

# High capacity double precision image steganography based on chaotic maps

Salwan F. Salman Al Rubaie, Maher K. Mahmood Al-Azawi

Department of Electrical Engineering, College of Engineering, Al-Mustansiriyah University, Baghdad, Iraq

## Article Info

### Article history:

Received Feb 19, 2023

Revised May 10, 2023

Accepted May 24, 2023

### Keywords:

Chaotic maps

Cryptography

Double precision image

High capacity

Key space size

Least significant bit

Steganography

## ABSTRACT

Steganography is the process of hiding confidential information within non-secret multimedia such that the 3<sup>rd</sup> party cannot distinguish if there is a secret message in it or not. Whereas cryptography is the technique of using mathematical concepts to convert information into unreadable codes via a key. This paper will propose two approaches, lossless and lossy image steganography. Both of them will use cryptography and steganography based on three different chaotic maps to ensure information security. In the cryptography part, two chaotic maps will be used to encrypt the secret information, while in the steganography section, one chaotic map is used to embed the message. The secret information will be concealed in the least significant bits (LSBs) of the double-precision image's pixels. The double precision image is a high-quality image and can be represented in 64 bits per pixel for grayscale images, leading to a very high redundant bit. Simulation results show a high embedding capacity of 60.938% and 400% for lossless and lossy approaches respectively with a peak signal to noise ratio (PSNR) reach of 69.964 dB. Furthermore, this system is extremely secure due to the use of 3 chaotic maps with key space  $2^{448}$ .

This is an open access article under the [CC BY-SA](#) license.



## Corresponding Author:

Salwan F. Salman Al Rubaie

Department of Electrical Engineering, College of Engineering, Al-Mustansiriyah University

Baghdad, Iraq

Email: salwan7@uomustansiriyah.edu.iq

## 1. INTRODUCTION

Industries have expanded rapidly within the last decade, resulting in the widespread use of interactive media for data transport, particularly with regard to the internet [1]. Although it has a lot of benefits, one significant drawback is the confidentiality and anonymity of the information. The transmission often takes place over insecure network connections. As technology advanced, a lot of new apps appeared, and most people started using the internet to share information. This data may be presented in any digital format and in a variety of sizes. Transferring this data online may not always be secure, because it is extremely confidential information regarding is for government institutions, military purposes, or banking business [2]. The development of several publicly available technologies with the power of abusing the secrecy and confidentiality of the information being conveyed has raised the risk of hostile assaults, data leaking, and other violent activities. Cryptographic algorithms become a popular approach that uses a secret key to encrypt the information into a code then, the same key is used to decrypt it back into readable data. When information is encrypted into a code, the real information is hidden, but the code is still discernible to human perception, attracting attention and requiring more examinations. Hiding information (also called steganography) inside multimedia is a relatively modern area of study and is now widely used in this situation to conceal information making it indistinguishable from human perception [3].

Natural language steganography and technical steganography are the two subcategories of steganography methodologies. Natural language steganography, which is produced using text systems, is different from technical steganography, which is constructed utilizing other media like audio, image, and video. Text steganography can hide secret information such that it is impossible for a third party to decode it after it is sent to the intended recipient. Two other types of natural language steganography exist. Linguistic steganography is the first. The grammatical arrangement of the sentences in the text is crucial to this kind of steganography, whereas the second technique is text steganography, which modifies the text's words, lines, spaces, and any other text-related elements in order to conceal the secret message [4].

The hidden data are transmitted through image steganography techniques with a carrier acting as the secret information cover image. Due to the ubiquitous use of visual data transfers on the internet cloud and the substantial redundant data found in pictures, image steganography techniques are among the most widely used interactive media steganography approaches [5].

There are certain approaches that should be utilized to enhance concealing efficiency in order to embed secret information in multimedia files. Additionally, multimedia steganography is employed in five different domains, and each domain includes different strategies that help to improve the processing used to hide data. These domains include distortion, statistical, spread spectrum, spatial, and transform [6].

Least significant bits (LSBs) replacement is a commonly used steganographic technique. By simply substituting  $n$  LSBs of a pixel with information bits, this technique embeds secret data. Not all pixels, though, can withstand the same degree of change. This causes a significant decline in image quality. As a result, several innovative and complex LSB strategies have been put forth to address this flaw [7]. This paper proposes one of these approaches to solve this issue by employing a 64-bit double precision image with very high redundant bits.

Several multimedia data, like those used in military, entertainment, government, finance, companies, and academic facilities, require assurances of secrecy, integrity, and identification or authority in order to sustain their value [8]. In this respect, cryptographic algorithms, which appear to be an effective means of information protection, have many practical applications. Regardless of this, the most common binary encrypting data ciphers, including the advanced encryption standard (AES), rivest shamir adleman (RSA) technique, international data encryption algorithm (IDEA), data encryption standard (DES), do not seem to be suitable for interactive media data. This can be attributed to a variety of factors including significant correlations, very big size, substantial redundancy, and comparable gray-scale values. The use of chaos theory to develop the encryption algorithm has gained popularity recently. The major benefit of these cryptographic techniques derives from the fact that chaotic signals, despite the method employed to produce them, seem to the third party as noise. Furthermore, the control parameters and initial variables of these chaotic generating functions have a significant impact on the chaotic signal's temporal development [9].

Chaos, which is a common occurrence in nature, is the innate unpredictability of a deterministic system and the unique behavior of a nonlinear dynamical system [10]. In a chaos-based cryptographic algorithm, the encryption key is generated using a chaotic map. The chaotic maps have received a lot of praise for adding complexity and strength to cryptographic techniques. Due to their ease of form and application, one-dimensional maps including logistic, sine, and tent maps are often employed. Nevertheless, the properties of maps utilized in cryptosystems that are most desirable include high nonlinearity, extreme unpredictability, and a considerable chaotic range. These are essential characteristics required to repel various attacks [11].

A novel real 1-dimensional cosine polynomial (1-DCP) chaotic map was suggested in [12]. The proposed map's statistical examination reveals that it has an endless chaotic range, a substantially chaotic nature, and a basic structure. In [13], a new large parameters interval 1-dimensional sine chaotic system (1-DSCS) was introduced. According to the 1-DSCS examination, the system demonstrated high chaotic properties, high sensitivity, and a wide range of parameter values.

Huang [14] presented a 2-dimensional logistic-sine-cosine map (2D-LSCM). According to performance investigation, it has a broader chaotic range, a greater Lyapunov exponent, and more intricate chaotic behavior than the typical 2D Logistic map.

