CHAOS
Theory and Applications
in Applied Sciences and Engineering

e-ISSN: 2687-4539
RESEARCH ARTICLE
Vol.6 / No.4 / 2024 / pp.284-293
https://doi.org/10.51537/chaos.1517688

A New Data Coding Algorithm for Secure Communication of Image

Hidayet Oğraş *
*Department of Electronics Communication Technology, Batman University, Central Campus, Batman, Turkiye.

ABSTRACT This paper proposes a new entropy-sensitive based data coding algorithm for the secure communication of image information between transceiver systems. The proposed algorithm utilizes chaos theory and the image information content of the reference image to create sensitivity on the decoding side for a high level of secrecy. It successfully recovers secret images at the receiver's side using secret code series derived from both the secret and reference images, instead of direct transmission of secret image. The image information can be retrieved only through the same reference image, the same system parameters and identical code series using the proper decoding technique at the receiver. Quantitative results indicate that the average coding time for 128x128 images is approximately 0.27 seconds, while the extraction time averages 0.19 seconds, yielding impressive rates of 0.487 Mbps and 0.677 Mbps, respectively. Moreover, according to qualitative results, even a single-bit change in the reference image leads to a complete inability to decode the secret image, highlighting the robustness and security of the algorithm. Experimental results on various images show that the proposed algorithm is reliable, fast and effective in securing confidential image information.

KEYWORDS
Chaos theory
Logistic map
Data coding
Image entropy

INTRODUCTION

Today, with the rapid development of data communication techniques and network technologies, many multimedia data have been transmitted and shared over open networks such as internet. Most of this information transmitted or shared over public networks need to be protected privately. Cryptography is a common approach used for the protection of data security by making the original message to unintelligible form (Sharma *et al.* 2022; Gurunathan and Rajagopalan 2020). However, one of the greatest challenges in cryptography is the secure generation, distribution and storage of the keys used in encryption and decryption processes (Rana *et al.* 2023). Cryptography has no ability to protect against vulnerabilities and threats resulting from poor design of systems, protocols and algorithms. Strongly encrypted, authentic and digitally signed information can sometimes be difficult to access, even for an authorized user. Moreover, creating suitable secret keys that meet sufficient security conditions is not easy in terms of circuit complexity, resource and time costs (Rana *et al.* 2023; Abba *et al.* 2024).

Another approach that can be used in secure communication is steganography. In steganography the message is hidden into another medium in a way that cannot be noticed by eavesdropper during communication (Jaradat *et al.* 2021; Mishra and Bhanodiya 2015). One of the major disadvantages of steganography is that there is a large overhead for hiding a very small amount of information. It provides communication secrecy with limited data capacity. Digital multimedia data such as images, audio and video are used as the cover medium where confidential information can be hidden (Pradhan *et al.* 2018). Among them, image files are very ideal as cover media due to the having large amount of redundant space (Ogras 2019). Image data hiding techniques can be classified into two important types: Spatial domain and Transform domain. Spatial domain techniques deal with image pixels and all of them directly replace some bits in pixel values while hiding the data (Hussain and Hussain 2013). Most commonly used technique in spatial domain is Least-Significant-Bit (LSB) embedding in which the secret message inside an image by replacing least significant bit of cover image with the bits of message to be hidden (Abba *et al.* 2024). For instance, LSB technique embeds a secret message into the cover image with one bit, so the modified pixel value is increased or decreased by 1 according to the used algorithm.

However, even if there is a tinny change in pixel values, total difference can be easily detected statistically with known analysis methods without much effort and so the secret information can be easily revealed. Another popular spatial domain method used in image steganography is the pixel value differencing (PVD) technique in which secret data is embedded into a cover image depend-

Manuscript received: 17 July 2024,
Revised: 14 October 2024,
Accepted: 21 November 2024.

[1]hidayet.ogras@batman.edu.tr (**Corresponding author**).

ing on pixel neighbourhood differences. Some studies (Hosam and Ben Halima 2016; Swain 2018a,b) based on PVD methods are proposed in image steganography. However, both techniques are not strong against some steganographic attacks as PVD can be detected by pixel difference histogram analysis.

A good data hiding algorithms aims at two important purposes: Payload capacity and imperceptibility. Payload capacity means the maximum amount of secret information that can be hidden inside a cover medium (Jaradat *et al.* 2021). Imperceptibility refers the visual quality of a stego image. After hiding secret message into a cover medium such as an image, the visual quality will decrease compared to cover image that results a slight distortion on the stego image. This distortion should be at an unnoticeable level; otherwise the risk of confidential information being detected will increase (Huang and Wang 2020). If zero distortion occurs as a result of such data coding process, then this will be interpreted as maximum imperceptibility, hence no analysis methods can be applied to detect secret information.

Many existing studies in the field of secure image communication rely heavily on traditional cryptographic techniques or steganographic methods that often introduce additional complexities, such as encryption algorithms or compression schemes. These approaches can lead to increased processing times and power consumption, making them less suitable for real-time applications. In contrast, chaotic systems have gained substantial interest in secure communication due to their inherent unpredictability, sensitivity to initial conditions and complex dynamic behavior (Liu *et al.* 2022). In such systems, slight changes in initial parameters can lead to significantly different outcomes, a characteristic often described as the "butterfly effect" (Bonny and Al Nassan 2024). This sensitivity makes chaotic systems an ideal tool for secure data transmission, as even a minor deviation in transmission parameters can prevent unauthorized decoding attempts, which is a key strength of chaos-based communication systems in ensuring high security. These systems are also advantageous for real-time applications due to their computational efficiency, relying on simple mathematical operations compared to traditional security methods (Khan and Waseem 2024). This enables fast encoding and decoding, supporting secure, high-speed image transmission without requiring heavy computational resources. The proposed algorithm demonstrates this efficiency with low coding and decoding times, making it suitable for real-time image transmission.

