International Journal of Communication Networks and Information Security

*ISSN: 2073-607X, 2076-0930*
*Volume 15 Issue 03 Year 2023*

# An Adaptive Fractal Image Steganography Using Mandelbrot and Linear Congruent Generator

**Mohammed J. Bawaneh**
*Associated professor, Information Technology Department, Al-Huson University College, Al-Balqa Applied University, Irbid, Jordan*
*dr_mjab@bau.edu.jo*
**Atef A. Obeidat***
*Associated professor, Information Technology Department, Al-Huson University College, Al-Balqa Applied University, Irbid, Jordan*
*dr.atefob@bau.edu.jo*
**Obaida M. Al-Hazaimeh**
*Professor, Information Technology Department, Al-Huson University College, Al-Balqa Applied University, Irbid, Jordan*
*dr_obaida@bau.edu.jo*
**Malek M. Al-Nawashi**
*Assistant professor, Information Technology Department, Al-Huson University College, Al-Balqa Applied University, Irbid, Jordan*
*nawashi@bau.edu.jo*
**Amaal Rateb Shorman**
*Assistant professor, Information Technology Department, Al-Huson University College, Al-Balqa Applied University, Irbid, Jordan*
*amal.shorman@bau.edu.jo*

***Article History***

Received: 29 September 2023
Revised: 10 November 2023
Accepted: 11 December 2023

***Abstract***

Despite the advancements that occurred in the field of technology, information security (i.e., IS) is still deemed important and critical topic. It is still especially deemed so during the transfer process. In this research, a new approach is proposed for hiding information through the use of iterated function systems (i.e., IFS) from Fractals. This approach employs the main feature of fractals that concentrate on the idea that hackers who seek to find the hidden data shall not be able of locating it. Therefore, there is a need to carry out a decoding process in the aim of revering the conversion for securing the transmitted information. In this research, the secure information is hidden inside a fractal Mandelbrot image using the Linear Congruent Generator (i.e., LCG). Regarding the proposed system, it generates the fractal image through the use of the predefined knowledge gained from the hider site that works as a host for different types of secret messages. The knowledge that comes from the key of image dimensions, parameters of Mandelbrot, LCG key, and key agreement of cryptography method, which makes Stego-image analyses of hidden data unacceptable without the correct knowledge. Based on the results that are obtained through carrying out experiments showed the proposed method meets all the requirements for steganography. Such requirements include: the ones related to capacity, visual appearance, undetectability, robustness against extraction (i.e., security), and hit the highest

| | |
|---|---|
| **CC License**<br>CC-BY-NC-SA 4.0 | capacities with a visual appearance of high quality.<br>***Keywords: Fractal Image, Steganography, Mandelbrot, Linear Congruent Generator, Security*** |

## 1. Introduction

The internet is a public channel. It is used to transfer data from sender to receiver. Such data may include a violation of copyright. Thus, several techniques have been developed to address this problem. Steganography, watermarking, and cryptography are common methods that are used to maintain the security of data. They aim to meet the same goal, but they process and manipulate data in different ways. This article concentrates on Steganography which can be defined as an old and new technology for transferring data in a secure manner. Steganography could be introduced as a means for hiding data within a soft or hard host media. In the past, the sender employed animal leather, wood, and wax as a means to transfer the data in a secure manner to different geographical areas [1,2]. Today in the digital world, various types of files may be used as cover of secret data, like, audio, text, image, or video. The steganography methods could be classified as injection, substitution, and distortion or generation due to variations in the procedures carried out to do the embedding process [3].

The proposed work exposes a robust reliable fractal steganography system that is classified as a generation method. It is based on Mandelbrot as a fractal method for generating a bitmap image that serves as a host for the secret message, with pixel color codes, so each pixel has three colors (Red, Green, and Blue). Regarding the resulting image, it is divided into two areas: focus and non-focus areas. Here, the secret data shall be embedded in a random manner in a non-focus area. The colors of every pixel chosen from the non-focusing area shall be replaced with three bytes from the secret message. For granting every pixel the same probability for selection without biasing, the LCG shall be used.

The secret message is processed as binary data. Thus, any piece of data, like image, text, audio, or video may be handled by the proposed system. Through the embedding process, the system shall choose a set of arguments that shall be distributed later on between various units of the hidden stage. The arguments may be summarized by dimensions of Stego-image, Mandelbrot factors, LCG key, encryption key, and secret message. However, the process of extraction requires Mandelbrot factors, LCG key, decryption key, and length of secret message for retrieving the hidden message. The proposed method iterates the general formal definition of the Mandelbrot set for discovering the non-focus and focus locations [4], as you will see in the section of the Mandelbrot Computation Unit. The idea converges on hiding the secret data inside a non-focus area that will be built through the use of random colors.

## 2. Related Work

Regarding fractal research, it is a new field of interest. Nowadays, fractals may be decoded and generated with graphical representations. Fractal-image Compression (FIC) is a computational power that's needed for encoding and decoding them [5]. As the methods of detecting hidden data keep developing by methods of Mandelbrot Fractals, there is a need to develop more robust methods in this area. This part explains some of the works that belong to steganography by using fractals.