Single-precision float-point numbers represented in 32 bits are receiving more and more consideration due to high dynamic range (HDR) pictures. Based on Institute of Electrical and Electronics Engineers (IEEE) 754 standard, the secret message bits in [15] are carried by a float-point number with a single precision. This study employed 15 bits per pixel instead of a total of 32 bits to achieve a good compromise between security and payload with a peak signal to noise ratio (PSNR) of 75 dB by concealing the secret information bits within the mantissa spatial domain.

The discrete wavelet transform (DWT) approach-based improved image steganography was introduced in [16]. The suggested method makes use of blocking and secret key calculations, both of which are new concepts. In blocking, the least variation approach was utilized, whereas in the calculation of the secret key, the detail coefficients of the DWT concept and the least error matching standard were applied.

Testing showed that, in terms of PSNR and correlation coefficient, the suggested technique offered superior stego and secret images, where the PSNR reaches 44.84 dB in the Lena image.

Two new techniques for image steganography that make use of the Haar DWT were proposed in [17]. To reduce distortions on the carrier image during the hiding process, the secret information was concealed in the transform domain. The experiments demonstrated that, at a message size of 2000 bytes, the suggested approaches offered higher image quality with a PSNR  $\approx$  65 dB.

A straightforward yet effective image steganographic system based on discrete hadamard transform (DHT) was introduced in [18]. The simulation results showed that the suggested technique achieved good security and imperceptibility even in the hiding capacity of 100% or 8 bits per pixel (BPP). The proposed technique provided stego-images and recovered images with corresponding PSNRs of 37 dB and 35 dB respectively.

In [19], a novel hybrid image steganography technique based on the LSB replacement algorithms and enhanced modified signed digit (EMSD) was provided. The proposed study used the EMSD algorithm and the least significant k-bit for the LSB substitution process to conceal the secret information using  $n$  neighboring cover image pixels. As a result, it employed modification direction (EMD)-based techniques and had a greater embedding capacity than the EMSD algorithm. The experiments analysis showed that when the embedding capacity is 30.05% or 2.404 BPP, the stego image's PSNR is 43.46 dB.

In this paper, tag image file format (TIFF) images are used as cover images in the lossy and lossless approaches as they are considered very high-quality images. In TIFF images, the data is stored in double precision format. This format can be represented in 64 bits according to the IEEE 754 standard. This makes the size of these images very large as each pixel is represented in 64 bits instead of 8 bits for grayscale images. The proposed lossy and lossless image steganography approaches are based on LSBs and three novel chaotic maps. The 1-DSCS and 2D-LSCM chaotic maps are used for the cryptography process, whereas the 1-DCP chaotic map is utilized for the steganography process.

This study suggests a lossless technique by embedding any type of secret binary data format (such as jpg, tiff, rar, zip, and pdf) in the LSBs of the pixels of the carrier TIFF image, achieving a high PSNR of 69.964 dB, and an excellent hiding capacity of 60.938% or 39 BPP, where each pixel can be represented in 64 bits. Whereas in the lossy approach, the secret data is an image of the same dimensions as the carrier image. The simulation findings demonstrate that 4 secret TIFF images can be concealed in the carrier image successfully, achieving a hiding capacity of 400% or 256 BPP with a high PSNR of 69.964 dB and 108.957 dB for stego and recovered secret images, respectively.

The remainder of this study is divided into the following sections. Section 2 provides LSB, lossless, and lossy techniques. Section 3 covers the simulation results and discussion, whereas the conclusion is left till section 4.

## 2. METHOD

### 2.1. Least significant bit

Researchers frequently study steganography approaches that utilize pictures as a carrier. There are several methods for concealing data in images. Most data-hiding techniques used in image steganography aim to alter less crucial information included in the carrier image. LSB technique is the most widely used and straightforward way to hide data in a cover image. It has a large hiding capacity and a low computing complexity. In this technique, the cover image's pixel's LSBs are utilized to conceal the bits of confidential data. A stego-image is the term for the modified carrier image. For normal people, the quality of the changed image is essentially undetectable, but this approach can negatively affect other malicious procedural attacks like noising, cropping, and compression [20], [21]. To know how the LSB technique work, Table 1 shows an example of an red, green, and blue (RGB) 24-bit pixel with decimal values of 169, 144, and 207 for green, red, and blue channels respectively, and the secret bits are [1 0 0]. Then the process is as shown:

Table 1. An example of the LSB technique

24-bit pixel	Decimal form	Binary form	Secret bits	Stego pixel
Red pixel	144	10010000	1	10010001
Green pixel	169	10101001	0	10101000
Blue pixel	207	11001111	0	11001110

### 2.2. The proposed lossless approach

The binary data files (compressed data) are very sensitive to any alteration in their bits, where manipulation in 1 bit of data header could cause complete data corruption. In this approach, the confidential

binary data is encrypted, concealed, extracted, and then decrypted without any loss in data bits, where the embedding and extraction processes will be provided in steps. Figure 1 demonstrates the block diagram of the image steganography based on a lossless approach.