The application of chaos theory in secure communication benefits from the complex, non-linear behavior of chaotic systems to create robust encoding mechanisms, particularly for applications involving image data (Zhang and Liu 2023). For instance, chaotic maps such as the Logistic, Henon, and Tent maps are widely used due to their simplicity and effectiveness in generating high-entropy sequences. These maps can efficiently scramble image data, ensuring that the encrypted data appears random to unauthorized observers. Because chaotic maps require only a few key parameters to function, they simplify key management in encryption systems, reducing the complexity associated with traditional secret key distribution. Unlike traditional encryption methods, chaotic systems do not depend on large keys for security; instead, they use a set of initial parameters and iterative processes to generate unpredictable sequences. This property makes chaos-based encryption systems less susceptible to brute-force attacks and highly efficient in terms of computational resources (Khan and Waseem 2024). For image data in particular, chaos theory provides distinct advantages, as the encoding process can utilize the image's inherent complexity (entropy), making the encoded message directly tied to the image data. In this study, the entropy of a reference image is used along with chaos-based coding, which provides a higher level of security, as both the reference image and chaotic map parameters are essential for decoding. On the other hand, chaotic systems also present certain challenges in secure communication applications. One concern is the accurate reproduction of chaotic sequences at the decoding end, which requires that both the sender and receiver systems be perfectly synchronized (Liu *et al.* 2022). Any mismatch in parameters could cause the chaotic system to diverge, leading to decoding errors. Additionally, chaotic systems can sometimes exhibit periodic behavior, especially if the system parameter is not properly selected, which may compromise their unpredictability.

This paper introduces a new entropy-sensitive data coding algorithm that addresses these shortcomings by using a unique combination of chaos theory and reference image entropy without altering the reference image. The main contributions of this research include:

- The proposed method enhances security by requiring the correct reference image and codes for recovering secret images, which makes unauthorized access difficult.
- The high sensitivity of the system to even very small changes prevents successful decoding.
- The algorithm minimizes processing times, making it suitable for real-time applications while maintaining the integrity of the reference image and ensuring robustness against detection methods.

The structure of the paper is as follows: first, the definition of chaos and the chaotic map utilized in the algorithm are introduced. Next, the proposed coding and decoding algorithms are described in detail. Then, experimental results, including the running time of the coding and decoding processes, along with evaluation comparisons of the proposed algorithm, are given. Finally, the paper concludes with a summary of findings.

CHAOS

Chaos means a state of total confusion with disorder (Ahmad and Shin 2021). All systems that contain chaos exhibit extreme sensitivity to initial condition and control parameters (Ozkaynak 2020; Effah-Poku *et al.* 2018). Although chaotic systems have deterministic structure, their long-term behavior cannot be predicted (Umoh and Wudil 2016). Hence, these properties can be used to transform obvious events into an irregular and unpredictable form in some engineering fields such as cryptography (Roy *et al.* 2021; Gafsi *et al.* 2020; Irsan and Antoro 2019) and secure communication (Sharafi *et al.* 2021; Kumar and Raghava 2019; Ismail *et al.* 2020; Oğraş and Türk 2013). In this paper, the well-known chaotic Logistic map (LM) is used to generate secret key for the proposed coding algorithm.

Logistic map

LM is a simple but frequently used system for generating pseudo-random sequences. It has a simple iterative structure having a dynamical equation as in Equation 1:

$$x_{n+1} = r \cdot x_n \cdot (1 - x_n) \tag{1}$$

where r refers to the control parameter of the map defined in (0,4]. If the r is between 3.57 and 4, then the map behaves chaotically, exhibiting chaotic properties such as non-periodicity and sensitivity to initial conditions. In this case, the output is distributed randomly within the range from 0 to 1. As a result, the LM can

produce unpredictable series like a key generator, which can be used in fields of subjects where the randomness is needed. For instance, the output generated from the LM with $x_0 = 0.1234$ and $r = 3.9999$ after 500 iterations is shown in Figure 1.

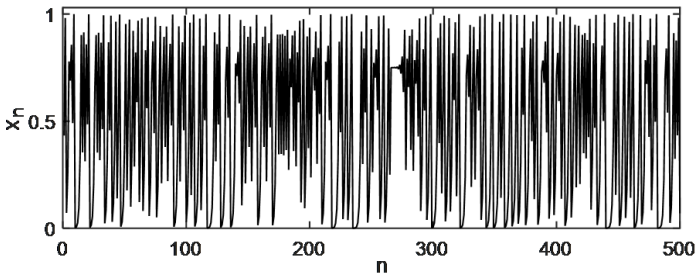

Figure 1 The output of chaotic LM

PROPOSED CODING ALGORITHM

Generating secret key and coding process

In this study, no data containing of the secret image information is sent directly to the receiver. Instead, unlike the cryptographic approach, secret image is mixed with a series of codes obtained from the reference image and the LM system through the proposed coding algorithm. The reference image in here is an arbitrary public image with which the secret image will be associated randomly. The secret image is a target image that will be decoded at the receiver side by being sensitively related to the reference image along with the secret code series. The coding process is carried out by associating the secret image with the secret codes using the reference image. The general block scheme of the proposed algorithm is shown in Figure 2

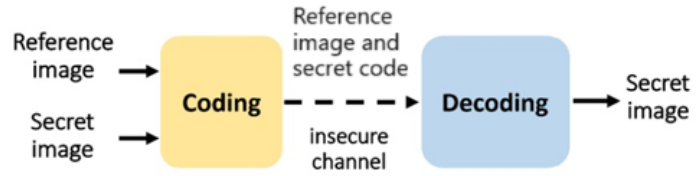

Figure 2 Block diagram of a proposed scheme

The algorithm uses a series of high-precision secret codes to decode the secret image. In addition to the parameters of the LM system, the reference image also affects the secret key over its entropy value as in Equation 2.

$$x_0' = (x_0 + \text{Entropy}(\text{reference\_img})) \mod 1 \tag{2}$$

x_0 and x_0' indicate the first and actual initial values of the LM, respectively. The equation introduces a perturbation to the initial value of the LM using the entropy of an image. Since entropy measures the amount of information or randomness in an image, this introduces variability based on the content of the image. The result is that x_0' will be slightly different from x_0 depending on the image characteristics. The use of the modulus operation ensures that x_0' remains within the range [0, 1]. This is crucial for LM, which typically require initial values in this interval. It is well known that chaotic systems are extremely sensitive to initial values. As a result of such an approach, the generated key series will be depended on the first initial value of the LM system and the entropy of the reference image with high sensitivity. This sensitivity on the coding side will also be reflected on the decoding side, which will play a major role in correctly decoding the encoded secret image information. Binary series are generated from the LM by using a simple mathematical transformation given in Equation 3.

$$b_n = \begin{cases} 1 & \text{if } x_n \geq 0.5 \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

A simple bit is obtained for each corresponding of x_n value. Notice that, the orders of the generated series are completely dependent on the initial condition and control parameter of the LM. Then, bitstreams of reference image are mixed with the generated binary series according to the bitstreams of the secret image. The whole processes for the generation of the secret codes are explained in detail as follows:

Step 1) A secret image to be coded has the same size with the reference image selected and then it is converted to serial bitstream format.