Al-Saidie et al. developed an approach that employs features of fractals for hiding secret data by employing an iterated function system (IFS) [5]. Regarding the converted message, it can't be understood; without the key of the method that was employed for recovering data, that by some knowledge of key agreement, with the original encryption. In addition, to improve the encoding process, the stenographic method was employed to hide the image of the attractor in another colored 256 x 256-pixel image.

Thamizhchelvy et al. proposed a method of hiding data using chaos theory [6]. It employs the dynamic system's initial state as a key produced by the pseudo-random number generator (PRNG). Regarding the fractals, they are generated by the fractal-image generation method. The data is hidden during the generation of the fractal. The created fractal images are watermarked when used for any online app as a digital signature.

Wu et al. developed a method to hide secret image information into fractal-images [7]. This method employs an arbitrary sequence that is generated by a chaotic map. It employs a wavelet converter for performing the hiding operation. The generated fractal cover image could be unique. The generated fractal cover image could be unique. The wavelet transform ensures the secret information is only embedded within the borders with the aim of preventing visual distortion.

Abbas developed an embedding approach for an image by identifying the features of the cover-image areas, by relying on the use of the cover-image fractal technique [8]. The method chooses regions that hide as much data in an image as possible without having its content sacrificed. It explained how the hidden data is robust in the case of image processing.

Gupta et al. proposed a method for the use of self-likeness for offering a picture for fractal regions to be decided [9]. To hide the information in that area, the location of the fractal region is presented in a picture. The algorithm proposed results in coded good images and tends to be just like a cover illustration. It assists the user in the process of covering information without identifying the picture that holds a secret message by the unauthorized person.

Desai et al. developed a technique that selects the fractal position of Mandelbrot on the input image. Regarding fractal, it is generated through a sequence of transformations from (0, 0) to ine segment (1, 0) [10].

Sun et al. developed a new 2D vector map hybrid data protection system [11]. The characteristics of the vector map are first separated into various separate classes to ensure that the localization of the tamper is identified accurately. The latter researchers developed a feature group correlation technique based on a vertex injection. They developed it to identify the batch deletion attack. The combination of polar coordinate transformation and the hash function is effective for avoiding the rotation, uniform scaling, and translation (RST) operations which produce a fragile watermark. The system employs an invariant RST watermarking approach for embedding the watermark. The proposed method has good invisibility and high specificity in the place of the ad.

Hosam developed a technique for hiding bitcoins in steganography fractals with reliable steganography [12]. The fractal tree is selected due to its basic recursive form. Through discretizing the angles and lengths of the tree branches, the private key bit stream shall be covered. Thus, the tree could be printed without being stolen or destroyed. The method shows a high level of safety and much strength in preventing attacks.

Sroor et al. found a range of cross-sectional fractal forms of lowest-losing amplitude [13]. Eigen modes show the directed production of fractal light inside a laser cavity with unstable canonical laser resonators. It displays the existing theory of fractal laser modes first, through the estimation of the three-dimensional, self-like, fractal structures in the middle of the magnified self-spousing plane and then, quantitatively, through the demonstration of cross sections of strength that are most self-like of the magnified self-spousing plane. The work reflects a major development in terms of understanding the basic existence symmetry that is observed in lasers.

Gao et al. suggested that a simplified Mandelbrot-Julia (M-J) set-based image encryption scheme [14]. Not only is the main space for the scheme big, but it is also a dynamic variable. The simplest two-round XOR encryption algorithm was employed to check the workability of the generalized M-J set in an encryption scheme. The main comes from background noise and SHA-512. Using once keys enhances the algorithm's stability and facilitates various hidden key transmissions. Based on the results of the simulation, the main space of the scheme is large, which is better than other approaches.

Bawaneh developed a random LSB image steganography system (LCG) by using a linear congruent generator (LCG) [3]. The system within the RGB color picture inserted the hidden message in random positions which LCG developed. The constructed system was robust against extraction and detection due to randomness distribution of data and complex secret keys. Four parameters were used for designing the key (Multiplier, Cycle length, Seed, and Non-common factor). Embedding channel selection depends on each change rate on the red, green, or blue channels. Based on the results, in terms of visual appearance and security random LSB was higher than sequential LSB.

In reference to Bawaneh et al. a steganography image scheme based on the principle of image segmentation has been designed based on a virtual grayscale [6]. The system divides the cover images into various segments in accordance with the image width and height. Segment selection is carried out by the user as a part of the key. The system employs one master key for random distribution, cryptography, and image segmentation. Based on the results, the Stego-image was robust against detection and analysis.

Bawaneh proposed a framework for image steganography through the use of simulated annealing (SA) [4]. It aims at hiding data inside an image by finding the minimum path that is between image pixels. The system was robust against data extraction due to the required knowledge for carrying out the processes of embedding and extraction.

Bawaneh designed an intelligent framework for hiding data in grayscale images through the use of an intelligent water drop (IWD) algorithm [15]. Based on the results, the framework meets the steganography requirements.

Masood et al. developed a system that is based on a method of shuffling with fractals [16]. This system is also based on a 3D chaotic Lorenz map.  Regarding the method of shuffling, it introduced the uncertainty property. It introduced regular picture pixels. Through using the three-dimensional Lorenz chaotic map, the diffusion process distorted all pixels of the image.