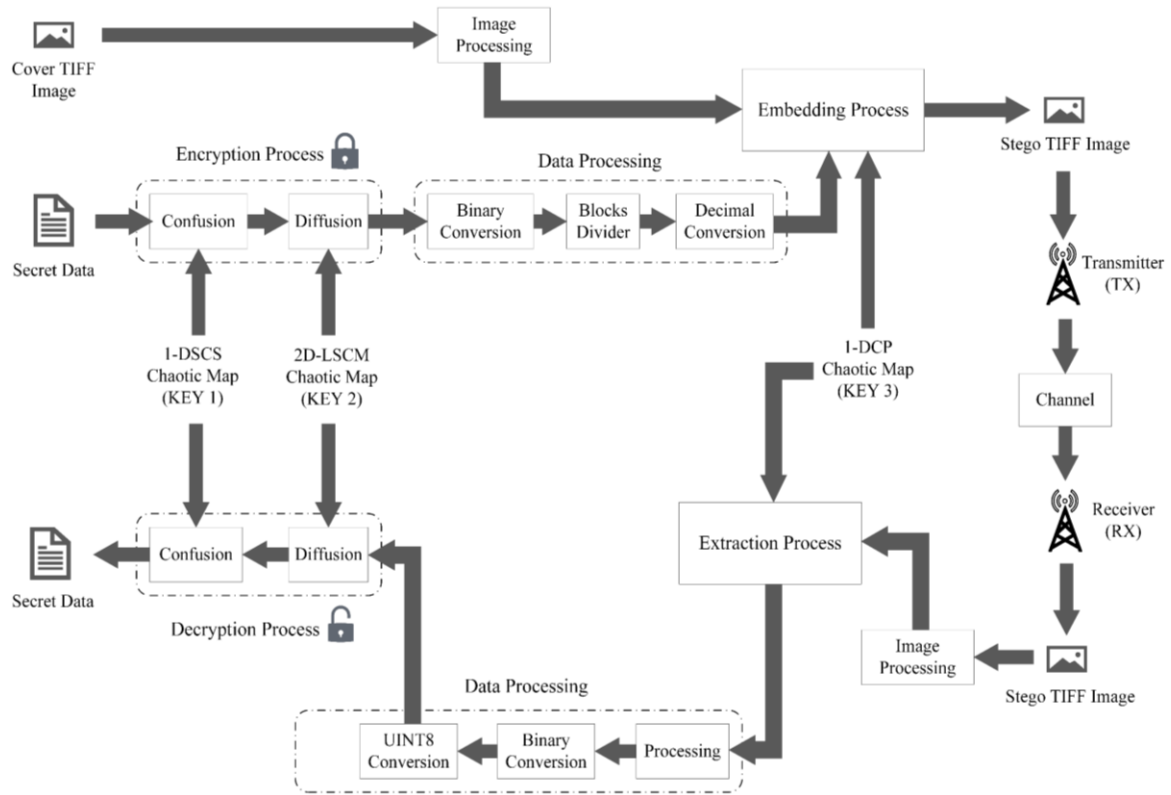


Figure 1. A block diagram of the proposed lossless approach

### 2.2.1. Embedding process

In the embedding stage, the private binary file bits will be ciphered and embedded randomly in the LSBs of the TIFF image's pixels with the use of three novel chaotic systems via secret keys as demonstrated in the following steps:

- Step 1: read any secret binary data (such as jpg, mp4, zip, and pdf). The binary data is read as an unsigned integer 8 (uint8).
- Step 2: encrypt the secret data using 1-DSCS and 2D-LSCM chaotic maps for confusion and diffusion processes respectively. The 1-DSCS chaotic map equation is defined as:

$$x_{n+1} = (\mu(3 + 2\lambda)(1 - \sin(\pi x_n))) \bmod 1 \quad (1)$$

The control parameters are  $\mu \in (4, +\infty)$  and  $\lambda \in (0, +\infty)$ . Whereas the 2D-LSCM equations are:

$$x_{n+1} = \sin(\pi x_n) + a(\sin(\pi x_n) - \sin^2(\pi x_n) + \sin(\pi y_n)) \quad (2)$$

$$y_{n+1} = \cos(\pi y_n) + b(\cos(\pi y_n) - \cos^2(\pi y_n) + \cos(\pi x_n)) \quad (3)$$

The control parameters  $a, b \in (0, +\infty)$ .

- Step 3: convert the encrypted secret data into binary bits, and store the result in a vector named S.
- Step 4: divide the binary vector S into blocks of length 39, where each block consists of 39 bits.
- Step 5: convert each block in S into a decimal number. The vector S now represents the encrypted secret data in double precision decimal numbers, where the maximum pixel value in  $S = 2^{39} - 1$  and the minimum pixel value = 0. The 64-bit double-precision number's computation precision is about  $10^{-15}$  according to the IEEE 754 standard for binary floating-point [22].

- Step 6: multiply all pixels in the vector  $\mathbf{S}$  by  $10^{-15}$  to make them in the range  $[0, 1]$ , where the double precision 64-bit floating-point can be represented in 15 digits. For instance, assume the vector  $\mathbf{S}$  consists of 4 random double precision pixels:

$\mathbf{S}=[37, 239834718, 1057, 89384721834]$

$\mathbf{S} \times 10^{-15} = [37 \times 10^{-15}, 239834718 \times 10^{-15}, \dots]$

Each floating-point pixel number can be represented with a precision of 15 digits as shown:

$37 \times 10^{-15} = 0.000000000000037$

And as mentioned before:

Maximum pixel value in  $\mathbf{S} = 2^{39} - 1 = 549755813887$

$549755813887 \times 10^{-15} = 0.000549755813887$

$\mathbf{S}$  in the range  $[0.000000000000000 - 0.000549755813887]$

The first 3 fractal digits specified in red color are left 0 for all pixels in vector  $\mathbf{S}$ , whereas the rest 12 fractal digits in green color represent the secret bits in double precision form. Finally, the encrypted secret data in vector  $\mathbf{S}$  is ready to be concealed in the LSBs of the carrier TIFF image.

- Step 7: read any cover 64-bit TIFF image, rescale it in the range  $[0, 1]$ , and store the result in a matrix named  $\mathbf{C}$ .
- Step 8: round the 3<sup>rd</sup> fractal digit after the point to the nearest integer for all pixels of the matrix  $\mathbf{C}$  (cover). Example, suppose the first 2 pixels of matrix  $\mathbf{C}$  are:

$0.847615290322759 \ 0.471945276881172$

Rounding the 3<sup>rd</sup> fractal digit will result in:  $0.848000000000000 \ 0.472000000000000$

This procedure is carried out on all pixels in the carrier TIFF image. The first 3 fractal digits have a value, whereas the rest are left at 0 value. A relationship is shown between the vector  $\mathbf{S}$  and the pixels of matrix  $\mathbf{C}$ . The vector  $\mathbf{S}$  pixels are in the range  $[0, 1]$ , and the first 3 fractal digits of each pixel are 0 as mentioned before. On the other hand, the pixels of matrix  $\mathbf{C}$  are also in the range  $[0, 1]$ , but their last 12 fractal digits are 0 as shown above in green color. This allows the secret data in vector  $\mathbf{S}$  to be embedded successfully in these 12 fractal digits since the maximum pixel value in vector  $\mathbf{S} = 0.000549755813887$ , which also has 12 digits, but with non-zero values. For example, assume:

Matrix  $\mathbf{C}$  pixel (cover)= $0.848000000000000$

Vector  $\mathbf{S}$  pixel (secret)= $0.000549755813887$

Stego pixel=cover+secret= $0.848549755813887$

- Step 9: convert matrix  $\mathbf{C}$  into a vector.
- Step 10: embed all pixels in vector  $\mathbf{S}$  randomly in the LSBs of the pixels of vector  $\mathbf{C}$  (cover) specified in blue color as shown above using the 1-DCP chaotic map equation as defined:

$$x_{n+1} = \cos(\mu(x_n^3 + x_n)) \quad (4)$$

The control parameter  $\mu \in (0, +\infty)$ .

The summation (or embedding) between the pixels in  $\mathbf{S}$  and LSBs of the pixels in  $\mathbf{C}$  as shown before is applied randomly using 1-DCP chaotic map to generate stego vector.

- Step 11: convert stego vector into a matrix (or image).
- Step 12: write the stego TIFF image.