Step 2) According to the values in the serial bitstream, XOR (Exclusively-OR) or XNOR (not XOR) operations are performed up to the length of the bitstream for 1 and 0, respectively.

Step 3) Chaotic binary series are generated by the LM with a chosen control parameter, initial condition and the entropy value from the reference image.

Step 4) Reference image is reshaped to the length of size and then converted to serial bitstream like secret image.

Step 5) According to the bitstreams value of the secret image, if '1' occurs, XOR operation is performed with chaotic series and the bitstreams of the reference image. Otherwise, XNOR operation is performed.

Step 6) Step 5 is continued by considering all of the bitstream values for the secret image. Finally, the output represents modified chaotic series that will be used as secret codes in the proposed algorithm.

Schematic diagram for generating secret code is shown in the Figure 3. In the algorithm, size of the secret image is same as the reference image which results maximum payload capacity indeed. Block diagram of the coding process is shown in Figure 4.

Decoding process

On the receiver side, a secret image can be recovered with the same reference image, identical secret key and correct system parameters of the LM. If there is a slight change in any one of them, the secret image will not be decoded correctly, even 1-bit change occurs in the reference image. Block diagram for the decoding process is shown in Figure 5.

All system parameters of the LM must be unknown except the receiving side. Keeping these values secret plays an important key role in integrity for decoding the secret image. In addition, the secret codes must be transmitted through a channel to the receiving side knowing which reference image is associated with the secret image. As a result, the secret image can be correctly decoded through the inverse algorithm of the proposed scheme. Schematic diagram of the inverse algorithm for decoding process is shown in the Figure 6.