Xian et al. used the fractal sorting matrix with irregular, self-similar, and infinitely iterative for scrambling images or data on this new matrix [17]. The combination process will increase the protection level of the encryption algorithm as they mentioned.

Xuejiao et al. explored the impact of the production on the presence of the fractal image of every affine transformation [18]. The analysis of the kinds and the variation law of affine transformation led to having fractal images with various transformation characteristics. It was done to provide a more controllable fractal picture for fields of use, like, steganography of information.

Kasapbaşi developed a new spatial-domain chaotic steganography scheme that uses a fractional encryption method to hide compressed Huffman Turkish cipher texts [19]. They collected Turkish texts from a group of newspapers, and then the Mandelbrot set was used to create an encryption key. After that, the least significant bit method was used to find the locations that would be used in the steganography process. Finally, the pixel that would contain the hidden data was chosen through a certain association. The results indicate that the proposed schemes are successful for encryption and provide robust information hiding.

Mohammad et al. presented an alternative scheme for information hiding, in which the image that will contain the secure hidden data is created as an image of a curve resulting from a set of mathematical operations carried out on mathematical chaotic fractal groups [20]. This method aims to improve the concealment process by increasing the volume of hidden data and reducing the ability of attackers to retrieve hidden information. The scheme is based on the matching process between hidden information and values that are generated through the Mandelbrot-Julia method. The proposed method was successfully evaluated and tested with different data and from different viewpoints.

Mohit et al. developed a research for the purpose of improving the quality of the recovered watermarks, by using a method that combines image encryption and hiding audio information using direct sequence spread spectrum, in addition to random switching of audio encryption in order to produce a grayscale image as a medium for carrying the hidden data [21]. The noise ratio, mean square error (MSE), and other measures were used to examine the quality and performance of the proposed method.

## 3. Methodology

The necessity of an intelligent steganography technique utilizes the celebrity of Mandelbrot in constructing a secure and robust steganography system. It generates a new image (Stego Image) for hiding any kind of secret messages through the use of the fractal of Mandelbrot. The proposed method is based on the idea of forming a three-byte pixel (RGB) from secret message data and then using the resulting pixel in building the Stego image according to Mandelbrot computation for XY coordination. At the sender site (Hidden process), the employed data must be reordered and encrypted before having any image pixel generated. The receiver or extractor must be aware of message data and Mandelbrot parameters that were used in the hidden process. The system requires having a set of inputs that must exist at the initial state of execution. These requirements include secret message and Mandelbrot computation parameters. The encryption key and reordering of bytes shall be computed in accordance with the length of the secret message and Mandelbrot parameters. Even more, any type of material (binary file), such as text, image, sound, post-script, HTML and so on shall become a secret message for embedding. After taking the input data, they shall be checked against system conditions. The length of secret message and capacity of cover image that will be constructed in accordance with the Mandelbrot parameters are compared to check the size compatibility. If the length of the message is less than or equal cover image capacity, the system shall continue to the next conditions. Otherwise, it shall have the whole process is terminated. Later, the secret message and fractal parameters will be sent to their units to prepare for the next stage. Finally, the encryption key will be sent to the encryption unit. After that, the processing shall be transferred to distinct units to meet the main goal of the system. To meet the required target, ulterior subsections illustrate the way in which the hiding and extraction process shall be carried out.

### *3.1 Mandelbort Computation Unit*

Now, the role shall be transferred to the Mandelbrot unit which builds the Stego image and finds out the possible locations to embed the secret message. The process of construction and computation passes into a set of stages. Firstly, the encoder must insert width, height, focus factor (XF), and loop termination factor (LTF) in order to have them used in the process of building the Stego image. Width and height shall determine the dimensions of the result Stego image, while XF and LTF define the focus of the Mandelbrot area. Figure 1 shows a Stego image with 400x400 in which the focus area changed according to values of XF and LTF. Regarding the value of XF, it has an inverse relationship with the size of the focus area. Therefore, the parameter value shall be reduced and the size will shall increase. As for the available locations for embedding, they shall decrease.

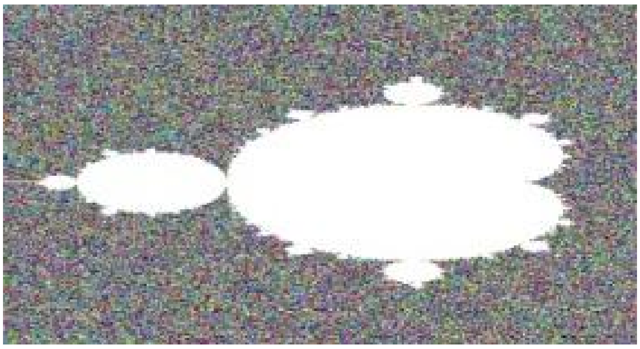

*Figure 1. Fractal Mandelbrot Image with size 400 x 400*

However, the LTF parameter controls the shape of the focus area, so the high value of LTF shall grant more accuracy to shape and consume more time for execution. Procedure1 below shows how the Stego image will be constructed. The process of construction based on a set of parallel and sequential steps, which are partially automated.