### 2.2.2. Extraction process

In the extraction stage, the encrypted, concealed binary data bits are first extracted from the LSBs of the 64-bit TIFF picture's pixels, and then deciphered, all via using the same secret keys utilized in the embedding stage and as shown in the following steps:

- Step 1: read the stego TIFF image.
- Step 2: convert the stego image into a vector.
- Step 3: replace the first 3 fractal digits of all pixels of the stego vector with 0. For instance, assume:

A stego pixel= $0.848549755813887$

Then the first 3 fractal digits will be:  $0.848$

Secret pixel= $0.848549755813887 - 0.848$

The retrieved secret pixel= $0.000549755813887$ .

- Step 4: extract the secret pixels from the LSBs of the pixels of stego vector specified in green color as shown above using 1-DCP chaotic map with the exact same secret key used in the hiding process. Store the result in a vector named  $\mathbf{S}$ .
- Step 5: multiply all pixels in vector  $\mathbf{S}$  by  $10^{15}$  to make them in the range  $[0, 2^{39} - 1]$ .
- Step 6: convert all decimal pixels in vector  $\mathbf{S}$  into binary.

- Step 7: convert each 8 bits in vector **S** into uint8.
- Step 8: decrypt vector **S** using 1-DSCS and 2D-LSCM chaotic maps for confusion and diffusion processes respectively by utilizing the exact same private keys used in the encryption process to recover the confidential data.
- Step 9: write the secret binary data.

### 2.3. The proposed lossy approach

In this method, TIFF images are employed as secret data. As previously indicated, the cover image pixels have 12 fractal digits illustrated in green color with 0 values after the rounding procedure in step 8 of the embedding process of the lossless approach. As a consequence, four secret TIFF images will be concealed in the LSBs of the pixels of the carrier image.

#### 2.3.1. Embedding process

In this stage, 4 confidential TIFF pictures will be encrypted and concealed randomly in the LSBs of the carrier TIFF picture's pixels by the use of three new chaotic maps via private keys as presented in the steps below:

- Step 1: read any 4 secret TIFF images, rescale them in the range [0, 1], and store the result in **G**, **U**, **V**, and **Q** matrices respectively. The TIFF image is read as a double precision 64-bit floating point number. Step 2 and step 3 are the same as steps 8 and 9 in the "embedding process" of the "lossless approach", but this time it will be applied to **G**, **U**, **V**, and **Q** matrices.
- Step 4: multiply **G**, **U**, **V**, and **Q** vectors with  $10^3$ .
- Step 5: convert all pixels of **G**, **U**, **V**, and **Q** vectors into signed integer 16 (int16).
- Step 6: encrypt **G**, **U**, **V**, and **Q** vectors using 1-DSCS and 2D-LSCM chaotic maps for confusion and diffusion processes respectively.
- Step 7: convert back all pixels of the 4 vectors into double precision numbers, reshape them back into matrices, and multiply each one of them with  $10^{-3}$ .
- Step 8: multiply **G**, **U**, **V**, and **Q** matrices with  $10^{-6}$ ,  $10^{-9}$ ,  $10^{-12}$ , and  $10^{-15}$  respectively, and then apply the summation to all of them. Store the result in a new matrix named **W**.

Assume **G**'s pixel=0.00045200000000, **U**'s pixel=0.00000081700000, **V**'s pixel=0.00000000566000, and **Q**'s pixel=0.000000000000938. The summation of the 4 pixels is equal to 0.000452817566938 and it is stored in **W**'s pixel. The same process is applied until all pixels of matrix **W** are obtained.

- Step 9: convert matrix **W** into a vector. The 4 encrypted TIFF images stored in matrix **W** are now ready to be concealed in the LSBs of the pixels of the carrier image.
- Steps 10, 11, 12, and 13 are the same as steps 7, 8, 9, and 10 in the "embedding process" of the "lossless approach" mentioned on the previous page, but this time it will be applied on the secret data considered in matrix **W**.
- Step 14: convert back stego vector into a matrix.
- Step 15: write the stego TIFF image.

#### 2.3.2. Extraction process

In this process, the four ciphered, embedded TIFF images are first extracted from the LSBs of the cover TIFF image's pixels, and then decrypted, all via utilizing the same private keys of the three novel chaotic maps used in the concealing stage and as demonstrated below:

Steps 1, 2, 3, and 4 are the same as in the "extraction process" of the "lossless approach", but this time the result will be stored in vector **W**.

- Step 5: convert vector **W** into a matrix.
- Step 6: in each pixel of matrix **W**, the first 3 fractal digits with 0 value are left. The next 3 after them are taken to obtain matrix **G**, then the next other 3 after them are taken as well to obtain matrix **U**, and so on ... until **V**, and **Q** matrices are obtained.
- Step 7: multiply **G**, **U**, **V**, and **Q** matrices with  $10^6$ ,  $10^9$ ,  $10^{12}$ , and  $10^{15}$  respectively.
- Step 8: convert **G**, **U**, **V**, and **Q** matrices into vectors.
- Step 9: repeat steps 4 and 5 in the "embedding process" of the "lossy approach".
- Step 10: decrypt **G**, **U**, **V**, and **Q** vectors using 1-DSCS and 2D-LSCM chaotic maps for confusion and diffusion processes respectively with the same keys used in the encryption process.
- Step 11: repeat step 7 in the "embedding process" of the "lossy approach".
- Step 12: write the 4 secret TIFF images.



The block diagram of the lossy approach is the same as the lossless approach shown in Figure 1, except for two differences. First, the input secret data is 4 TIFF images instead of binary data, whereas the second difference is considered by replacing the 3 blocks of the “data processing” in the embedding process with multiplier and summation blocks as used in steps 7 and 8. While in the extraction process, the 3 blocks of the “data processing” is replaced with subtraction and multiplier blocks as used in steps 6 and 7.

### 3. SIMULATION RESULTS AND ANALYSIS

The results of simulation tests and the accompanying analysis are provided in this part to validate the suggested lossless and lossy techniques. The tests are carried out on a windows PC with an Intel (R) Core (TM) i7-8550U CPU running at 1.80 GHz and 8GB of RAM with 1 TB SSD using MATLAB R2022b environment. A well-known dataset in [23], [24] is employed in both approaches. Figure 2 shows the 512×512 picture resolution of RGB and grayscale images like: Figure 2(a) lena, Figure 2(b) house, Figure 2(c) baboon, Figure 2(d) jet, Figure 2(e) boat, Figure 2(f) peppers, Figure 2(g) lake, and Figure 2(h) cameraman, that are chosen as test images to compare the performance with other techniques.