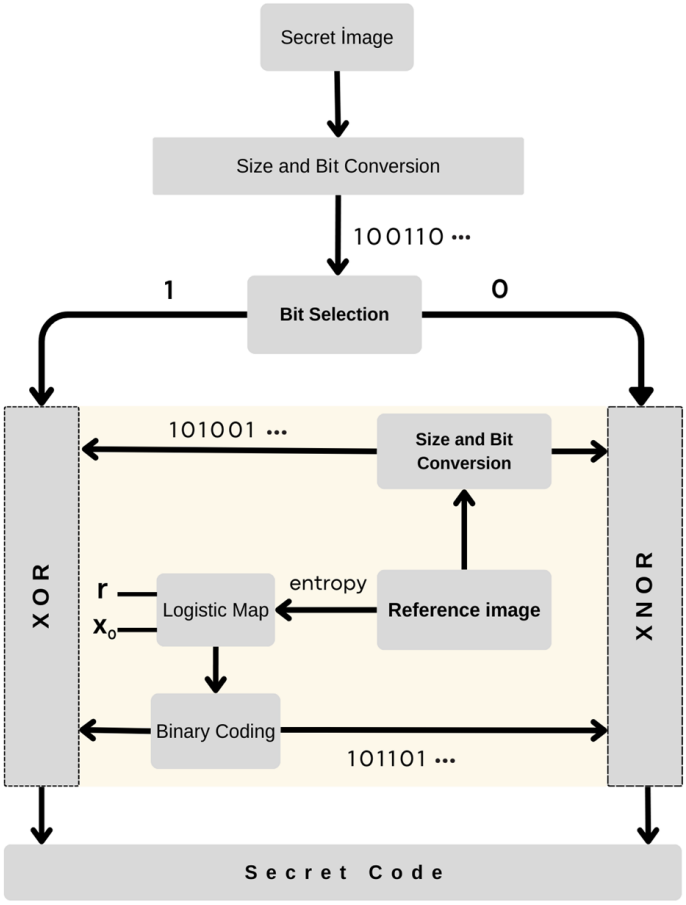

Figure 3 Schematic diagram for the generating secret code

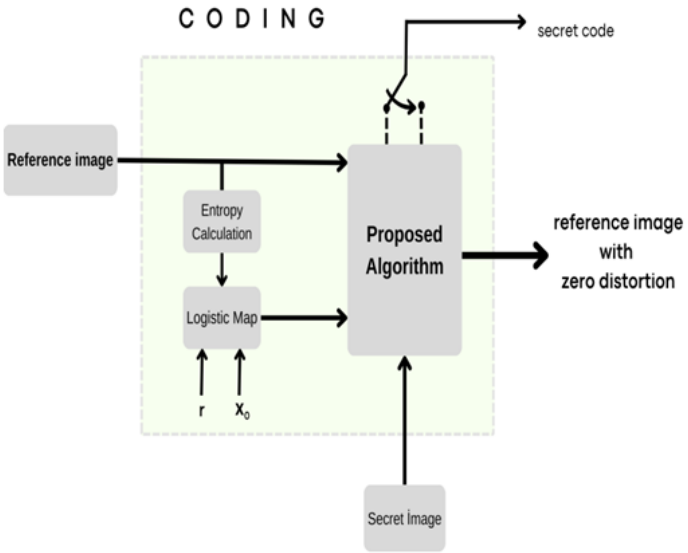

Figure 4 Block diagram of the coding process

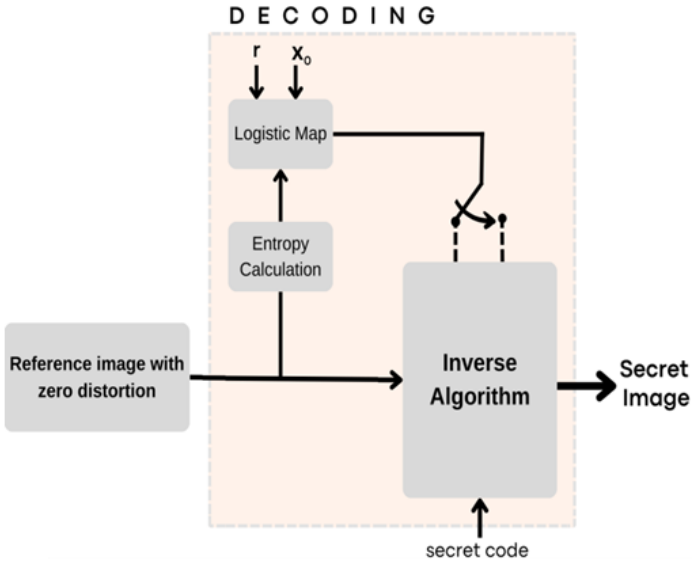

Figure 5 Block diagram of the decoding process

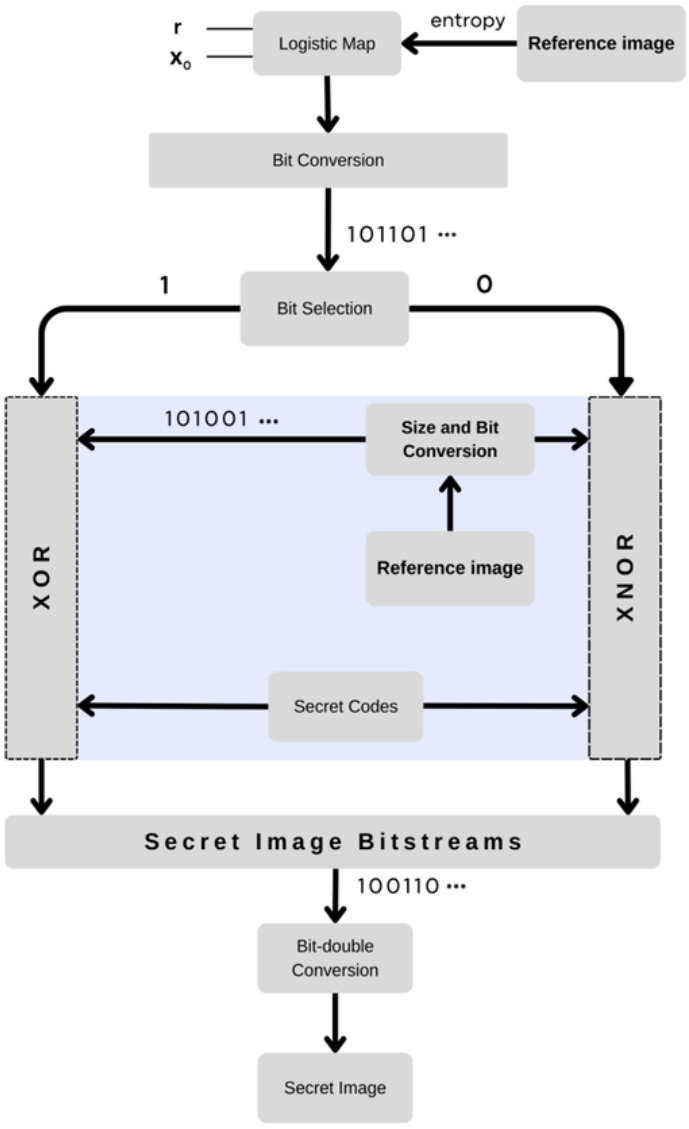

Figure 6 Schematic diagram of the inverse algorithm

EXPERIMENTAL RESULTS

Twelve grey test images are used to evaluate the performance of the proposed coding algorithm. The properties of the images are listed in Table 1, and some are shown in Figure 7.

To evaluate the performance of the proposed algorithm, sufficient number of simulation is performed with various test images by using Matlab software. The visual analysis results, as well as the running speed and decoding rate of the algorithm, are calculated using an Intel Core i3 2.13 GHz processor with 4 GB of RAM.

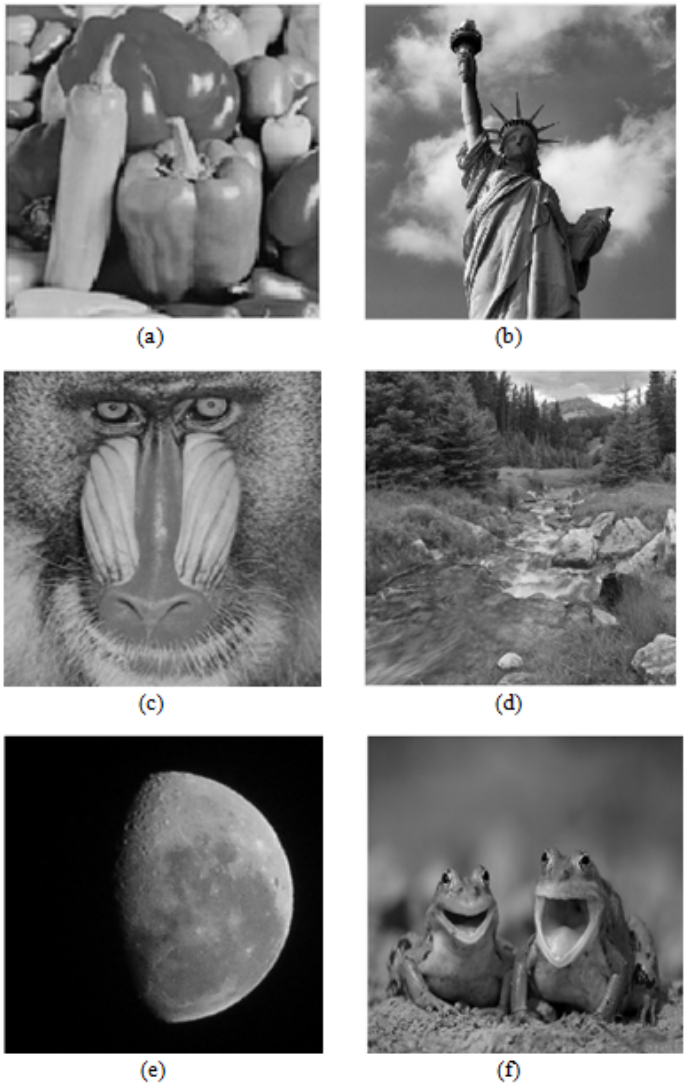

Figure 7 Some test images used in the proposed algorithm (a) Peppers (b) Liberty (c) Baboon (d) Landscape (e) Moon (f) Frog

Sensitivity analysis

In the proposed scheme, changing the entropy value of the reference image directly affects the initial value of the LM which results significant change in secret codes due to the chaos effect. Therefore, exactly the same reference image must be used in the receiver side to decode the secret image. The reference image is associated with the secret image at bit-level by using Bitwise operations and that correlation can only be appeared with the same secret key used at the receiver side. Furthermore, the secret image can only be detected under exactly the same reference image due to the entropy sensitivity of the algorithm. If a different reference image is used in the receiver side, then the secret image should not be decoded correctly. For this case, a test image of "Frog.jpg" is used as secret image and "Moon.jpg" image is used as reference image in the proposed algorithm. LM parameters are selected as $r = 4$ and $x_0 = 0.12345$. Then, on the receiver side, the secret image is tried to be decoded with the same LM parameters but using "Landscape.jpg" as reference image. The visual result is shown in Figure 8. According to the results, the secret image can only be decoded correctly for the associated correct reference image. If there is even a 1-bit pixel change in the correct reference image, the secret image should not still be decoded correctly. The secret "Frog" image is tried to be decoded by making only 1-bit change in the middle pixel of the correct reference image in Figure 8(c). In other words, the middle pixel value of 121 is changed to 122 in correct "Moon" image. The analysis result is shown in Figure 9.