**Procedure .1: Steps of constructing the Stego-image**

```
Input width, height, X-Factor, LTF
Bitmap Stego-Image = new Bitmap (width, height)
Compute XInc , YInc
Index = 0
MaxIteration = 100
For X in range (0 , Width)
    CX = (XInc * X) – XF
    For Y in range (0 , Height)
        ZX = ZY= 0
        CY = (YInc * Y) - XF
        LoopCounter = 0;
        Radius = $ZX^2 + ZY^2$
        While (Radius <= LTF and LoopCounter <
MaxIteration)
```

```
            Loop Counter= LoopCounter+1
            TempZX = ZX
            $ZX = ZX^2 + ZY^2 + CX$
            $ZY = (2 * TempZX * ZY) + CY$
            $Radius = ZX^2 + ZY^2$
        While End
      IF  Loop Counter != MaxIteration Then
        LS.AddNode(Index, X, Y)
        Index = Index + 1
        StegoImage SetPixel(X, Y, Color(Rnd, Rnd,
Rnd))
      Else
        StegoImage.SetPixel(X, Y, Color(255, 255,
255))
      End IF
    Next Y
  Next X
```

However, after having the required parameters inserted, the system incarcerates a frame buffer which is defined in terms of width and height parameters. The amount of jump inside the frame buffer in X and Y directions is defined by XInc and YInc. That can be seen through Equation 1 and 2.

$$\mathrm{XInc} = \frac{2 * \mathrm{X} - \mathrm{Factor}}{\mathrm{Width}} \quad (1)$$

$$\mathrm{YInc} = \frac{2 * \mathrm{X} - \mathrm{Factor}}{\mathrm{Height}} \quad (2)$$

Posteriorly, a linked list shall be defined to store available locations for secret data. Every location in the list shall include an index for node, x coordination, and y coordination. The index of a node is employed in the process of selecting a location randomly. That’s shown in the subsequent steps of the embedding process. Equation 3 presents the Mandelbrot set which shall be used in constructing focus and out area.

$$\begin{aligned} &CX = ((\mathrm{XInc} * \mathrm{X}) - \mathrm{XF} \;\; \mathrm{X} \text{ € } [0, Width) \\ &CY = ((\mathrm{YInc} * \mathrm{Y}) - \mathrm{XF} \;\; \mathrm{Y} \text{ € } [0, \mathrm{Height}) \\ &ZX = ZX^2 + ZX^2 + \mathrm{CX} \\ &ZY = 2 * ZY + \mathrm{CX} \\ &Radius = ZX^2 + ZY^2 \end{aligned} \quad (3)$$

Procedure 1 has a set of overlapping loops for accomplishing the task of location verification. In the the outer loop (X-Loop), it begins with index zero and stills working until reaching the value less than width by one. Within the X-Loop, it firstly computes the value of the complex parameter for X coordination (CX), then starts an inner loop for y coordination(Y-Loop) with the aim to determine the location of the pixel (X, Y). Inside the Y-Loop, the complex set for X and Y coordination is computed based on Eqs. 3. The complex set is kept going till the Radius value less than or equal LTF or reaches the greatest number of iterations. Based on the value of the *LoopCounter* parameter, the procedure shall choose the location of the pixel. If the value of the *LoopCounter* equals the *MaxIteration* parameter value, the location shall be set outside the focus and given the white color. Otherwise, the pixel shall be added to the list of possible locations and granted a random color. The random colors shall be employed for the non-focus area to improve the noise visual appearance that may result from embedding secret data in that area and reduce suspicion about the existence of hidden data. The result of this stage is a fractal image with no secret data and a list of locations that shall be passed to the embedding unit for utilizing them for selecting and embedding in the required locations.

### *3.2 Embeding Unit*

It looks to hide the secret data randomly in the non-focusing area of a fractal image. The processing starts by fetching the Mandelbrot constructed image and list of available locations from the Mandelbrot unit. After that, the number of secret message bytes shall be compared with the number of locations that are available in the retrieved list. Every pixel in a fractal image shall store three bytes from a secret message. Thus, the number of available locations must be one-third of the number of message bytes. In a true case of comparison, the system begins by reading three bytes from a secret, encrypting message bytes, choosing a random location from the list then embedding bytes within the image. The bytes of messages shall be encrypted through the use of a simple cryptograph algorithm named Caesar. To avoid the redundancy of locations from the list, a common random generator (named LCG) shall be used. It grants every location within the generator cycle the same chance to be selected if the generator preconditions are satisfied. Message bytes are employed for composing an RGB color which shall be employed in setting existing pixels in the fractal image. The result of this unit is a Stego image that has a comparable visual appearance to the fractal one. Procedure 2 presents the major steps of embedding secret data in the fractal image.

