in [37]. An increase in the number of keys used in the algorithm enlarges the key space of the system, and thus makes it more secure against brute force attacks of any adversary. With the advances in technologies and computer speeds, any algorithm with a key space of less than \( 2^{128} \) is termed as insecure [38]. For the proposed algorithm, the key-stream can be considered as a vector \((x_0, \mu, q_x, s, sum(I(:,)), Ite_A, Ite_B, DF, wav))\), where the definition of each symbol is given in Table 2. Floating point precision is chosen as \( 10^{-15} \) to comply with IEEE-754-binary64 standards [39]. Thus the total key-space of the algorithm becomes

\[ Key - Space = (10^{15} \cdot 10^{15}) \cdot (10^{15} \cdot 10^{15}) \cdot 2^{27} \cdot (10^8 \cdot 10^{15}) \cdot 37 \approx 2^{277} \]

(20)

Since the key-space of the algorithm can be seen to be much larger than the threshold, the encryption can be termed as safe against brute-force attacks.

5. Results and discussions

5.1. Encryption analysis

This section presents the qualitative and quantitative results to validate our claims of generating a secure VMCI. We use ‘Lena’, ‘Elaine’, ‘Boat’ as the original image, and ‘Mandril’, ‘Man and Women’, ‘House’ as the reference image. The size of each image in this study is \( 512 \times 512 \). These original and reference images are shown in Fig. 11. The encryption results of the proposed algorithm are compared against [27,29]. The execution of all experiments have been done on Matlab R2019a platform with a PC having Intel(R) Xeon(R) CPU E5-2620-0 processor and 16.0 Ghz RAM.

(i) Execution Time Analysis: The execution time of the proposed scheme depends upon two factors, (i) the execution time of pre-encryption subroutine, and (ii) the VMCI generating phase. Since the focus of this study is efficient VMCI generation, we give the results based on the embedding and the VMCI generation phase. For this, the original images of sizes \( n \times n \) have been taken, where \( n = 256, 512 \), the corresponding VMCI generated are of the same size as claimed. The results obtained in Table 3 presents with the numerical values for the embedding phase. We also perform the time complexity analysis comparison against
Algorithm 1 Proposed decryption algorithm.

1: function DECRYPTION_ALGORITHM(Δ(C, wav, μ, q0, s, t, iteA, iteB, DF)
2: LS = if(‘hasr’, double)
3: [A, H, V, B] = inv2(C, LS)
4: for i = 1 : n/2 do
5: for j = 1 : n/2 do
6: if (H(i,j) < 0) then
7: TR(i,j) = |H(i,j)|
8: TI(i,j) = TR(i,j) − H(i,j)
9: else
10: TR(i,j) = |H(i,j)|
11: TI(i,j) = H(i,j) − TR(i,j)
12: end if
13: end for
14: end for
15: [TVR, TV1] = ICP(Y)
16: [TDRI, TD1] = ICP(D)
17: The function ICP is defined as
18: function INVERSE CANTOR PARING :ICP(X)
19: Max X(s) = |X|
20: for i = 1 : Max do
21: if X(i) < [\sqrt{X(i)}]^2 then
22: T = [\sqrt{X(i)}]
23: B(i) = T
24: A(i) = X(i) − T^2
25: else
26: T = \left\lfloor\frac{1+\sqrt{4+X(i)}}{2}\right\rfloor
27: A(i) = T
28: B(i) = X(i) − T^2 − T
29: end if
30: end for
31: Return A, B
32: end function
33: TVRINV=IP(TVR)
34: TDRIINV = IP(TDR)
35: THRINV = IP(TR)
36: The function IP is defined as
37: function INVERSE PROCESSING :IP(X)
38: for i = 1 : n^2/2 do
39: for j = 1 : n^2/2 do
40: if ((X(i,j) mod 2) == 0) then
41: A(i,j) = −X(i,j)/2
42: else
43: A(i,j) = (X(i,j) + 1)/2
44: end if
45: end for
46: end for
47: Return A
48: end function
49: COMBINED = TV1 * 10 + TDI + TI
50: for i = 1 : 2 do
51: if i==1 then
52: [A, B] = ICP(round(COMBINE1 * 10^8))
53: COMBINE1D = A|B
54: else
55: [A, B] = ICP(COMBINE1D)
56: COMBINE1D = A|B
57: end if
58: end for
59: Reshape C into matrix of size n x n
60: ID = Inverse Diffusion(C, ite_p), Inverse of equation (8) and (7)
61: D = Inverse ARNOLD3D(ID, q, s, t, ite_p), Inverse of equation (8) and (7)
62: R = inv2(A, THRINV, TVRINV, TDRINV, LS)
63: Return D, R
64: end function

Table 3

| VMCI size | Time (in seconds) |
|-----------|--------------------|
| 256 x 256 | 0.228292           |
| 512 x 512 | 0.608653           |