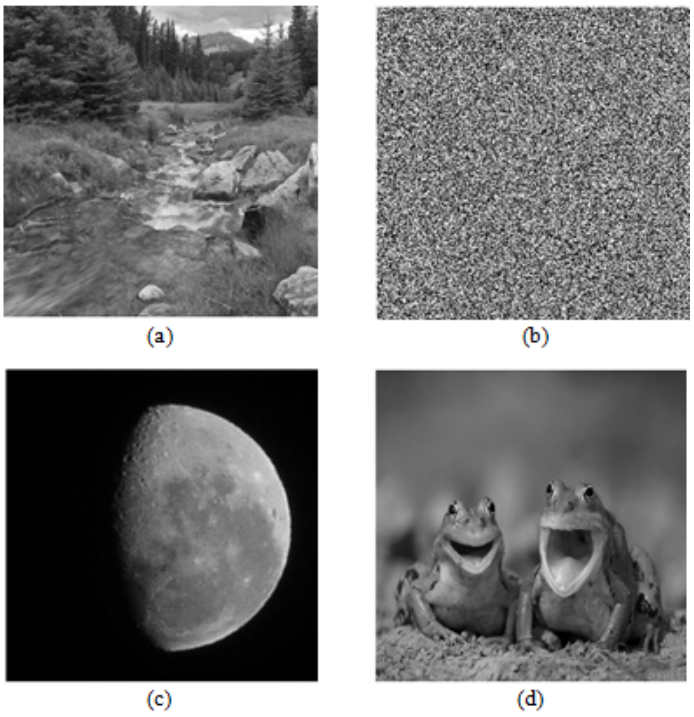

Figure 8 Reference image sensitivity analysis (a) Incorrect reference image (b) Incorrectly decoded image (c) Correct reference image (d) Correctly decoded image

In this analysis, the entropy of the correct reference image is 5.35447619; for the other one is 5.35447489 where 1-bit change is made in the middle pixel value. The results of the reference image sensitivity analyses show that the proposed algorithm is highly sensitive to the reference image even with only a single pixel change on it. In security manner, the proposed algorithm should also be sensitive to the system parameters of r and x_0. For this analysis, "Baboon" image is used as a secret image and "Liberty" image is used as a reference image. "Baboon" and "Liberty" images are used in the proposed algorithm for the parameters of $r = 4$ and $x_0 = 0.12345$. On the receiving side, a tinny change of 10^{-6} is applied to the one of the parameters while other remains same, and performs the inverse algorithm of the proposed scheme to decode secret image. The results are shown in Figure 10. According to the results, the proposed algorithm is highly sensitive to the all system

Table 1 Entropy values of different test images

Size	Name	Purpose of usage	Entropy
128x128	Flowers	Secret	7.44118
128x128	Frog	Reference	6.92897
256x256	Cameraman	Reference	7.10514
256x256	Frog	Secret	6.96951
256x256	Moon	Reference	5.35447
256x256	Baboon	Secret	7.22002
256x256	Landscape	Reference	7.34015
512x512	Baboon	Secret	7.18316
512x512	Liberty	Reference	7.49462
512x512	Peppers	Reference	7.59386
1024x1024	Cat	Secret	7.20682
1024x1024	Airplane	Reference	7.20493

parameters which enhance the security of the secret image.

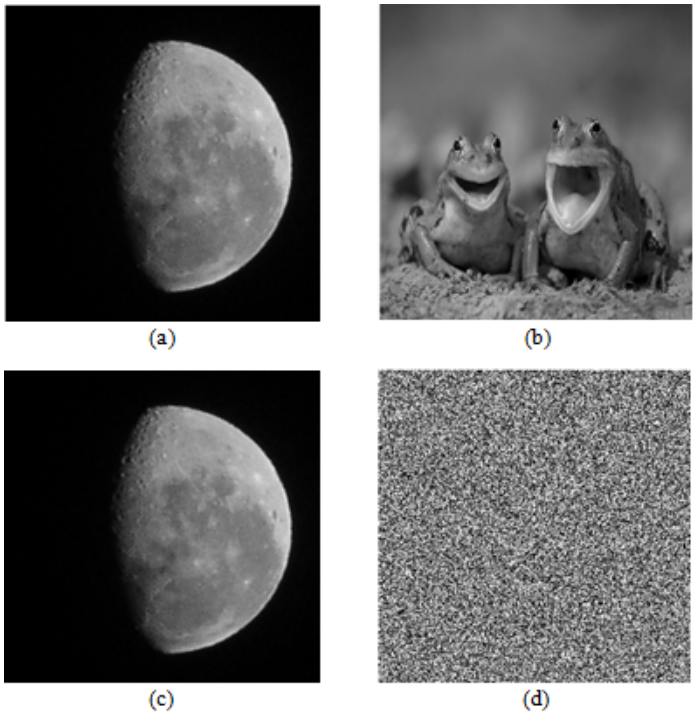

Figure 9 Entropy sensitivity analysis (a) Correct reference image (b) Correct decoded image (c) Similar reference image with only 1-bit change in the middle pixel value (d) Incorrectly decoded image

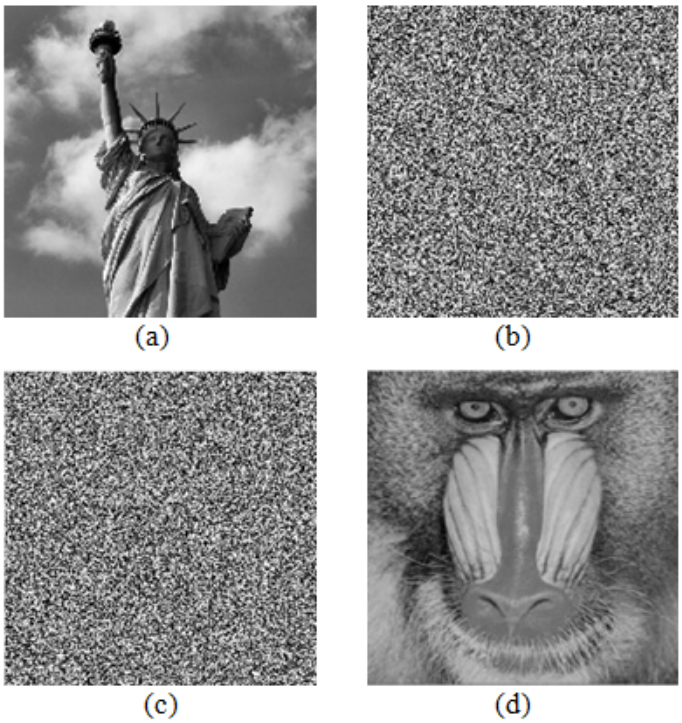

Figure 10 Parameters sensitivity analysis (a) Reference image coding with $r = 4$ and $x_0 = 0.12345$ (b) Incorrect decoded image using with $r = 4$ and $x_0 = 0.123451$ (c) Incorrect decoded image using with $r = 3.999999$ and $x_0 = 0.12345$ (d) Correct decoded image with $r = 4$ and $x_0 = 0.12345$