**Procedure.2: Embedding unit step**

```
Input LocationList, SecretMessage, FractalImage
Input LCGKey, EncryptionKey
Set Position = FilePosition (SecretMessage)
Set DataSize = FileSize (SecretMessage)
While (Position < DataSize)
    Data[0] = SecretMessage.ReadByte
    Data[1] = SecretMessage.ReadByte
    Data[2] = SecretMessage.ReadByte
    LCGKey= GetRandom (LocationList, LCGKey)
    Color C = Color (Data[0], Data[1], Data[2])
    FractalImage.SetPixel ( List.X, List.Y, C)
END
StegoImage = FractalImage
```

### *3.3 Linear Congruent Generator Unit*

It is a common unbiased random generator that grants every number in its cycle the same probability of being selected without redundancy. The returned number of the generators is located within the interval [0, M), such that the value of M represents the length of the generator cycle. LCG is employed as it’s seen in Equation 4.

$$X_{i+1}= (AX_i+C) \ Mod \ M \tag{4}$$

Where $X_i$ is the current random number, $X_{i+1}$ is the next random number, $A$ is the multiplier factor, $C$ is the non-common factor and is the cycle of the generator. Based on the formula, the seed value or X0 should be given by the user. LCG avoids data redundancy by satisfying a set of preconditions [22]. These conditions are:

- The value 1 is only the common factor between C and M.
- (A-1) must be a multiple of all prime numbers that divide M.
- If M is Multiple of 4 then (A-1) should be as well.

### *3.4 Caesar Cipher Unit*

Caesar cipher is a cryptography algorithm that is simple and reliable. It keeps the size of the encoding or decoding item as it is. The task of decryption and encryption is carried out through the use of two keys. One of them is a predefined list of special symbols that are used for substituting the letters in the encryption and decryption processes [23]. Regarding the other key, it defines the starting index inside the list of symbols. The keys (list and starting index) must be conjoint between the parties of the process of transmission.

### *3.5 Extraction Unit*

The focus of this unit is represented in retrieving the secret data from Stego-image and building their hosting file. To meet the goal, a set of steps should be accomplished. Firstly, the unit gets Stego image, XF and LTF from the interface then builds the list of possible locations that may store the secret bytes of the message that is hidden. The values of width and height as two parameters for the list construction procedure shall be chosen from the Stego image itself. Thus, any change to the dimensions of the Stego image results in losing the secret data. After having the list construction completed, the task of the extraction process starts by taking LCGKey, DecryptionKey, and message length. Based on the value of LCGKey a random location from the list shall be chosen. The color is split into green, red, and blue where each one represents a byte of the hidden message. After that, the extracted bytes from color are decrypted through the use of Caesar procedure and the outcome will be written into the secret message file. The extraction process will be carried out as shown in Procedure 3.

**Procedure.3: Extraction procedure steps**

```
1.  StegoImage, XF, LTF
2.  Build LocationList
3:  NodeCounter = LS.NodeCounter();
4:  Input MsgLen, LCGKey, DecryptionKey
5:  IF (NodeCounter<MsgLen / 3.0) Then
    Return with error.
    End IF
6:  Set RetrieveBytes = 0;
7:  While (true)Do
      LCGKey = GetRandom(LocationList, LCGKey);
      Color Data = StegoImage.GetPixel(List.X, List.Y)
      IF RetrieveBytes<MsgLen Then
        WriteByteToFile(Data.R)
         RetrieveBytes= RetrieveBytes+1
      End IF
```

```
  IF RetrieveBytes < MsgLen Then
     WriteByteToFile(Data.G)
     RetrieveBytes= RetrieveBytes+1
  End IF
  IF RetrieveBytes<MsgLen Then
    WriteByteToFile(Data.B)
     RetrieveBytes= RetrieveBytes+1
 End IF
IF RetrieveBytes>= MsgLen Then
    Break While
  End IF
End While
```

## 4. Results and Discussion

Regarding the proposed system, it was evaluated by means of several hidden messages that are embedded in a 400 X 500 fractal image. Table 1 displays the data sets that are used in the testing process; it consists of four secret messages of various sizes.

*Table 1. Secret Messages*

| **Message** | **Size** |
|---|---|
| M 1 | 77 Bytes |
| M 2 | 1.43 Kbytes |
| M 3 | 393 Bytes |
| M 4 | 1.15 Kbytes |

Every image was assessed at various values for *XF* and LTF. As mentioned previously, every pixel of the host image can store three bytes from the secret message, thus the number of available locations inside a fractal Mandelbrot image will based on image fetching size, *XF,* and LFT. The relationship that is between fractal image size and available locations is a positive one. Table 2 presents the available locations for a used image with *XF* values from 1.5 to 1.9 and LTF with values from 1 to 4. The number of available locations shall be increased when the value of *XF* increases and the inverse is true for LTF.

*Table 2. Available Locations for Used Image*

| **Width** | **Height** | **XF** | **LTF** | **Available Locations** |
|---|---|---|---|---|
| 400 | 500 | 1.5 | 1 | 169643 |
| 400 | 500 | 1.5 | 2 | 165680 |

| | | | | |
|---|---|---|---|---|
| 400 | 500 | 1.5 | 3 | 165656 |
| 400 | 500 | 1.5 | 4 | 165656 |
| 400 | 500 | 1.6 | 1 | 173339 |
| 400 | 500 | 1.6 | 2 | 169812 |
| 400 | 500 | 1.6 | 3 | 169780 |
| 400 | 500 | 1.6 | 4 | 169776 |
| 400 | 500 | 1.7 | 1 | 176363 |
| 400 | 500 | 1.7 | 2 | 173288 |
| 400 | 500 | 1.7 | 3 | 173238 |
| 400 | 500 | 1.7 | 4 | 173236 |
| 400 | 500 | 1.8 | 1 | 178885 |
| 400 | 500 | 1.8 | 2 | 176141 |
| 400 | 500 | 1.8 | 3 | 176096 |
| 400 | 500 | 1.8 | 4 | 176080 |
| 400 | 500 | 1.9 | 1 | 181082 |
| 400 | 500 | 1.9 | 2 | 178605 |
| 400 | 500 | 1.9 | 3 | 178557 |
| 400 | 500 | 1.9 | 4 | 178531 |