Table 4

| Image   | NPCR |
|---------|------|
| Lena    | 99.4171 |
| Elaine  | 99.4160 |
| Cameraman | 99.5594 |
| Peppers | 99.3996 |
| Couple  | 99.2661 |
| Scenery | 99.4392 |

The algorithms reported in [27] and [29], to highlight the efficiency of the proposed scheme. The time complexity results are the average of the results obtained after performing each test on 50 different images. For schemes [27] and [29] to get a VMCI of size 512 x 512, the input reference image is of size 1024 x 1024, however in our scheme it remains 512 x 512. Therefore, the final time of the embedding and VMCI generation given in Table 3 is based on the size of the final VMCI. As can be seen, the algorithm takes comparatively lesser time in generating the VMCI, than its counterpart algorithms.

(ii) Histogram analysis: The main objective of generating VMCI, is that the final encrypted images should look structurally coherent and visually meaningful. Contrary to this, traditional cryptosystems convert the original image into a noise-like appearance, giving a uniform looking histogram of the final cipher. The histogram of the VMCI is plotted in the last row of Fig. 12, which represents a meaningful/coherent image.

(iii) Original Image Variation attack: This refers to the change in the cipher corresponding to a minuscule change in the original image. For a secure cryptosystem, even the slightest alteration in the original image must produce an entirely different cipher image. This feature is a straightforward result of the diffusion scheme implemented using the chaotic iterates of the logistic map. The security against this attack, restricts an adversary from picking and choosing a set of plain/original images and drawing out inferences from the ciphers obtained by encrypting each one of them. The metric that we have used for this test is the Number of Pixel Change Rate (NPCR) metric to claim the security of the image cryptosystem. The NPCR test [40] measures the number of different pixels in the original and the modified cipher.

Mathematically, it is given by

\[ NPCR = \left( \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} D_{ij}}{MN} \right) \times 100\% \]

where \( C \) and \( C' \) are the original and modified ciphers respectively, \( M \) and \( N \) gives the dimensions of the original and cipher images. For the grayscale images ideal \( NPCR \) values are \( \geq 99.6075 \) [40]. We subject our cryptosystem to this test with the following settings: Encrypt each image twice, first with the original image, and then with the slightly changed image. The modification in the original image is the change in the pixel intensities by ‘1’. For ex. 255 is changed to 254, and this tiny change would not affect the appearance and texture of the image. However, the cryptosystem should adapt to this change and reflect this in the final cipher. Table 4 gives the numerical
scores of this metric on images used in Fig. 11. As can be seen, the scores are high and thus back our claim of a secure cryptosystem. Fig. 13 gives the edges of the image obtained by the difference of the original and modified cipher using the Canny-edge detector. As can be seen, there is no coherent edge structure that can leak any useful information to the adversary based on the chosen plain-text attack.

(iv) Key-Sensitivity Analysis: Key-sensitivity is an attribute that a robust encryption algorithm should possess to produce an entirely different output due to a slight alteration in the encrypting keys. We have subjected our algorithm to this test by changing the set of encryption keys at the time of decryption. The encryption-decryption keys used in this test are listed in Table 5. One of the important keys in our algorithm is the wavelet involved in the VMCI generation. We change the 'Haar' wavelet to 'Lazy' wavelet in this sensitivity analysis. The 'Lazy' wavelet involves the sub-sampling of even and odd indexed samples and works precisely as polyphase representation used in designing filter banks [41].

Our algorithm boasts of the perfect reconstruction of both the original and the reference image. In Fig. 14 for every parameter that is changed, the corresponding decrypted and reconstructed versions of the original and the reference image are shown, respectively. It is clearly visible that the even the reference image is not perfectly reconstructed when the key is changed (second row of the figure). Moreover, wrongly decrypted image give a noise-like appearance. Therefore, we can claimed that the proposed algorithm is highly sensitive to key-changes.
Fig. 12. Histogram analysis: First Row — Original Images, Second Row — The generated VMCI, Third Row — Original images histograms, Fourth Row — VMCI histograms.

Table 5
Encryption-Decryption keys.

| Parameter | Encryption key | Decryption key |
|-----------|----------------|----------------|
| $x_0$     | 0.22           | 0.2199999999999999 |
| $\mu$     | 4              | 3.9999999999999 |
| $s$       | 15             | 16             |
| $(\Delta x, \Delta y)$ | (1, 1) | (2, 1) |
| $w_{rp}$  | 1              | 2              |
| DF        | $10^6$         | $10^6$         |
| $w_{av}$  | $|haar'|$      | $|lazy'|$      |
| Step 49 of Algorithm 1 | $F = (E(2, :) + E(3, :) + 10 * E(4, :) + DF)$ | $F = (E(2, :) + 10 * E(3, :) + E(4, :) + DF)$ |

(v) Occlusion Attack: The transmission of a digital image over the internet may involve a data loss either through the network or can be a deliberate attempt by an adversary to destroy the integrity of an image. The encryption algorithm should be designed in such a way that even if there is some occlusion in the transmitted digital image, the receiver at the other end may still be able to extract visually meaningful information upon decryption of the image. The amount or level of information extracted depends upon the level of occlusion, so we have subjected the VMCI to different levels of data losses and recovered the original image with a sufficient amount of visual quality, as can be seen in Fig. 15.