Entropy attack analysis

The entropy of an image measures the level of randomness or information content in that image, and is calculated as in Equation 4.

$$H = -\sum_{i=0}^{L-1} p(i)\log_2(p(i)) \tag{4}$$

where H is the entropy value, L is the total number of levels in an image and p(i) is the probability of the i-th level. In the proposed algorithm, the entropy value is utilized to directly modify the initial value of the LM, thereby affecting the generation of the chaotic binary sequence. However, it is important to note that even if the entropy values are the same, the pixel-level distribution and arrangement of bits within the image may still differ, which might still affect the encoding and decoding process. Moreover, the entropy value alone does not guarantee that the images are identical in terms of their content or structure and even slight differences in their pixel values can lead to variations in the generated secret codes. To perform an entropy attack analysis, the proposed algorithm uses two different images with the same entropy values. The Arnold Cat Map (ACM) method is then applied to permute a reference test image, generating a scrambled image with identical entropy. ACM is a method used in image processing to scramble the pixel positions of an image (Bhardwaj and Bhagat 2018). It is a transformation technique, as defined in Equation 5, which rearranges the pixel locations while preserving the image's overall properties, such as entropy and pixel values.

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix} \mod N \tag{5}$$

(x, y) and (x', y') represent the original coordinates of a pixel and the new coordinates after the transformation, respectively. N denotes the size of the image. For the entropy analysis, the test image of 'cameraman.jpg' with entropy value 7.10514 is chosen as reference image to be used for the secure transmission of 'baboon.jpg' image in the proposed scheme. During the decoding process of the secret image on the receiver side, the scrambled image obtained by applying the ACM method to the 'cameraman' image just one iteration is used. This leads to no change in entropy, and the visual results are shown in Figure 11.

The correlation coefficient between the reference image and the scrambled image is found to be 0.0529, while the correlation coefficient between the secret image and the decoded image is calculated as -0.0337. According to the entropy results, the secret image can only be decoded correctly if the exact same reference image, secret key and system parameters are used. The decoding process is highly sensitive to the reference image's entropy; even if a different image has the same entropy value, incorrect decoding may still occur because the actual pixel values can differ. This result verifies the sensitivity of the proposed algorithm to both the entropy and the precise structure of the reference image.

Differential analysis

NPCR (Number of Pixels Change Rate) and UACI (Unified Average Changing Intensity) are two metrics used to evaluate the differential analyses for the sensitivity of particular algorithms, especially in the context of image encryption (Louzzani *et al.* 2021). They are defined in Equations 6 and 7.

$$\text{NPCR} = \left(\frac{N_{\text{changed}}}{N_{\text{total}}}\right) \times 100 \tag{6}$$

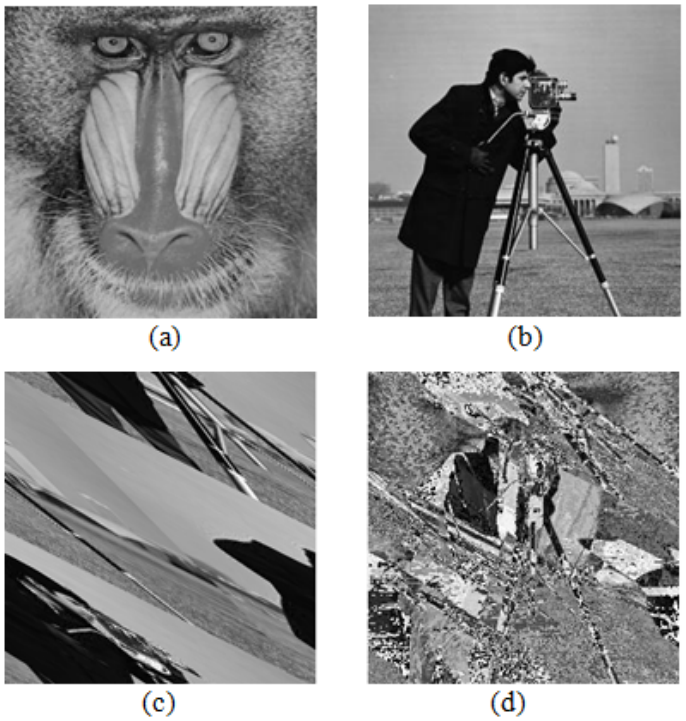

Figure 11 Entropy attack analysis (a) Secret image (b) Reference image (c) Scrambled image (d) Incorrectly decoded secret image

N_{changed} is the number of pixels that differ between the original and decoded image after a small change is made to the reference image. N_{total} is the total number of pixels in the secret image. A higher NPCR value, ideally close 100% indicates better sensitivity of the coding algorithm (Alawida 2023). UACI measures the average change intensity of the pixels. It gives an indication of how much the output changes with a small change in the input (Haridas *et al.* 2024).

$$\text{UACI} = \left(\frac{1}{N_{\text{total}}}\sum_{i=1}^{N_{\text{total}}}\frac{|C_1(i) - C_2(i)|}{255}\right) \times 100 \tag{7}$$

$C_1(i)$ and $C_2(i)$ are the pixel values of the original and decoded images, respectively. A higher UACI values, typically around 33%, indicates that the algorithm has strong sensitivity to changes in the input image. The theoretical values for NPCR and UACI are 99.60% and 33.46%, respectively (Jin *et al.* 2024). Random test images of different sizes are chosen to calculate the correlation coefficients, NPCR and UACI values for the proposed coding algorithm. First, all the secret test images are used to perform differential analyses with a key through the algorithm. Then, the pixel value in the middle of each reference image is incremented by one. Afterwards, the modified reference image is applied to the receiver to decode the secret image using the same system parameters. The corresponding results are listed in Table 2.