To assess the constructed steganography system, there is a need to take a variant measure into consideration. Peak Signal to Noise Ratio (PSNR), Mean square error (MSE), robustness, visual appearance, detection, capacity, and security are the common criteria that are used for checking the proposed system [4]. The MSE and PSNR are defined as quality parameters that carry out a comparison between the clear fractal image and the result Stego image. Regarding MSE and PSNR, they are defined respectively in Equation 5 and 6.

$$MSE = \frac{1}{w * H(\sum_{j=0}^{H-1} (Pixel_X(I,J) - Pixel_Y(I,J))^2} \quad (5)$$

$$PSNR = 20 * \log_{10} (\frac{255}{\sqrt{MSE}}) \quad (6)$$

The modification rate indie the Stego-image is measured through MSE. Thus, the minimum MSE is set as the better, and on other words the minimum MSE has a minimum noise in the Stego-image. PSNR set the highest value as the best one, due to inverse relationship between MSE and PSNR. The values of MSE and PSNR are displayed in the Table 3 that gives a disparity due to randomness process of constructing the non-focus color part and the random distribution of bytes inside the required area. The value of MSE is based on the data to be hidden. That means that different data formats can lead to having different MSEs. Since the system deals with data at the byte level, therefore, the similarity between the data to be hidden and the image that will be used as the hiding medium leads to reducing the error rate. For example, the error rate is small if image data is hidden within an image due to the similarity or closeness between the color values in the two images. The irregularity of the MSE curve with different messages that were embedded in a fractal image of size 400 by 500 is presented in Figure 2.

The constructed method uses some random locations inside non-focus area, thus replacing any pixel in worst case will increase the MSE. More ever the size of secret message influences the value of MSE, so the large secret messages will increase the value of MSE, also the *XF* value has a major effect on the value of MSE, but LTF has few impacts on value of MSE as shown in Figure 2. PSNR is deemed as a reflection to value of MSE; it grants an indication about the noise or the modification that is in the Stego image.

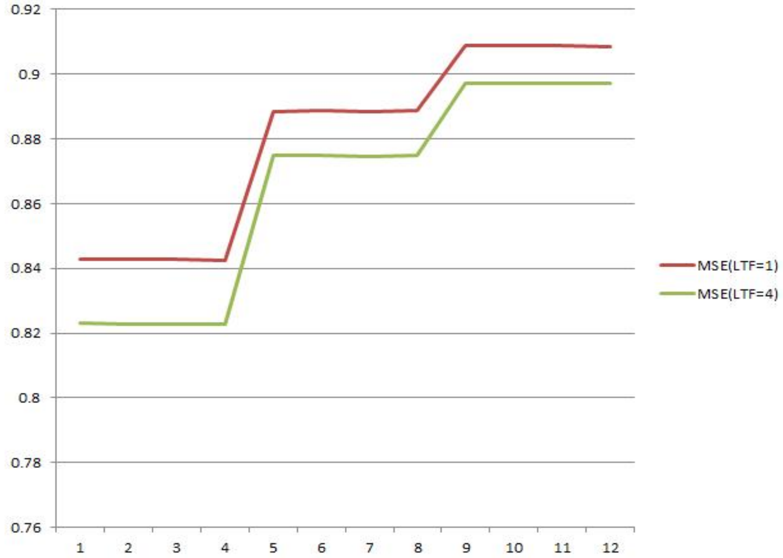

*Figure 2. MSE for Different Cases*

*Table 3. MSE and PSNR at Different Cases*

| Fractal Image | XF | LTF | Message | MSE | PSNR |
|---|---|---|---|---|---|
| 1 | 1.5 | 1 | M 1 | 0.842664 | 48.87426 |
| 2 | 1.5 | 1 | M 2 | 0.84293 | 48.872889 |
| 3 | 1.5 | 1 | M 3 | 0.84263 | 48.874434 |
| 4 | 1.5 | 1 | M 4 | 0.842577 | 48.874706 |
| 5 | 1.5 | 4 | M 1 | 0.823107 | 48.976239 |
| 6 | 1.5 | 4 | M 2 | 0.822746 | 48.978145 |
| 7 | 1.5 | 4 | M 3 | 0.822733 | 48.978216 |
| 8 | 1.5 | 4 | M 4 | 0.822882 | 48.97743 |
| 9 | 1.8 | 1 | M 1 | 0.888559 | 48.643939 |
| 10 | 1.8 | 1 | M 2 | 0.888638 | 48.643556 |
| 11 | 1.8 | 1 | M 3 | 0.888454 | 48.644455 |
| 12 | 1.8 | 1 | M 4 | 0.888632 | 48.643586 |
| 13 | 1.8 | 4 | M 1 | 0.874931 | 48.711068 |
| 14 | 1.8 | 4 | M 2 | 0.874794 | 48.711746 |
| 15 | 1.8 | 4 | M 3 | 0.87452 | 48.713106 |
| 16 | 1.8 | 4 | M 4 | 0.874851 | 48.711465 |
| 17 | 2 | 1 | M 1 | 0.908809 | 48.546075 |
| 18 | 2 | 1 | M 2 | 0.908792 | 48.54616 |
| 19 | 2 | 1 | M 3 | 0.908736 | 48.546426 |
| 20 | 2 | 1 | M 4 | 0.908489 | 48.547607 |
| 21 | 2 | 4 | M 1 | 0.89732 | 48.60133 |
| 22 | 2 | 4 | M 2 | 0.897188 | 48.60197 |