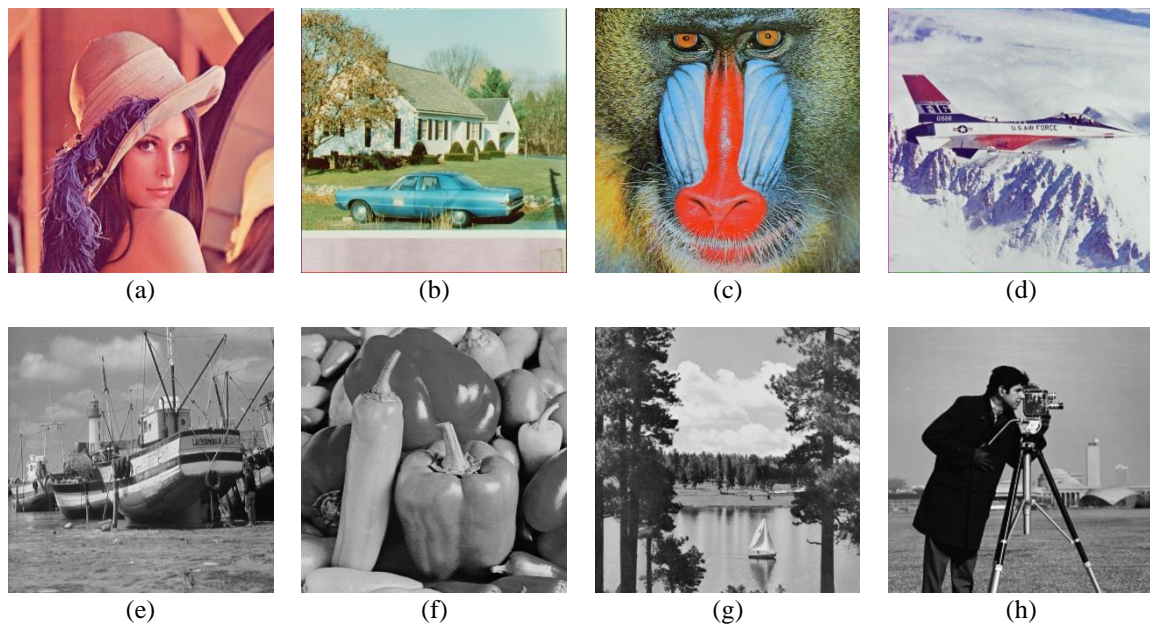


Figure 2. A dataset test images: (a) lena, (b) house, (c) baboon, (d) jet, (e) boat, (f) peppers, (g) lake, and (h) cameraman

#### 3.1. Performance analysis

The performance analysis metrics mentioned in this study are embedding capacity and imperceptibility. The embedding capacity (EC) can be defined as the ratio of amounts of embedded secret bits to the cover image size in bits, and calculated as:

$$EC = \frac{\text{Total Number of Embedded Secret Bits}}{\text{Size of Cover Image in Bits}} \times 100\% \quad (5)$$

Whereas the imperceptibility measure of the stego-image is assessed using the PSNR, structural similarity index measure (SSIM), and correlation coefficient (CC). These measurements are typically employed to quantify the distortion produced by the hiding procedure into the stego image.

The human visual system (HVS) cannot distinguish between the stego-image and the original image when the value of PSNR is over 36 dB [25]. The PSNR can be mathematically expressed as [26]:

$$PSNR = 10 \log_{10} \left( \frac{A^2}{MSE} \right) \quad (6)$$

Where  $A$  denotes the maximum pixel value of the cover image  $C$ . Mean square error (MSE) can be defined as [26]:

$$MSE = \frac{1}{M \times N} \sum_{i=1}^m \sum_{j=1}^n [C(i,j) - S(i,j)]^2 \quad (7)$$

Where  $N$  and  $M$  represent the dimensions of the cover image,  $m$  and  $n$  represent the number of rows and columns respectively, and  $C(i,j)$  and  $S(i,j)$  are the pixels of the cover and stego images respectively specified in  $i$  and  $j$ . The SSIM can be defined mathematically as [27]:

$$SSIM = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \quad (8)$$

Where  $\mu_x$  and  $\mu_y$  stand for the mean values of cover image  $\mathbf{C}$  and stego image  $\mathbf{S}$ . The variance of  $\mathbf{C}$ , the variance of  $\mathbf{S}$ , and the covariance of  $\mathbf{C}$  and  $\mathbf{S}$  are represented by the symbols  $\sigma_x$ ,  $\sigma_y$ , and  $\sigma_{xy}$ , respectively. The values  $(0.01 \times A)^2$  and  $(0.03 \times A)^2$  are assigned to the constant factors  $c_1$  and  $c_2$ . The CC can be expressed as [16]:

$$CC = \frac{\sum_{i=1}^m \sum_{j=1}^n (C(i,j) - \bar{C})(S(i,j) - \bar{S})}{\sqrt{\sum_{i=1}^m \sum_{j=1}^n [C(i,j) - \bar{C}]^2} \sqrt{\sum_{i=1}^m \sum_{j=1}^n [S(i,j) - \bar{S}]^2}} \quad (9)$$

Where  $\bar{C}$  and  $\bar{S}$  are the mean values of the cover and stego images respectively.

### 3.2. Security analysis

The term “security” in steganography systems is related to “unnoticeability”. Therefore, any data-hiding method is considered safe if the secret information cannot be determined by statistical analysis or removed after being discovered by the intruder [28]. The safe transmission of sensitive data is a crucial need for steganographic methods. In order to prevent data exposure by malicious hackers when sending across an open route, security is thus of utmost importance.

In this paper, the probability distribution of cover image (PDC) and probability distribution of stego image (PDS) have been compared using relative entropy or discrimination to assess the security of the steganography system. The relative entropy (E) for both approaches can be defined mathematically as [16], [29]:

$$E(PDC || PDS) = \sum PDC \left| \log \left( \frac{PDC}{PDS} \right) \right| \quad (10)$$

For identical images, E is approximately zero. Table 2 displays the relevant quantitative analysis of both approaches, which include EC, PSNR, SSIM, CC, and relative entropy, whereas Table 3 shows the PSNR, SSIM, CC, and relative entropy for different retrieved secret pictures of the lossy approach. The experiments show that the proposed lossless and lossy approaches have high embedding capacities of 60.938% and 400% respectively as well as imperceptibility compared with the existing techniques. The PSNR, SSIM, and CC of the stego image for both approaches are the same, and they reach 69.964 dB, 0.999999, and 1.000000 respectively at the maximum case in baboon image, whereas 66.828 dB, 0.999961, and 0.999999 respectively at the minimum case in lake image. These measurements are considered to be very high and satisfactory, and as a result, the normal human eye cannot detect any distortion in the stego image after the concealing process. Figure 3 illustrates a trade-off between imperceptibility and security with the embedding capacity by plotting Figure 3(a) PSNR vs BPP, Figure 3(b) SSIM vs BPP, Figure 3(c) CC vs BPP, and Figure 3(d) relative entropy vs BPP. Where BPP varies from 1 to 46, whereas the comparative results with the other schemes have been demonstrated in Table 4.