The results show that both NPCR and UACI values are very close to their ideal values, suggesting that the proposed algorithm is highly sensitive to the reference image. Moreover, the correlation coefficient is very close to 0, which means there is no relationship at all between the original and decoded images.

Table 2 Differential analysis of the proposed coding algorithm

Size	Secret Image	Cover Image	NPCR (%)	UACI (%)	Correlation Coefficient
128x128	Flowers	Frog	99.6459	32.4016	-0.004564
256x256	Baboon	Landscape	99.5986	29.5655	0.006156
512x512	Goldhill	Liberty	99.6208	31.6547	0.000108
1024x1024	Cat	Airplane	99.6023	30.5923	-0.001699

Table 3 Speed analyses of the proposed algorithm

Size of secret image	Average Coding time (sec)	Average Decoding time (sec)	Coding Rate (Mbps)	Decoding Rate (Mbps)
128x128	0.2691	0.1934	0.487	0.677
256x256	0.9380	0.7422	0.558	0.706
512x512	3.8149	2.3508	0.549	0.892
1024x1024	16.3716	10.4283	0.512	0.804

Speed analysis

In order to evaluate the running speed of the coding and decoding processes, 20 test images with different sizes are used in the proposed algorithm. Then average coding and decoding rates are calculated by using Matlab R2015a software. The average running time for the results can be found in Table 3.

Comparison with related works

The proposed coding algorithm can be considered as an alternative approach to steganographic algorithms. Generally, in steganographic algorithms, confidential information is hidden by manipulating the cover image in time or frequency domain. The coding process performed here is the modification of the random-like chaotic series against the secret image through bitwise operations. It is known that a secure cryptographic algorithm is extremely sensitive to system parameters. In the proposed algorithm, whole system sensitivity is achieved by affecting the initial value of the chaotic system by the information content of the reference image.The proposed scheme is compared with the closely related works of (Elshoush *et al.* 2021) and (Bai *et al.* 2017) in the field of zero image steganography. Table 4 summarizes the comparison of proposed scheme with the related works. This study (Elshoush *et al.* 2021), has proposed a zero distortion steganographic method that uses ASCII code matching for the character of secret message in a cover image without changing image size and pixel values. Here, only the positions of the secret message are noted in a mapping table. In extraction process, the ASCII codes of the secret messages are provided from the positions arrays in the cover image.

All related works of (Elshoush *et al.* 2021), (Bai *et al.* 2017) and (Bilal *et al.* 2013), include additional operations such as encryption, compression or mathematical transformation to increase the security of secret data. However, these processes should take complex structure, high power consumption and can increase time duration of hiding and extracting processes. The proposed scheme uses grayscale images for simpler implementation and efficient processing, particularly where color is unnecessary. Unlike methods that depend on color images, which increase complexity and computational demands, this approach omits encryption while maintaining security through chaos theory and inherent sensitivity. This ensures significant output changes from minor input variations, protecting secret images from unauthorized access. By avoiding encryption-related overhead, the scheme achieves high confidentiality, making it ideal for applications where speed and simplicity are essential. For instance, in (Elshoush *et al.* 2021), the secret message size with 2000 Bytes takes 9.5 seconds in "Cat1" image with a size of 200x200. For the proposed scheme, average time of different secret test images with 128x128 in size (16384 Bytes) takes about 0.27 seconds for coding process.

Evaluation of the proposed algorithm

The imperceptibility of a steganographic algorithm is the most important feature to consider. The ability of unnoticed as a first requirement is closely related to the strength of the steganographic algorithm. In the proposed algorithm, payload capacity is same as the volume of the reference image. Reference image can be thought as a cover image in this study. Therefore, larger volume of reference image means more payload capacity. Secret message should not leave a trace on the cover medium; otherwise it can be detected by steganalysis methods. The proposed algorithm makes no any change to the reference image and brings zero distortion, so it has a perfect imperceptibility and robustness. The evaluation of the proposed algorithm is given in Table 5.

Table 4 Comparison of other references

Criteria	Proposed scheme	Elshoush et al.	Bai et al.	Bilal et al.
Usage of RGB Component	Grayscale (8 bits)	All RGB (24 bits)	All RGB (24 bits)	Grayscale (8 bits)
Domain	Spatial	Spatial	Frequency	Spatial and Frequency
Hiding Data Type	Image	Message	Message	Image
Extracting Process	Bit by bit	Byte by byte	DCT Transform	DCT transform
Compression	No	Yes (Huffman algorithm)	No	Yes (Hash function)
Encryption	No	Yes (AES-128)	Yes (AES)	Yes (RSA)
Usage of positions	No	Yes	No	No
Invisibility	No change	No change	No change	Change
Usage of key	Yes	No	No	No
Precision sensitivity	Yes	No	-	Yes

Table 5 Evaluation of the proposed algorithm

Criteria	Evaluation
Invisibility	No change in reference image
Payload Capacity	As the volume of the reference image
Robustness against statistical attacks	Very high (zero distortion)
Independent of file format	Any format readable by Matlab
Unsuspicious files	Very high as no change in image

CONCLUSION

This paper introduces a new data coding algorithm designed to enhance the security of image communication through the innovative application of chaos theory. A key feature of this algorithm is its ability to recover secret images using only a reference image, specific system parameters and identical code series, rather than relying on direct transmission of secret data. This approach not only improves security but also minimizes the risk of unauthorized access to confidential information. Furthermore, the algorithm demonstrates remarkable robustness and security, exhibiting extreme sensitivity to changes in the reference image. Even minor alterations, such as a single-bit change, can lead to complete failure in decoding the secret image. This resilience is supported by metrics like NPCR and UACI, confirming the algorithm's strong defences against potential attacks. Theoretical and experimental results verify that the proposed algorithm is very efficient and usable, demonstrating its capability to securely transmit image data while maintaining rapid processing speeds and high levels of confidentiality. It is part of the future plans to design similar coding algorithms for different types of multimedia data and to use them in practical applications.

Availability of data and material

The data collected in this study are available from the corresponding author upon reasonable request.

Conflicts of interest

The author declares that there is no conflict of interest regarding the publication of this paper.

Ethical standard

The author has no relevant financial or non-financial interests to disclose.

LITERATURE CITED

Abba, A., J. S. Teh, and M. Alawida, 2024 Towards accurate keyspace analysis of chaos-based image ciphers. Multimedia Tools and Applications pp. 1–20.