| | | | | | |
|---|---|---|---|---|---|
| 23 | 2 | 4 | M 3 | 0.897319 | 48.601336 |
| 24 | 2 | 4 | M 4 | 0.897262 | 48.601612 |

After studying most of the proposed systems that work on hiding data that were published previously [1], [2], [3], [15], it became clear that some of them work on hiding data in the least significant bits, either in a sequential or random way, while others work on dividing the image and then hiding secret message data in different parts depending on a previously defined function or using some artificial intelligence methods. The current system, which is being studied, hides three bytes in a location in the image, which is built using Mandelbrot equations, which makes it difficult to compare it with the methods that were previously published.

Regarding the result Stego-image, it has a comparable visual appearance to fractal Mandelbrot fractal image, due to random colors that were employed in building the non-focus area. Figure 3 shows on left side Mandelbrot fractal image with no secret data, while the same image on the right side involves the secret message M 2. The features of randomness in the constructing image and the distributing data inside image grant no evidence about the existence of hidden data. Thus, the constructed system passed in visual appearance evaluation.

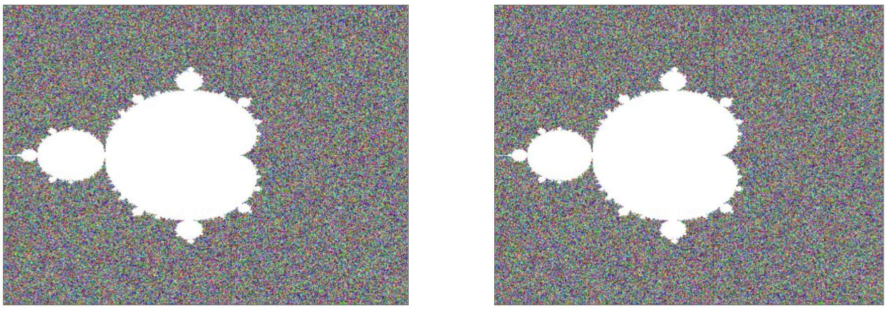

*Figure 3. Different Cases of Stego-image*

Regarding the maximum capacity of the cover image, it is calculated by several factors (dimension of image, *XF* and LFT), to evaluate this criterion a message of size of 318 bytes was embedded inside an image of dimensions 20 X 20. Figure 4 submits a strong evidence about the succession of proposed system in the measure of capacity, it shows the visual appearance of Stego-image on the right side. That is very similar to Mandelbrot image on left side.

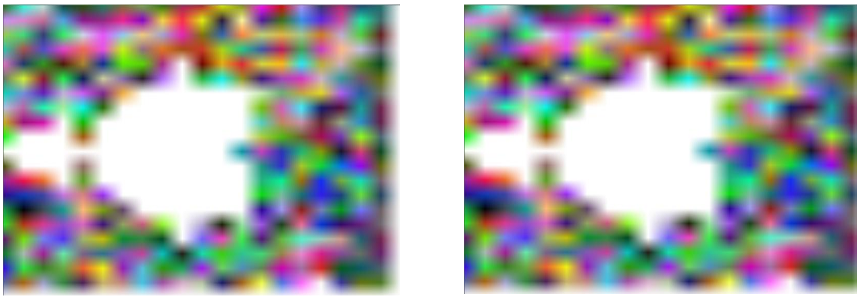

*Figure 4. Stego-image with Maximum Size Message*

Robust image steganography system encounters the modification and extraction of data inside the Stego image. However, the proposed method is robust against data extraction due to various keys that are employed in the processes of embedding and extraction of secret data, but not robust against the alterations in Stego-image size or format. Undetectability as a measure was applied in the constructed framework through having a randomness construction for the image, randomness distribution of data within the non-focusing area, encryption of data, and comparable visual appearance of the Stego image. Thus, the hidden bytes are deemed not clear for the analyzer or the detector.

To have the secret message recovered, the extractor needs to have a few keys. Such keys are represented in the: length of the secret message, extension of the secret message, decryption algorithm, decryption key, bytes distribution, Mandelbrot parameters, and LCG parameters. Thus, the extraction process requires full knowledge of the used keys in the embedding process. Thus, it's deemed as a secure one.

## 5. Conclusion and Future Works

MSE, PSNR, visual appearance, security, undetectability, capacity, and security as requirements for image steganography systems were satisfied as mentioned in the results and analysis. The idea of using a Mandelbrot fractal image in a non-focus area with LCG selection is considered an adaptive framework. The utilized method built a fractal image steganography system that is characterized by: robust against unauthorized data extraction, effectiveness in hiding data and comparable visual appearance. The main weakness of the proposed system and most of the image steganography one's is non-robust against image modification in terms of resizing or formatting changing.