Table 2. BPP, PSNR, SSIM, CC, and relative entropy for different cover images for both approaches

Cover image	Embedding capacity (%)		Stego-image for lossless and lossy			
	Lossless	Lossy	PSNR (dB)	SSIM	CC	Relative entropy
Lena			67.457	0.999997	0.999998	0.000277
House			67.242	0.999986	0.999999	0.000224
Baboon			69.964	0.999999	1.000000	0.000234
Jet	60.938	400	67.176	0.999954	0.999998	0.000129
Boat			67.460	0.999962	0.999998	0.000276
Peppers			68.052	0.999967	0.999999	0.000250
Lake			66.828	0.999961	0.999999	0.000291
Cameraman			67.638	0.999949	0.999999	0.000285



Table 3. PSNR, SSIM, CC, and relative entropy for different secret images for the lossy approach

Secret image	Retrieved secret image for lossy approach			
	PSNR (dB)	SSIM	CC	Relative entropy
Lena	70.850	0.999998	0.999999	0.000200
House	70.676	0.999989	0.999999	0.000159
Baboon	108.957	1.000000	1.000000	0.000000
Jet	70.810	0.999965	0.999999	0.000090
Boat	70.929	0.999972	0.999999	0.000194
Peppers	71.085	0.999978	0.999999	0.000175
Lake	70.242	0.999971	0.999999	0.000209
Cameraman	71.124	0.999967	0.999999	0.000198

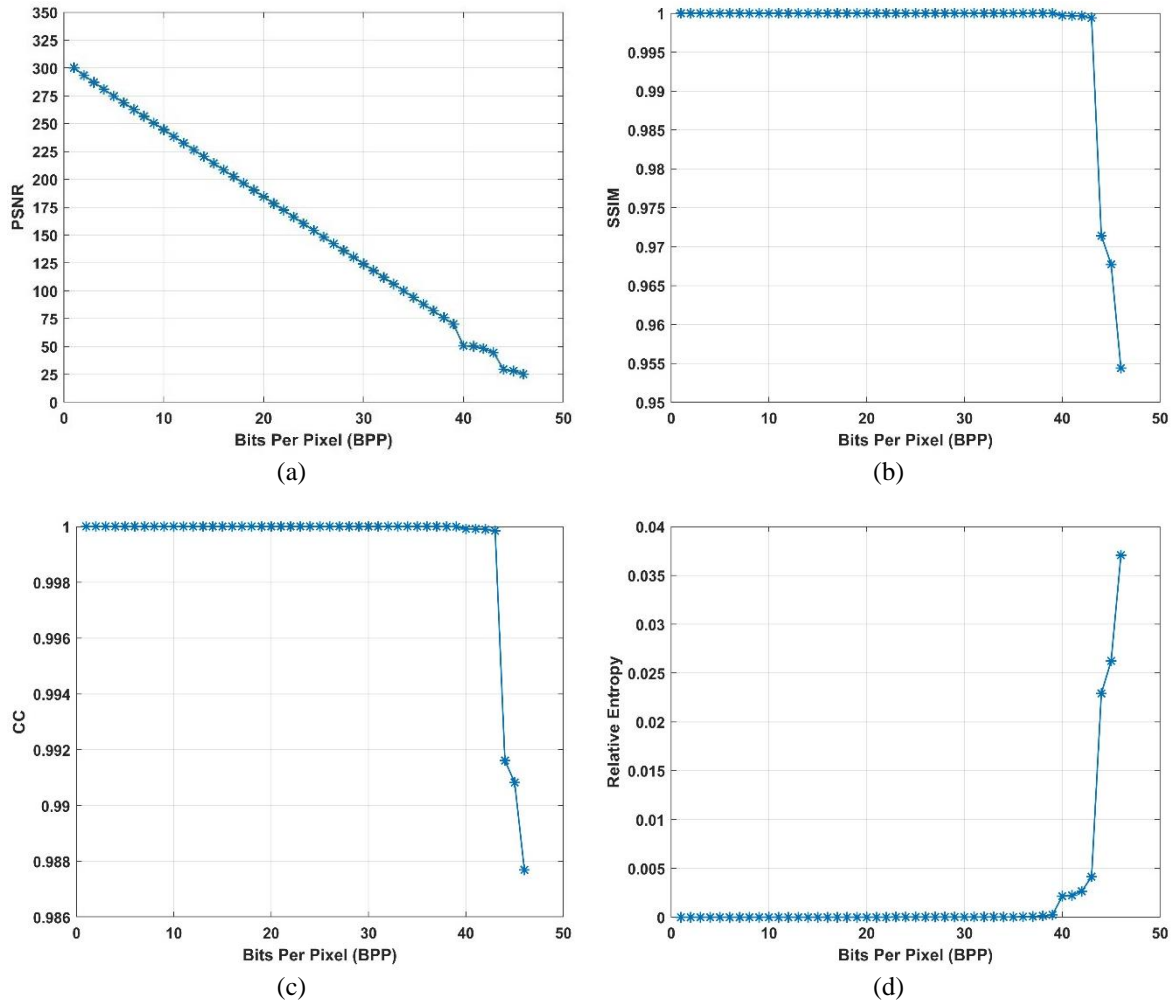


Figure 3. A trade-off between imperceptibility and security with embedding capacity: (a) PSNR vs BPP, (b) SSIM vs BPP, (c) CC vs BPP, and (d) relative entropy vs BPP

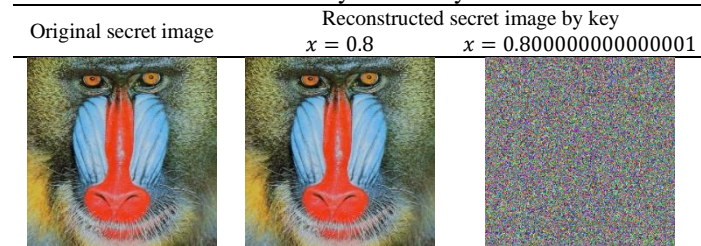
### 3.3. Key sensitivity analysis

This section presents the key sensitivity of the cryptography and the steganography systems in terms of chaotic maps used in this study. The simulation results showed that the system of both approaches is very sensitive to any initial value of any of the 3 chaotic maps 1-DSCS, 2D-LSCM, and 1-DCP. Table 5 demonstrates that changing the initial value  $x$  from 0.8 to 0.8000000000000001 in any of the 3 chaotic maps is far enough to generate a new distinct random sequence which is used in the encryption and embedding processes for both approaches. As a result, the security of the proposed methods is very superior.