(vi) Noise Attacks: The transmission of images may involve contamination in terms of the addition of different types of noises. A noise can be termed as any unwanted signal that has high variations with the true signal. The noises that arise in digital images may be due to some technical reasons like data compression, aliasing, etc. The other source is due to human intervention to intentionally disrupt the images by adding unwanted frequencies in the VMCI so that even the authorized party cannot get information about the original image. A reliable and robust algorithm should be flexible enough to resist noise attacks to facilitate near-perfect or visually meaningful decryption. We have subjected our cryptosystem to ‘Salt and Pepper’ noises with varying densities: 0.2, 0.1, 0.05, and 0.01. The corresponding decrypted images after applying the median filtering are presented in Fig. 16. Fig. 16 also contains row 4 and 5 that shows the reconstructed images i.e., original and the reference image, recovered in algorithm [29]. As can be seen, the results are better, especially when it comes to the reconstruction quality of the reference image. Numerical results based on this test are summarized in Table 6. As can be inferred from both Fig. 16 and Table 6, both the original image as well as the reference image can be recovered with a sufficient amount of visual quality. Moreover, the algorithm fairs pretty well and seems to outperforms the algorithm presented in [29].
Fig. 13. Edge detection results: First Row (left to right) shows the edges of the difference of ciphers corresponding to Lena, Elaine, Cameraman images. Second Row (left to right) shows the edges of the difference of ciphers corresponding to Peppers, Couple and Scenery images.

Fig. 14. Key-Sensitivity Analysis. First row: (a) Wrongly decrypted image when ‘wavelet’ is changed, (b) Wrongly decrypted image when $x_{ij}$ is changed, (c) Wrongly decrypted image when $\mu$ is changed, (d) Wrongly decrypted image when $s$ is changed, (e) Wrongly decrypted image when $\alpha_{ij}$ is changed. Second Row: Corresponding decrypted reference image. Third Row (f) Wrongly decrypted image when $\alpha_{ij}$ is changed, (g) Wrongly decrypted image when $DP$ is changed, (h) Wrongly decrypted image when Step 49 of Algorithm 1 is changed, (i) Correctly decrypted image. Fourth Row: Corresponding reconstructed reference image.
5.2. Steganalysis

Steganalysis is the technology to deal with steganography. Since digital steganography conceals information into the carrier image without any obvious visible distortions to it, therefore it becomes imperative to develop tools and algorithms to find out those subtle manipulations and hence the secret. The steganalysis methods can be broadly classified into six major attacks: (a) Stego-only attack, (b) Known cover attack, (c) Known message attack, (d) Known stego attack, (e) Chosen stego attack, (f) Chosen message attack. We can be safe in assuming that the adversary has with it only the VMCI/Stego-image without any other discernible information and tries to draw out conclusions about the secret. The steganalysis methods can be broadly classified into six major attacks: (a) Stego-only attack, (b) Known cover attack, (c) Known message attack, (d) Known stego attack, (e) Chosen stego attack, (f) Chosen message attack. We can be safe in assuming that the adversary has with it only the VMCI/Stego-image without any other discernible information and tries to draw out conclusions about the secret. We consider the following two famous steganalysis tests on our stego-image [42].

(I) Statistical Test: The Chi-square test for image steganalysis [42] is based on comparing Pair-of-Values (POV) observed frequencies with their expected frequencies to calculate the $p$-value, which is the probability of having something embedded in the image. The basic principle of this test works with the assumption that when an image is embedded with secret information, the LSB values change in such a way that the number of these POV pairs become nearly equal while they differ when nothing is embedded. We performed this test on five images giving quantitative and qualitative results on each. The first four images are the VMCI generated by our algorithm, and the fifth is the stego-image generated by LSB embedding. Table 7 gives the quantitative results of this test, while Fig. 17 presented the qualitative results. The proposed VMCI has a very low probability of getting detected for a secret message, whereas the probability is 0.999999432510001 in the conventional LSB embedded stego-image. As a comparative analysis, we embed all the encrypted images from Fig. 1 into an 8-bit ‘Lena’ image and perform the chi-square test on the resultant stego-image. The embedding is done by flipping the LSB of the cover image to match the LSB of the secret image. We assume that the number of bits in the secret image is almost equal to the number of pixels in the cover image to satisfy the necessary condition that each pixel contains at most one embedded bit. To assure this constraint, we embed the LSB of the secret image of size 512 × 512 into the LSB of the cover image. Tables 8 and 9 shows the results of the stego-image generated by embedding the AES encrypted image. One can observe that the probability of detecting the embedded data is higher in this case as compared to the VMCI.