By studying published research in the field of hiding data inside images, there is no way to limit the protection of hidden data if the size or settings of the image carrying the data are changed.

In the future, we look forward to applying this system using neural networks in order to improve performance and compare the current method with artificial intelligence methods.

## References

[1] M. J. Bawaneh, "A novel approach for image steganography using LCG," *International Journal of Computer Applications*, vol. 102, no. 10, pp. 34-38, 2014.

[2] M. J. Bawaneh, and A. A. Obeidat, "A secure robust gray scale image steganography using image segmentation," *Journal of Information Security*, vol. 7, no. 3, pp. 152-164, 2016.

[3] M. J. Bawaneh, "An Adaptive Virtual Gray Scale Image Steganography Using Simulated Annealing," *International Journal of Computer Science and Information Security*, vol. 14, no. 9, p. 612, 2016.

[4] D. Dewey, "Introduction to the Mandelbrot Set. A guide for people with little math experience," *Accessed online: http://www. olympus. net/personal/dewey/mandelbrot. Html,* 1996.

[5] N. M. Al-Saidie, and T. A. Kadhim, "Using fractals in information Hiding," *Engineering & Technology Journal*, vol. 27, no. 16, 2009.

[6] K. Thamizhchelvy, and G. Geetha, "Data hiding technique with fractal image generation method using chaos theory and watermarking," *Indian Journal of Science and Technology*, vol. 7, no. 9, pp. 1271-1278, 2014.

[7] Y. Wu, and J. P. Noonan, "Image steganography scheme using chaos and fractals with the wavelet transform," *International Journal of Innovation, Management and Technology*, vol. 3, no. 3, pp. 285-289, 2012.

[8] T. A. Abbas, and H. K. Hamza, "Steganography using fractal images technique," *IOSR Journal of Engineering (IOSRJEN)*, vol. 4, no. 2, pp. 52-61, 2014.

[9] R. Gupta, D. Mehrotra, R. K. Tyagi, and R. Kumar, "Digital image encoding scheme using fractal approach," In *2018 International Workshop on Advanced Image Technology (IWAIT)*, pp. 1-8, IEEE, 2018, January.

[10] H. V. Desai, and Desai, A. A. "Image steganography using mandelbrot fractal," *International Journal of Computer Science Engineering and Information Technology Research (IJCSEITR)*, vol. 4, no. 2, pp. 71-80, 2014.

[11] Q. Da, J. Sun, L. Zhang, L. Kou, W. Wang, Q. Han, and R. Zhou, "A novel hybrid information security scheme for 2D vector map," *Mobile Networks and Applications*, vol. 23, pp. 734-742, 2018.

[12] O. Hosam, "Hiding bitcoins in steganographic fractals," In *2018 IEEE International Symposium on Signal Processing and Information Technology (ISSPIT)*, pp. 512-519, IEEE, 2018, December.

[13] H. Sroor, D. Naidoo, S.W. Miller, J. Nelson, J. Courtial, and A. Forbes, "Fractal light from lasers," *Physical Review A*, vol. 99, no. 1, p. 013848, 2019.

[14]W. Gao, J. Sun, W. Qiao, and X. Zhang, "Digital image encryption scheme based on generalized Mandelbrot-Julia set," *Optik*, vol. 185, pp.917-929, 2019.

[15]M. J. Bawaneh, "A Preferential Virtual Gray Scale Image Steganography Using Intelligent Water Drop," *International Journal of Computer Science and Information Security*, vol. 14, no. 11, p. 538, 2016.

[16]F. Masood, J. Ahmad, S.A. Shah, S.S. Jamal, and I. Hussain, "A novel hybrid secure image encryption based on julia set of fractals and 3D Lorenz chaotic map," *Entropy*, vol. 22, no. 3, p. 274, 2020.

[17]Y. Xian, and X. Wang, "Fractal sorting matrix and its application on chaotic image encryption," *Information Sciences*, vol. 547, pp. 1154-1169, 2021.

[18]X. Tao, S. Bai, C. Liu, H. Chen, and Y. Yan, Algorithm of Controllable Fractal Image Based on IFS Code. In *2021 IEEE International Conference on Power Electronics, Computer Applications (ICPECA)*, pp. 801-809, IEEE, 2021, January.

[19]M. C. Kasapbaşi, "A new chaotic image steganography technique based on Huffman compression of Turkish texts and fractal encryption with post-quantum security," *IEEE Access*, vol. 7, pp. 148495-148510, 2019.

[20]M. A. Alia, and K. Suwais, "Improved steganography scheme based on fractal set," *The International Arab Journal of Information Technology*, vol. 17, no. 1, pp. 128-136, 2020.

[21]M. Bansal, and R. Ratan, "Designing a Novel Technique for Multi-Level Security System for Digital Data by Combined DSSS Audio Steganography & Random Permutation Cryptography," *Tuijin Jishu/Journal of Propulsion Technology*, vol. 44, no. 4, pp. 768-781, 2023.

[22]B. J. Morgan, *Elements of simulation*. *Routledge*, 2018.

[23]W. Stallings, "Cryptography and Network Security Principles and Practices," Publisher: Prentice Hall. *November*, vol. 16, pp. 463-486, 2005.