The ISC Int'l Journal of Information Security
ISeCure

July 2022, Volume 14, Number 2 (pp. 167–179)
http://www.isecure-journal.org

# Secure and Imperceptible Image Steganography in Discrete Wavelet Transform Using the XOR Logical Function and Genetic Algorithm

Vajiheh Sabeti [1,*], and Mahsa Amerehei [1]
[1] *Department of Computer Engineering, Faculty of Engineering, Alzahra University, Tehran, Iran.*

ARTICLE INFO.

*Article history:*
**Received:** February 20, 2021
**Revised:** June 25, 2021
**Accepted:** February 13, 2022
**Published Online:** February 20, 2022

*Keywords:*
Digital Steganography, Steganalysis, Discrete Wavelet Transform, Genetic Algorithm

**Type:** Research Article

**doi:** 10.22042/ISECURE.2022.274305.641

**dor:** 20.1001.1.20082045.2022.14.2.5.7

ABSTRACT

A steganography system must embed the message in an unseen and unrecognizable manner in the cover signal. Embedding information in transform coefficients, especially Discrete Wavelet Transform, is one of the most successful approaches in this field. The proposed method in this paper has two main steps. In the first step, the XOR logical function was used to embed two bits of data in the adjacent DWT coefficient pair. No change in the coefficients will occur if the XOR result of the two bits of low-value data of the two adjacent coefficients is identical to the two bits of secret data. Otherwise, one or both of the coefficient(s) will need a one-unit increase or decrease. In the second step, the genetic algorithm was used to select, between the two possible solutions, a new value for the adjacent coefficient pair that needs to be changed. Using the genetic algorithm, the selections were made such that the generated stego image experienced the least change relative to the cover image. The results of comparing this method with the existing methods in low- and high-level embedding showed that the proposed method was successful in producing stego images with high-quality criteria. In addition, the SPAM steganalysis method did not show high accuracy in its detection. One of the benefits of the proposed method is the need for a short key to embed and extract the secret message. This issue increases the security and feasibility of the proposed method.



## 1 Introduction

Today, digital communication is expanding through different data infrastructures and formats, and therefore, the need for secure communication is felt more than ever. Cryptography and data hiding are two major categories in information security systems [1, 2]. Although techniques of both categories are used to hide the information, their algorithms and applications are different. Innovative data hiding techniques have already been suggested to hide information, while image steganography is one of the most interesting and important research areas in this field. The main concerns of steganography are to conceal the presence of communication and protect the secret data [3].

In image steganography, the secret message is embedded in the cover image and transmitted in such

* Corresponding author.
Email addresses: v.sabeti@alzahra.ac.ir, mahsa.amerehei@gmail.com
ISSN: 2008-2045 

a way that the information is undetectable. An image in which secret information is hidden is called a "cover" or "host" image. The secret message embedding should ensure no significant changes in the statistical properties of the cover image. The "stego" image is an image obtained by embedding a secret message into the cover image. The secret message may be plain text, cryptographic text, and images [4].

Three key parameters to consider in evaluating the performance of a steganography system are imperceptibility, capacity, and security. Imperceptibility means that the stego media and the cover media should not be perceptually different. The carrying capacity refers to the concept of the maximum number of bits that can be hidden in the cover media. Security or undetectability also means that existing steganalysis attacks are unable to detect the stego images generated by the system. Notably, it is not possible to simultaneously maximize all three parameters, and there is always a compromise between them [1, 5].

Various methods in different categories have already been suggested for image steganography. In the most common categorization, the existing steganography methods are generally divided into two categories; spatial domain methods and transform domain methods [6]. In spatial-based methods, the secret message is embedded directly into the pixels of the image without any changes or transformations in the image before embedding. These methods have a high embedding capacity but a low resistance to attacks. One of the simplest steganography techniques in the spatial domain is to embed in low-value bits that embed the secret message bits directly into the least significant bit (LSB) of pixels of the cover image. This method is easy and simple, but easy to detect, as well. However, the human visual system may not be able to identify the presence of secrets in the cover medium, and hence, the spatial domain steganography provides stronger imperceptibility [7].

Steganography in the transform domain, first by applying a conversion, converts the image into a transform domain and then embeds the secret message into the transform coefficients. Eventually, the reverse conversion is taken from the converted image containing the secret message, and the stego image is obtained. Transform domain methods have lower embedding capacity than spatial domain methods, but as they propagate changes across the signal range, they are more resistant to different attacks [8]. Among the transformations, Discrete Cosine Transform (DCT) and Discrete Wavelet Transform (DWT) are more popular. Since most images on the Internet are in JPEG format, and this format is DCT-based, there is a great deal of interest in hiding information in the DCT domain. The DWT domain is also of interest in two respects: first, it is more compatible with the human visual system, and second, the JPEG 2000 image format is based on the wavelet transform. As the Fourier transform is less commonly used to convert images, the number of methods employed in this field is limited [9].