Table 6
Noise Attack Evaluation based on PNSR and SSIM.

| Noise     | Original Image | Reference Image |
|-----------|----------------|-----------------|
| PSNR      | SSIM           | PSNR            |
| Intensity |                |                 |
| 0.2       | 17.0654        | 0.3342          |
| 0.1       | 20.9480        | 0.5833          |
| 0.05      | 26.1711        | 0.8060          |
| 0.01      | 31.5997        | 0.9323          |

(II) RS test: To analyze the stego-image for the presence of a secret, Fridrich et al. in [43] developed a method that focused on dividing the image into three groups known as Regular (R), Singular (S) and Unused (U) group. The method involves choosing a specific discriminant function for the variation of the intensities in the image as $f(p_1, p_2, \ldots, p_N) = \sum_{i=1}^{N} |p_{i+1} - p_i|$. The classification for a group of pixels $G$ is done as follows:

$$G \in \begin{cases} 
R, & \text{if } f(F_M(G)) > f(G) \\
S, & \text{if } f(F_M(G)) < f(G) \\
U, & \text{otherwise} 
\end{cases}$$

where $G$ is the $2 \times 2$ pixel group, $M$ is a specific vector $[0 \ 1 \ 1 \ 0]$ and the function $F$ is defined as follows:

$$F_1(p) = \begin{cases} 
p + 1, & \text{if } p = 2k \text{ for some } k \\
p - 1, & \text{if } p = 2k + 1 \text{ for some } k 
\end{cases}$$

and $F_2(p) = p$. For example $f(F_M([163, 163, 162, 162])) = f([163, 162, 163, 162]) = 3$. Parsing through the entire image in blocks of two, we calculate the total number of $R_M, S_M, R_S,$ and $S_S$ denote the regular and singular groups indexed by the vector $M$. Similarly, we calculate $R_M, S_M$ by replacing the vector $M$ by $-M$ and the function $F$ as $F'_1(p) = F_1(x+1) - 1$. The analysis of these values gives the percentage of pixels likely to be embedded in the stego-image. The inferences made by these analyses is that the difference between the $R_M$ and $S_M$ would be meager. Whereas, the difference between $R_M$ and $S_M$ would increase with the increase in the length of data embedded. We subjected our VMCI to this test to find out the percentage of embedded data that the RS analysis gives. To give a comparative analysis, we subject two more images, (i) Lena-Stego image with LSB embedding to bring out the clear advantages of the proposed scheme over LSB embedding, and (ii) Original image to bring out the difference in the values between the LSB-embedded, VMCI and the original image. The results in Table 10...
show that VMCI can withstand any steganalysis based on the RS analysis. To emphasize our claim that the AES encrypted stego-image is susceptible to the RS-analysis, we give the results on the eight images used in Fig. 1 in Table 11. The analysis of the results vividly depicts that this method of AES combined with steganography is not at par with our proposed method of VMCI generation.

5.3. Comparative analysis with existing work

A comparison of the existing schemes of [27,29] with the method presented in the proposed work is carried out concerning their embedding algorithms and give a numerical justification of why our proposed method surpasses them. One of the main contributions of this work lies in the perfect reconstruction of the reference image, which is a crucial drawback in the schemes presented in [27,29]. In this section, we give the reconstruction quality of the reference image in terms of the PSNR value and two other important parameters that capture the details in an image. We give a brief description of the two schemes to highlight the areas to which we have proposed a solution.

Scheme proposed in [29] : Although the embedding algorithm is a new one, some substantial improvements can be made. The first step of the embedding algorithm is the rounding-off error that
The Chi-square results indicating probability of embedded data in the VMCI.