Ahmad, I. and S. Shin, 2021 A novel hybrid image encryption–compression scheme by combining chaos theory and number theory. Signal Processing: Image Communication **98**: 116418.

Alawida, M., 2023 A novel chaos-based permutation for image encryption. Journal of King Saud University-Computer and Information Sciences **35**: 101595.

Bai, D., X. Chen, and M. Tian, 2017 A satellite communication zero steganography algorithm. Multimedia Tools and Applications **76**: 26447–26462.

Bhardwaj, R. and D. Bhagat, 2018 two level encryption of grey scale image through 2d cellular automata. Procedia Computer Science **125**: 855–861.

Bilal, M., S. Imtiaz, W. Abdul, and S. Ghouzali, 2013 Zero-steganography using dct and spatial domain. In *2013 ACS international conference on computer systems and applications (AICCSA)*, pp. 1–7, IEEE.

Bonny, T. and W. Al Nassan, 2024 Optimizing security and cost efficiency in n-level cascaded chaotic-based secure communication system. Applied System Innovation **7**: 107.

Effah-Poku, S., W. Obeng-Denteh, and I. Dontwi, 2018 A study of chaos in dynamical systems. Journal of Mathematics **2018**: 1808953.

Elshoush, H. T., I. A. Ali, M. M. Mahmoud, and A. Altigani, 2021 A novel approach to information hiding technique using ascii mapping based image steganography. J. Inf. Hiding Multim. Signal Process. **12**: 65–82.

Gafsi, M., N. Abbassi, M. A. Hajjaji, J. Malek, and A. Mtibaa, 2020 Improved chaos-based cryptosystem for medical image encryption and decryption. Scientific Programming **2020**: 6612390.

Gurunathan, K. and S. Rajagopalan, 2020 A stegano-visual cryptography technique for multimedia security. Multimedia Tools and Applications **79**: 3893–3911.

Haridas, T., S. Upasana, G. Vyshnavi, M. S. Krishnan, and S. S. Muni, 2024 Chaos-based audio encryption: Efficacy of 2d and 3d hyperchaotic systems. Franklin Open **8**: 100158.

Hosam, O. and N. Ben Halima, 2016 Adaptive block-based pixel value differencing steganography. Security and Communication Networks **9**: 5036–5050.

Huang, D. and J. Wang, 2020 High-capacity reversible data hiding in encrypted image based on specific encryption process. Signal Processing: Image Communication **80**: 115632.

Hussain, M. and M. Hussain, 2013 A survey of image steganography techniques. International Journal of Advanced Science and Technology **54**: 113–124.

Irsan, M. and S. C. Antoro, 2019 Text encryption algorithm based on chaotic map. In *Journal of Physics: Conference Series*, volume 1341, p. 062023, IOP Publishing.

Ismail, S. M., A. M. Ghidan, and P. W. Zaki, 2020 Novel chaotic random memory indexing steganography on fpga. AEU-International Journal of Electronics and Communications **125**: 153367.

Jaradat, A., E. Taqieddin, and M. Mowafi, 2021 A high-capacity image steganography method using chaotic particle swarm optimization. Security and Communication Networks **2021**: 6679284.

Jin, B., L. Fan, B. Zhang, R. Lei, and L. Liu, 2024 Image encryption hiding algorithm based on digital time-varying delay chaos model and compression sensing technique. Iscience **27**.

Khan, M. and H. M. Waseem, 2024 An efficient confidentiality scheme based on quadratic chaotic map and fibonacci sequence. AIMS Mathematics **9**: 27220–27246.

Kumar, A. and N. Raghava, 2019 Chaos-based steganography technique to secure information and integrity preservation of smart grid readings using wavelet. International Journal of Computers and Applications **44**: 57–63.

Liu, X., C. Li, S. S. Ge, and D. Li, 2022 Time-synchronized control of chaotic systems in secure communication. IEEE Transactions on Circuits and Systems I: Regular Papers **69**: 3748–3761.

Louzzani, N., A. Boukabou, H. Bahi, and A. Boussayoud, 2021 A novel chaos based generating function of the chebyshev polynomials and its applications in image encryption. Chaos, Solitons & Fractals **151**: 111315.

Mishra, R. and P. Bhanodiya, 2015 A review on steganography and cryptography. In *2015 International Conference on Advances in Computer Engineering and Applications*, pp. 119–122, IEEE.

Ogras, H., 2019 An efficient steganography technique for images using chaotic bitstream. International Journal of Computer Network and Information Security **15**: 21.

Oğraş, H. and M. Türk, 2013 Utilizing simulink and matlab graphical user interface in modelling and simulation of chaos-based digital modulation techniques. International Journal of Electrical Engineering Education **50**: 19–33.

Ozkaynak, F., 2020 A novel random number generator based on fractional order chaotic chua system. Elektronika ir Elektrotechnika **26**: 52–57.

Pradhan, A., K. R. Sekhar, and G. Swain, 2018 Digital image steganography using lsb substitution, pvd, and emd. Mathematical Problems in Engineering **2018**: 1804953.

Rana, S., F. K. Parast, B. Kelly, Y. Wang, and K. B. Kent, 2023 A comprehensive survey of cryptography key management systems. Journal of Information Security and Applications **78**: 103607.

Roy, M., S. Chakraborty, and K. Mali, 2021 A chaotic framework and its application in image encryption. Multimedia Tools and Applications **80**: 24069–24110.

Sharafi, J., Y. Khedmati, and M. Shabani, 2021 Image steganography based on a new hybrid chaos map and discrete transforms. Optik **226**: 165492.

Sharma, D. K., N. C. Singh, D. A. Noola, A. N. Doss, and J. Sivakumar, 2022 A review on various cryptographic techniques & algorithms. Materials Today: Proceedings **51**: 104–109.

Swain, G., 2018a Digital image steganography using eight-directional pvd against rs analysis and pdh analysis. Advances in multimedia **2018**: 4847098.

Swain, G., 2018b High capacity image steganography using modified lsb substitution and pvd against pixel difference histogram analysis. Security and communication networks **2018**: 1505896.

Umoh, E. A. and T. Wudil, 2016 Engineering applications of chaos. In *12th International Conference and Exhibition on Power and Telecommunications (ICEPT 2016)*, pp. 39–49.

Zhang, B. and L. Liu, 2023 Chaos-based image encryption: Review, application, and challenges. Mathematics **11**: 2585.

How to cite this article: Ogras, H. A new Data Coding Algorithm for Secure Communication of Image. *Chaos Theory and Applications*, 6(4), 284-293, 2024.