Table 4. Comparison with currently available high-capacity image steganography techniques

Technique	Cover image	Embedding capacity (%)		PSNR (dB)	SSIM	Security
		Lossless	Lossy			
[15], (2020)	-	46.875	-	75	-	Minimal total testing error
[16], (2018)	Lena	0.3814	-	67.491	-	Relative entropy measure
[17], (2016)	Baboon			67.474	-	
	Pepper			67.467	0.918500	
[18], (2022)	Lena	-	100	37.467	0.918500	RS-attack and chi-square attack
	House			37.413	0.933100	
	Baboon			37.268	0.971300	
	Jet			37.249	0.906600	
	Boat			37.387	0.923800	
	Peppers			37.151	0.906700	
	Lake			37.369	0.940600	
[19], (2020)	K=1	30.05	-	43.460	0.985000	-
	K=2	42.55		37.280	0.943000	
	K=3	55.05		31.200	0.822000	
The proposed	Lena	60.938	400	67.457	0.999997	Relative entropy measure
	House			67.242	0.999986	
	Baboon			69.964	0.999999	
	Jet			67.176	0.999954	
	Boat			67.460	0.999962	
	Peppers			68.052	0.999967	
	Lake			66.828	0.999961	
	Cameraman			67.638	0.999949	

Table 5. Key sensitivity test



### 3.4. Key space analysis

The key space is related to the total number of secure and confidential keys that may be employed in cryptography algorithms. The total number of keys is  $10^k$ , with each key being represented by  $k$  binary bits. A very large key space makes brute force attacks extremely impossible to succeed [30]. In this paper, the key in both approaches is a combination of 1-DSCS and 2D-LSCM chaotic maps for the encryption process, and 1-DCP chaotic map for the embedding process. The 1-DSCS chaotic map inputs consist of 1 initial value  $x$  and 2 parameters, the 2D-LSCM chaotic map inputs have 2 initial values  $x$  and  $y$ , and 2 parameters, whereas the 1-DCP chaotic map has 1 initial value with 1 parameter. As a result, 9 keys have been generated in the entire system with a precision of  $10^{-15}$  for each, resulting in a large key space size equal to  $10^{15 \times 9} = 10^{135}$  or  $2^{448}$ , where the no. of bits to assign a key  $= \log_2 10^{135} \approx 448$  bits.

## 4. CONCLUSION

The lossless and lossy image steganography techniques are suggested in this study, where the spatial domain is chosen for both schemes to conceal the secret data by embedding it in the LSBs of the pixels of the double-precision image. To ensure data security, three distinct novel chaotic maps have been employed in the steganography and cryptography stages. Two chaotic maps (1-DSCS and 2D-LSCM) are used in the cryptography section to encrypt the confidential data, whereas one chaotic map (1-DCP) is utilized in the steganography stage to hide the secret data in randomly chosen pixel locations. For grayscale images, the double precision image may be represented with 64 bits per pixel, resulting in an unusually large redundant bit. The findings of the experiments demonstrate a high embedding capacity of 60.938% for the lossless technique and 400% for the lossy method, along with an extraordinary PSNR reach of 69.964 dB. The usage of three chaotic maps with a key space size of up to  $2^{448}$  makes this system exceptionally secure and uncrackable against all kinds of brute-force attacks. Due to the fact that both approaches use the spatial domain to embed the secret data, their main drawback is that they are not robust to violent attacks like noise, cropping, and compression.

## ACKNOWLEDGMENT

The authors thank Al-Mustansiriya University (www.uomustansiriya.edu.iq) Baghdad, Iraq for their support to this work.