| Pixels | Probability of embedding |
|--------|------------------------|
| Scanned | Lena | Barbara | Goldhill | Manik Woman | Lena-LSB |
| 0%     | 0   | 0        | 0        | 0            | 0        |
| 5%     | 0.00377295644734 | 0.75737096130764 | 0.01421823023988 | 0.23298639038598 | 0.9971694504467 |
| 10%    | 0.000192362403081 | 0.00735846719764 | 0.00603187499134 | 0.99251742959142 | 0.9953731107514 |
| 15%    | 0.15968749519664 | 0.00387878491759 | 0.01041440565397 | 0.9999739527916 | 0.9999739527916 |
| 20%    | 0.197564482628050 | 0.00873309949712 | 0.01078322011916 | 0.13351394612935 | 0.9999450121784 |
| 25%    | 0.115640475342815 | 0.00664498160767 | 0.13168152720694 | 0.23589061304937 | 0.999982140175 |
| 30%    | 0.041196221426751 | 0.01542762257516 | 0.02276548873753 | 0.16351352176176 | 0.9999394015918 |
| 35%    | 0.032654487790185 | 0.00695313002215 | 0.00706758926068 | 0.4169674131165 | 0.9999655530781 |
| 40%    | 0.004152068385484 | 0.00250351815230 | 0.00176646785822 | 0.95962971039253 | 0.9999085065850 |
| 45%    | 0.005925116655050 | 0.00469172652055 | 0.00309632134088 | 0.80306180117682 | 0.9999213607555 |
| 50%    | 0.001419412094805 | 0.00464769562883 | 0.00589603259545 | 0.54267934651235 | 0.9999409536932 |
| 55%    | 0.000044658084650 | 0.00862416994832 | 0.007102999959 | 0.00202267525654 | 0.9999613569168 |
| 60%    | 0.000462583958235 | 0.05945297353620 | 0.00502145893354 | 9.06134794234586-09 | 0.9999917131239 |
| 65%    | 0.000452619312798 | 0.00880019906988 | 0.0099499413538 | 1.07580611086178-12 | 0.9999213607555 |
| 70%    | 0.000181440907703 | 0.00241557357265 | 0.00064164835722 | 0.00006146485327 | 0.00269113134097 | 0.9999949895021 |
| 75%    | 4.38321397374321-05 | 6.2487571332292-05 | 0.00121138314097 | 0.9999949895021 |
| 80%    | 7.81896744313790-06 | 9.82755724527542-06 | 1.07830250701842-05 | 0.99998194718405 |
| 85%    | 1.30912600657214-06 | 8.73761807682472-06 | 4.66641379090104-05 | 0.99998194718405 |
| 90%    | 1.48032490698356-05 | 0.0009175432875812 | 7.6299153222492-06 | 0.99998194718405 |
| 95%    | 2.07152079512163-06 | 0.0006890143514086 | 5.63310389667393-07 | 0.99999527904524 |
| 100%   | 2.20578453512919-06 | 0.0002168694529278 | 2.77643488098119-07 | 0.99999614844258 |
and reference image used in the other two algorithms are of size $512 \times 512$ and $1024 \times 1024$ respectively, and the pre-encryption algorithm used for them is the same as given in [45].

As can be seen, both the versions of our proposed schemes (i) guarantee perfect reconstruction of the reference image, (ii) the

| Table 10 | RS analysis of the VMCI |
| --- | --- |
| Images | $R_M$ | $S_M$ | $R_M - S_M$ | $S_M$ | % of Embedded text |
| Barbara VMCI | 18132 | 13454 | 18155 | 13434 | 0.0045 |
| Lena VMCI | 19418 | 13353 | 19420 | 13248 | 0.0093 |
| Goldhill VMCI | 17413 | 13963 | 17447 | 14028 | 0.0040 |
| Cameraman VMCI | 22177 | 10793 | 22399 | 10840 | 0.0079 |
| Lena LSB stego image | 19736 | 19641 | 30118 | 11531 | 1.0211 |
| Original Lena Image | 24406 | 17093 | 24400 | 17135 | 0.0033 |

| Table 11 | RS analysis of the AES embedded stego-image used in Fig. 1 |
| --- | --- |
| Images | $R_M$ | $S_M$ | $R_M - S_M$ | $S_M$ | % of Embedded text |
| Sierpinsky | 19622 | 19715 | 30027 | 0.0045 |
| TUX | 19759 | 19628 | 30058 | 0.0045 |
| Farhan | 19803 | 19278 | 29841 | 0.0045 |
| Hilbert | 19605 | 19263 | 29926 | 0.0045 |
| Goldhill | 19836 | 19545 | 29954 | 0.0045 |
| Lena | 19697 | 19413 | 29838 | 0.0045 |
| Barbara | 19828 | 19399 | 29838 | 0.0045 |
| Man & Woman | 20100 | 19360 | 29699 | 0.0045 |

5.4. Applications of the proposed scheme for integrity of digital data

We explain the potential applications of our VMCI generating algorithm in the context of Online Social Interaction Network (OSIN). Due to a sudden boom in the IT sector and digital globalization, there is a huge flow of cross-border information exchange. The results of this digitization can be seen in the form of outbursts of OSIN’s like Facebook, Instagram, Twitter, Whatsapp, Amazon. With no such Data Protection Laws (DPL), even in the most developed countries, there is a severe need to safeguards our digital content while interacting on these OSIN’s. The network of transmission over the internet is far more complicated than just a sender–receiver model. The longer the networks, more are the chances that the data being transmitted gets corrupted by noise in the channel, or by an intruder. The data on these social networking sites ranges from texts to images relating to personal information, events, etc. Imagine transmitting a digital image on the OSIN to a friend which is intercepted by a saboteur. Cryptography is an option, but since it converts the image into noise, it calls for suspicion on the transmitted text, which may eventually result in integrity loss and corrupt file transmission. The proposed scheme incorporates an encryption algorithm and also generates a visually meaningful cipher that can be uploaded on a cloud database.

An important application of the proposed VMCI would be in the areas of medical imaging. Critical information about a COVID-19 patient’s nasopharyngeal test images can be transmitted by embedding

5.4. Applications of the proposed scheme for integrity of digital data

We explain the potential applications of our VMCI generating algorithm in the context of Online Social Interaction Network (OSIN). Due to a sudden boom in the IT sector and digital globalization, there is a huge flow of cross-border information exchange. The results of this digitization can be seen in the form of outbursts of OSIN’s like Facebook, Instagram, Twitter, Whatsapp, Amazon. With no such Data Protection Laws (DPL), even in the most developed countries, there is a severe need to safeguards our digital content while interacting on these OSIN’s. The network of transmission over the internet is far more complicated than just a sender–receiver model. The longer the networks, more are the chances that the data being transmitted gets corrupted by noise in the channel, or by an intruder. The data on these social networking sites ranges from texts to images relating to personal information, events, etc. Imagine transmitting a digital image on the OSIN to a friend which is intercepted by a saboteur. Cryptography is an option, but since it converts the image into noise, it calls for suspicion on the transmitted text, which may eventually result in integrity loss and corrupt file transmission. The proposed scheme incorporates an encryption algorithm and also generates a visually meaningful cipher that can be uploaded on a cloud database.

An important application of the proposed VMCI would be in the areas of medical imaging. Critical information about a COVID-19 patient’s nasopharyngeal test images can be transmitted by embedding
the converted VMCI in the Electro Cardiogram Signals (ECG) [46] so that only designated health care officials can access it.

The other relationship, though it is not directly related, is with the concept of network steganography and Stream Control Transmission Protocol (SCTP) [47] which relies on manipulating the internet protocol (IP) which is imperative in any communication. Among the few network steganographic methods, Protocol Steganography, which manipulates the packet header files, is an area where our proposed scheme can be utilized. Since exclusively adhering to these Protocol steganographic methods involve the risk of detection, potential loss of information, our proposed method can work in tandem to embed the stego-image in the unused, partial, or optional fields and transmit the data without violating any constraints of the excess size of the data transmitted and without an actual sender-receiver synchronization.

6. Conclusion

This algorithm generated a visually meaningful cipher image in the wavelet domain and facilitated reversible data hiding. Experimental results validate our claim that the use of mathematical pairing function produces a VMCI of the same size as that of the original image without incurring any excess storage, transmission, and computational costs. This fact was missing in the related existing algorithms. Further, the generation of partial cipher using an alternative chaos-based GBDA scheme was found quite comparable to the traditional algorithm like AES-ECB. Obtaining a perfect reconstruction of the reference images scheme was found quite comparable to the traditional algorithm like AES-ECB. Obtaining a perfect reconstruction of the reference images employing a pseudo image technique in the Fourier domain, Opt. Commun. 321 (2014) 12–30.

Thus, $h^{-1}(n_2) \leq z < h^{-1}(n_2 + 1)$

Thus, $h^{-1}(n_2) \leq z < h^{-1}(n_2 + 1)$

$n_2 \leq h(z) < n_2 + 1$

$n_2 \leq \sqrt{z} < n_2 + 1$

Now, since $n_2$ and $n_2 + 1$ are two consecutive numbers, thus we obtain that

$n_2 = \lfloor \sqrt{z} \rfloor$ \hspace{1cm} (23)

$n_1 = z - (n_2)^2$ \hspace{1cm} (24)

Case 2: When $n_1 = \max(n_1, n_2)$. Let $z = n_1^2 + n_1 + 1$, define $t = n_1^2 + n_1$ so that $z = t + n_2$. Since

$t = n_1^2 + n_1$ \implies n_1 = -1 \pm \sqrt{(1 + 4t)} 2$

but since $n_1 \geq 0$ \implies $n_1 = -1 + \sqrt{(1 + 4t)} 2$

Thus, $n_1 = h(t) = -1 + \sqrt{(1 + 4t)} 2$

proceeding as above we see that $h$ is invertible

Now, $1 \leq t + n_2 < t + 2n_2 + 1 \implies n_1^2 + n_1 \leq z < (n_1 + 1)^2 + (n_1 + 1)$

Thus, $h^{-1}(n_1) \leq z < h^{-1}(n_1 + 1)$

$n_1 \leq h(z) < n_1 + 1$

$n_1 \leq -1 + \sqrt{(1 + 4z)} 2 < n_1 + 1$

$n_1 = \left[ -1 + \sqrt{(1 + 4z)} 2 \right] \hspace{1cm} (25)$

$n_2 = z - n_1^2 - n_1$ \hspace{1cm} (26)

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgments

One of the authors, Farhan Musanna is grateful to the Ministry of Human Resource Development, India and the Indian Institute of Technology Roorkee for being the funding agency of this work, with grant number MHR-01-23-200-428. All authors approved the version of the manuscript to be published.