## REFERENCES

- [1] O. M. Osman, M. E. A. Kanona, M. K. Hassan, A. A. E. Elkhair, and K. S. Mohamed, "Hybrid multistage framework for data manipulation by combining cryptography and steganography," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 1, pp. 327–335, Feb. 2022, doi: 10.11591/eei.v11i1.3451.
- [2] J. N. Shehab, H. A. Abdulkadhim, and T. F. H. Al-Tameemi, "Robust large image steganography using LSB algorithm and 5D hyper-chaotic system," *Bulletin of Electrical Engineering and Informatics*, vol. 10, no. 2, pp. 689–698, Apr. 2021, doi: 10.11591/eei.v10i2.2747.
- [3] N. Subramanian, O. Elharouss, S. Al-Maadeed, and A. Bouridane, "Image Steganography: A Review of the Recent Advances," *IEEE Access*, vol. 9, pp. 23409–23423, 2021, doi: 10.1109/ACCESS.2021.3053998.
- [4] R. Din, R. Bakar, S. Utama, J. Jasmis, and S. J. Elias, "The evaluation performance of letter-based technique on text steganography system," *Bulletin of Electrical Engineering and Informatics*, vol. 8, no. 1, pp. 291–297, Mar. 2019, doi: 10.11591/eei.v8i1.1440.
- [5] M. R. D. Farahani and A. Parsayan, "A DWT Based Perfect Secure and High Capacity Image Steganography Method," in *2013 International Conference on Parallel and Distributed Computing, Applications and Technologies*, IEEE, Dec. 2013, pp. 314–317. doi: 10.1109/PDCCAT.2013.56.
- [6] F. Q. A. Alyousuf, R. Din, and A. J. Qasim, "Analysis review on spatial and transform domain technique in digital steganography," *Bulletin of Electrical Engineering and Informatics*, vol. 9, no. 2, Apr. 2020, doi: 10.11591/eei.v9i2.2068.
- [7] C.-H. Yang, C.-Y. Weng, S.-J. Wang, and H.-M. Sun, "Adaptive Data Hiding in Edge Areas of Images With Spatial LSB Domain Systems," *IEEE Transactions on Information Forensics and Security*, vol. 3, no. 3, pp. 488–497, Sep. 2008, doi: 10.1109/TIFS.2008.926097.
- [8] O. M. Al-Hazaimah, M. F. Al-Jamal, N. Alhindawi, and A. Omari, "Image encryption algorithm based on Lorenz chaotic map with dynamic secret keys," *Neural Computing and Applications*, vol. 31, no. 7, pp. 2395–2405, Jul. 2019, doi: 10.1007/s00521-017-3195-1.
- [9] O. M. Al-hazaimah, A. A. Abu-Ein, M. M. Al-Nawashi, and N. Y. Gharaibeh, "Chaotic based multimedia encryption: a survey for network and internet security," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 4, pp. 2151–2159, Aug. 2022, doi: 10.11591/eei.v11i4.3520.
- [10] W. Yan, Z. Jiang, X. Huang, and Q. Ding, "A Three-Dimensional Infinite Collapse Map with Image Encryption," *Entropy*, vol. 23, no. 9, p. 1221, Sep. 2021, doi: 10.3390/e23091221.
- [11] J. S. Muthu and P. Murali, "A new chaotic map with large chaotic band for a secured image cryptosystem," *Optik*, vol. 242, p. 167300, Sep. 2021, doi: 10.1016/j.ijleo.2021.167300.
- [12] M. Z. Talhaoui, X. Wang, and M. A. Midoun, "A new one-dimensional cosine polynomial chaotic map and its use in image encryption," *The Visual Computer*, vol. 37, no. 3, pp. 541–551, Mar. 2021, doi: 10.1007/s00371-020-01822-8.
- [13] X. Wang and P. Liu, "A New Image Encryption Scheme Based on a Novel One-Dimensional Chaotic System," *IEEE Access*, vol. 8, pp. 174463–174479, 2020, doi: 10.1109/ACCESS.2020.3024869.
- [14] H. Huang, "Novel Scheme for Image Encryption Combining 2D Logistic-Sine-Cosine Map and Double Random-Phase Encoding," *IEEE Access*, vol. 7, pp. 177988–177996, 2019, doi: 10.1109/ACCESS.2019.2958319.
- [15] Y. Huo, Y. Qiao, and W. Gao, "High Capacity Steganography on Float-Point Number with Single Precision," in *2020 2nd International Conference on Video, Signal and Image Processing*, New York, NY, USA: ACM, Dec. 2020, pp. 48–54. doi: 10.1145/3442705.3442713.
- [16] V. Kumar and D. Kumar, "A modified DWT-based image steganography technique," *Multimedia Tools and Applications*, vol. 77, no. 11, pp. 13279–13308, Jun. 2018, doi: 10.1007/s11042-017-4947-8.
- [17] Y. Taouil, E. B. Ameer, A. Souhar, A. El, and M. T. Belghiti, "High Imperceptibility Image Steganography Methods based on HAAR DWT," *International Journal of Computer Applications*, vol. 138, no. 10, pp. 38–43, Mar. 2016, doi: 10.5120/ijca2016908984.
- [18] Y.-Q. Zhang, K. Zhong, and X.-Y. Wang, "High-Capacity Image Steganography Based on Discrete Hadamard Transform," *IEEE Access*, vol. 10, pp. 65141–65155, 2022, doi: 10.1109/ACCESS.2022.3181179.
- [19] S. Solak, "High Embedding Capacity Data Hiding Technique Based on EMSD and LSB Substitution Algorithms," *IEEE Access*, vol. 8, pp. 166513–166524, 2020, doi: 10.1109/ACCESS.2020.3023197.
- [20] R. K. Thakur and C. Saravanan, "Analysis of steganography with various bits of LSB for color images," in *2016 International Conference on Electrical, Electronics, and Optimization Techniques (ICEEOT)*, IEEE, Mar. 2016, pp. 2154–2158. doi: 10.1109/ICEEOT.2016.7755073.
- [21] P. Das, S. C. Kushwaha, and M. Chakraborty, "Multiple embedding secret key image steganography using LSB substitution and Arnold Transform," in *2015 2nd International Conference on Electronics and Communication Systems (ICECS)*, IEEE, Feb. 2015, pp. 845–849. doi: 10.1109/ECS.2015.7125033.
- [22] "IEEE Standard for Binary Floating-Point Arithmetic," *ANSI/IEEE Std 754-1985*, pp. 1–20, 1985, doi: 10.1109/IEEESTD.1985.82928.
- [23] "Image Databases," [https://www.imageprocessingplace.com/root\\_files\\_V3/image\\_databases.htm](https://www.imageprocessingplace.com/root_files_V3/image_databases.htm) (accessed Dec. 21, 2022).
- [24] "SIPI Database, Volume 3: Miscellaneous," *University of Southern California*. <https://sipi.usc.edu/database/database.php?volume=misc> (accessed Dec. 21, 2022).
- [25] R. O. El Safy, H. H. Zayed, and A. El Dessouki, "An adaptive steganographic technique based on integer wavelet transform," in *2009 International Conference on Networking and Media Convergence*, IEEE, Mar. 2009, pp. 111–117. doi: 10.1109/ICNM.2009.4907200.
- [26] E. H. J. Halboos and A. M. Albakry, "Hiding text using the least significant bit technique to improve cover image in the steganography system," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 6, pp. 3258–3271, Dec. 2022, doi: 10.11591/eei.v11i6.4337.
- [27] S. Alsamarace and A. S. Ali, "A crypto-steganography scheme for IoT applications based on bit interchange and crypto-system," *Bulletin of Electrical Engineering and Informatics*, vol. 11, no. 6, pp. 3539–3550, Dec. 2022, doi: 10.11591/eei.v11i6.4194.




- [28] I. J. Kadhim, P. Premaratne, P. J. Vial, and B. Halloran, "Comprehensive survey of image steganography: Techniques, Evaluations, and trends in future research," *Neurocomputing*, vol. 335, pp. 299–326, Mar. 2019, doi: 10.1016/j.neucom.2018.06.075.
- [29] C. Cachin, "An information-theoretic model for steganography," *Information and Computation*, vol. 192, no. 1, pp. 41–56, Jul. 2004, doi: 10.1016/j.ic.2004.02.003.
- [30] I. Hussain, A. Anees, H. Alkhalidi, M. Aslam, N. Siddiqui, and A. Rehan, "Image encryption based on Chebyshev chaotic map and S8 S-boxes," *Optica Applicata*, vol. XLIX, no. 2, 2019, [Online]. Available: [https://opticaapplicata.pwr.edu.pl/files/pdf/2019/no2/optappl\\_4902p317.pdf](https://opticaapplicata.pwr.edu.pl/files/pdf/2019/no2/optappl_4902p317.pdf)

## BIOGRAPHIES OF AUTHORS



**Salwan F. Salman Al Rubaie**    was born in Baghdad, Iraq in 1997. He received his B.Sc. degree in Electrical Engineering in 2020 from Al-Mustansiriyah University, Iraq. Currently, he is an M.Sc. student in the Department of Electrical Engineering, College of Engineering, Al-Mustansiriyah University, Iraq. His research interests include data compression, digital steganography and watermarking, information security, cryptography algorithms, and chaos theory. He can be contacted at email: [salwan7@uomustansiriyah.edu.iq](mailto:salwan7@uomustansiriyah.edu.iq).



**Maher K. Mahmoud Al-Azawi**    was born in Baghdad, Iraq in 1958. He received his B.Sc. degree in Electrical Engineering in 1980 and his M.Sc. degree in Electronics and Communication Engineering in 1983, both from the University of Baghdad, Iraq. He is the head of the communications group/Elec. Engineering Department/College of Engineering/Al-Mustansiriyah University. He is the author of more than 50 publications published in international and local journals and conferences. Most of them are within the subjects of digital communications, secure communications, digital speech, and image processing. He can be contacted at email: [maher.alazawi@uomustansiriyah.edu.iq](mailto:maher.alazawi@uomustansiriyah.edu.iq).