Appendix

We give a theoretical justification of the invertibility of the function defined in Section 5.2.

Case 1: When $n_1 \neq \max(n_1, n_2)$. Let $z = \sigma(n_1, n_2) = n_1 + n_2^2$. Define $t = n_2^2$ so that $z = t + n_2$. Since

$t = n_2^2$ \implies n_2 = \pm \sqrt{t}$, but since $n_2 > 0$ \implies $n_2 = \sqrt{t}$

Thus, $n_2 = h(t) = \sqrt{t}$, now since $h'(t) = \frac{1}{2\sqrt{t}} > 0$, \forall t > 0, hence one–one and clearly onto \implies $h$ is invertible with $h^{-1}(n_2) = t^2$

References

[1] Xuejai Lai, James L. Massey, Sean Murphy, Markov Ciphers and differential cryptanalysis, in: Workshop on the Theory and Application of of Cryptographic Techniques, Springer, 1991, pp. 17–38.

[2] Joan Daemen, Vincent Rijmen, The Design of Rijndael: AES-The Advanced Encryption Standard, Springer Science & Business Media, 2013.

[3] Ronald L. Rivest, Adi Shamir, Leonard Adleman, A method for obtaining digital signatures and public-key cryptosystems, Commun. ACM 21 (2) (1978) 120–126.

[4] Yaobin Mao, Guanrong Chen, Shiguo Lian, A novel fast image encryption scheme based on 3D chaotic baker maps, Internat. J. Bifur. Chaos 14 (10) (2004) 3613–3624.

[5] Z. Hua, Y. Zhou, C.M. Pun, C.L.P. Chen, 2D Sine logistic modulation map for image encryption, Inform. Sci. 297 (2015) 80–94.

[6] J. Fridrich, Symmetric ciphers based on two-dimensional chaotic maps, Int. J. Bifurcation Chaos 8 (6) (1998) 1259–1284.

[7] M.S. Baptista, Cryptography with chaos, Phys. Lett. A 240 (1–2) (1998) 50–54.

[8] Vinod Patidar, N.K. Pareek, G. Purohit, K.K. Sud, A robust and secure chaotic standard map based pseudorandom permutation-substitution scheme for image encryption, Opt. Commun. 284 (19) (2011) 4331–4339.

[9] Chong Fu, Bin-bin Lin, Yu-sheng Miao, Xiao Liu, Jun-je Chen, A novel chaotic-based bit-level permutation scheme for digital image encryption, Opt. Commun. 284 (23) (2011) 5415–5423.

[10] Guodong Ye, Kwok-Wo Wong, An efficient chaotic image encryption algorithm based on a generalized Arnold map, Nonlinear Dyn. 69 (4) (2012) 2079–2087.

[11] Raul Enayatifar, Abdul Hanan Abdullah, Ismail Fauzi Insin, Chaos-based image encryption using a hybrid genetic algorithm and a DNA sequence, Opt. Lasers Eng. 56 (2014) 83–93.

[12] Changliang Guo, Shi Liu, John T. Sheridan, Optical double image encryption employing a pseudo image technique in the Fourier domain, Opt. Commun. 321 (2014) 61–72.

[13] Juliano B. Lima, L.F.G. Novaes, Image encryption based on the fractional Fourier transform over finite fields, Signal Process. 94 (2014) 521–530.
[14] Juliano B. Lima, Edmar S. da Silva, Ricardo M. Campello de Souza, A finite field cosine transform-based image processing scheme for color image encryption, in: Signal and Information Processing (GlobalSIP), 2015 IEEE Global Conference on, IEEE, 2015, pp. 1071–1075.
[15] Musheng Chen, Zhifang Zhang, Zhishan Cai, Yuzhuo Pan, A novel image encryption method based on fractional Fourier transform and odd-even quantification, in: Eighth International Conference on Digital Image Processing (ICDIP) 2016, International Society for Optics and Photononics, Vol. 10033, 2016, pp. 1003332.
[16] Karl Martin, Rastislav Lukac, Konstantinos N. Plataniotis, Efficient encryption of wavelet-based coded color images, Pattern Recognit. 38 (7) (2005) 1111–1115.
[17] Ali Bouamandria, The 3D modular chaotic map to digital color image encryption, Future Gener. Comput. Syst. 99 (2019) 489–499.
[18] Ercan Solak, Cahit Çokal, Olcay Taner Yildiz, Türker Biyikoglu, Cryptanalysis of Fridrich’s chaotic image encryption, Int. J. Bifurcation Chaos 20 (05) (2010) 1405–1413.
[19] S.J. Li, Xuan Zheng, Cryptanalysis of a chaotic image encryption method, in: Proceedings of 2002 IEEE International Symposium on Circuits and Systems, Vol. 2, 2002, pp. 708–711.
[20] Hong Liu, Yanbing Liu, Cryptanalyzing an image encryption scheme based on hybrid chaotic system and cyclic elliptic curve, Opt. Laser Technol. 56 (2014) 15–19.
[21] Neil F. Johnson, Sushil Jajodia, Exploring steganography: Seeing the unseen, Computer 31 (2) (1998).
[22] J.C. Judge, Steganography: Past, present, future. SANS institute publication, 2001.
[23] Niels Provos, Peter Honeyman, Hide and seek: An introduction to steganography, IEEE secu. priv. 99 (3) (2003) 32–44.
[24] Ruizhen Liu, Tieniu Tan, et al., An SVD-based watermarking scheme for protecting rightful ownership, IEEE trans. multimed. 4 (1) (2002) 121–128.
[25] Rakesh Dugad, Krishna Ratakonda, Narendra Ahuja, A new wavelet-based scheme for watermarking images, in: Image Processing, 1998. ICIP. 98. Proceedings. 1998 International Conference on, IEEE, Vol. 2, 1998, pp. 419–423.
[26] Ashima Anand, Amit Kumar Singh, An improved DWT-SVD domain watermarking for medical information security, Comput. Commun. 152 (2020) 72–80.
[27] Long Bao, Yicong Zhou, Image encryption: Generating visually meaningful encrypted images, Inform. Sci. 324 (2015) 197–207.
[28] Wim Sweldens, The lifting scheme: A construction of second generation wavelets, SIAM J. Math. Anal. 29 (2) (1998) 511–546.
[29] Ali Kanso, Mohammad Ghebleh, An algorithm for encryption of secret images into meaningful images, Opt. Lasers Eng. 90 (2017) 196–208.
[30] Wenying Wen, Yushu Zhang, Yuming Fang, Zhijun Fang, Image salient regions encryption for generating visually meaningful ciphertext image, Neural Comput. Appl. 29 (3) (2018) 653–663.
[31] Xiuli Chai, Zihua Gan, Yiran Chen, Yushu Zhang, A visually secure image encryption scheme based on compressive sensing, Signal Process. 134 (2017) 35–51.
[32] Farhan Musanna, Sanjeev Kumar, A novel fractional order chaos-based image encryption using Fisher Yates algorithm and 3-D cat map, Multimedia Tools Appl. (2018) 1–29.
[33] Georg Cantor, Ueber eine Eigenschaft des Inbegriffs aller reellen algebraischen Zahlen, Journal für die reine und angewandte Mathematik 77 (1874) 258–262.
[34] Michael François, Thomas Grorges, Dominique Barchiesi, Robert Erra, Image encryption algorithm based on a chaotic iterative process, Appl. Math. 3 (12) (2012) 1910.
[35] William Kahan, IEEE Standard 754 for binary floating-point arithmetic, in: Lecture Notes on the Status of IEEE, 754, (94720–1776) 1996, p. 11.
[36] Yue Wu, Joseph P. Noonan, Sos Agaian, et al., NPCR and UACI randomness tests for image encryption, Cyber journal: multidisciplinary journals in science and technology, J. Selected Areas Telecommun. (JSAT) 1 (2) (2011) 31–38.
[37] Andreas Westfeld, Andreas Pfitzmann, Attacks on steganographic systems, in: System Science and Engineering (ICSSE), 2012 International Conference on, IEEE, 2012, pp. 69–73.
[38] Ayman Ibaida, Ibrahim Khalil, Wavelet-based ECG steganography for protecting patient confidential information in point-of-care systems, IEEE Trans. Biomed. Eng. 60 (12) (2013) 3322–3330.
[39] Jessica Fridrich, Miroslav Goljan, Rui Du, Detecting LSB steganography in color, and gray-scale images, IEEE multimedia 8 (4) (2001) 22–28.
[40] Gaurav Bhatnagar, Ashirbani Saha, Q.M. Jonathan Wu, Pradeep K. Atrey, Analysis and extension of multiresolution singular value decomposition, Inform. Sci. 277 (2014) 247–262.
[41] Long Bao, Yicong Zhou, C.L. Philip Chen, Hongli Liu, A new chaotic system for image encryption, in: System Science and Engineering (ICSSIE), 2012 International Conference on, IEEE, 2012, pp. 69–73.
[42] Ayman Ibaida, Ibrahim Khalil, Wavelet-based ECG steganography for protecting patient confidential information in point-of-care systems, IEEE Trans. Biomed. Eng. 60 (12) (2013) 3322–3330.
[43] Wojciech Frączek, Wojciech Mazurczyk, Krzysztof Szczypiorski, Hiding information in a stream control transmission protocol, Comput. Commun. 35 (2) (2012) 159–169.