Improving image quality, embedding capacity and security of steganography methods is the main goal of researchers in developing new methods in this field. However, given the contradiction between these three criteria, it is almost impossible to achieve a method that, in addition to having a high capacity for embedding and maintaining the quality of the stego image, is less likely to be detected by attacks. Because of the greater embedding, the statistical impact of the embedding process increases, and as a result, steganalysis methods are more likely to be detected. Therefore, one must look for a method that, in addition to having the proper capacity and quality of a stego image, provides desirable security against existing attacks. Accordingly, the main purpose of the current article is to introduce a method with these features.

The algorithm presented in this paper is based on DWT and embeds and extracts secret data in this domain. In this approach, the image is divided into $8\times8$ non-overlapping blocks. Then, a two-dimensional $2-D$ DWT is applied to each block. This method uses $2-D$ Haar-DWT. If the XOR result of the two LSBs of the two adjacent coefficients is identical to the two-bit secret data, no change in the coefficients will occur. Otherwise, one of the coefficients, or both, will need to be increased or decreased by one unit to generate the desired XOR result. For each pair of adjacent coefficients that need to be changed, there are two alternative values, and the genetic algorithm is used to select between them. Using a genetic algorithm will help select a state that results in a stego image with the least variation over the cover image. This could hopefully improve the quality of the stego image and the security of the proposed method against steganalysis attacks.

In Section 2, a number of related steganography techniques implemented in the DWT domain will be reviewed. In Section 3, the proposed method including embedding algorithm and extraction algorithm will be described. Section 4 will present the output of the evaluation of the proposed method and compare it with some of the existing approaches, and finally, Section 5 will provide a conclusion.

## 2 Review of Related Works

Haar-DWT is the simplest type of wavelet transform. In this transform, low-frequency wavelet coefficients

are generated by averaging two adjacent pixels, while high-frequency coefficients are assumed as half of the difference between two adjacent pixels. The four obtained subbands are approximation (LL) subband, vertical (LH), horizontal (HL), and diagonal (HH) detail subbands. The approximation subband contains low-frequency wavelet coefficients that form a significant portion of the spatial domain image. Other subbands, called detail subbands, include high-frequency coefficients that make up the details of the edge of spatial domain image.

Haar-DWT operates first on adjacent horizontal elements and then on adjacent vertical elements. A desirable feature of the Haar wavelet transform is that the conversion is equal to its inverse. Figure 1 shows four subbands of the Lena image after the Haar wavelet transform [10].

**Figure 1**. Lena image after Haar wavelet transform

The wavelet transform has certain features that make it special in signal processing:

1. Spatial-Frequency Concentration: This property enables the detection of image properties, including edges and textures, which are located at different locations in the image and appear as larger coefficients in subbands [11, 12].
2. Multi-Resolution Display: By applying a wavelet transform to an image, four subbands are produced, one of which is an approximation of the image, and the other three subbands complete the image details. At each of these levels, the image properties can be distinguished in different locations and directions [13].
3. Human Visual System: Multi-resolution representation of the wavelet transform works following the eye-vision model in processing the received images [13].
4. Complexity: The wavelet transform has the order of complexity of $O(n)$, which is linear, while the complexity of the DCT transform is of the order $O(nlog(n))$ [9].

DCT is a widely used transform domain technique among the first generation transform-based embedding systems and was later replaced by DWT due to its better embedding capacity and imperceptibility [14]. In addition to DWT, there are steganography methods that used other types of wavelet tranform, such as Integer Wavelet Transform (IWT), Complex Wavelet Transform (CWT), or Dual-Tree Complex Wavelet Transform (DT-CWT). Following is a review of the works performed around the wavelet transform domain.

Genetic algorithm-based steganography methods using DWT and DCT are presented in [15]. In these methods, the genetic algorithm was used to generate a number of stego images based on the fitness functions. One of these images, which was more successful in terms of statistical criteria, was selected as the best stego image. It was observed that a method using DWT was more successful in bit error rate (BER), PSNR, and embedding capacity than the method using DCT. In DWT based method, the secret image is converted to the wavelet domain and according to the fitness function; different bands of the secret image are selected and embedded in the cover image. This process continues until the best stego image is obtained. This stego image is then given to a statistical properties test block. If the test results meet the target criteria, that image will be an optimal stego image. Extraction of the stego image is conducted using the stego key. The secret image contained in the stego image can only be extracted if the stego key is recognized.

In [16], Ghasemi *et al.* used wavelet transform and genetic algorithm in a steganography scheme. A genetic-based mapping function is used to embed data in DWT coefficients in $4 \times 4$ blocks onto the cover image. The Optimized Pixel Adjustment Process (OPAP) is applied after embedding the message. The frequency domain has been used to improve steganography robustness and the genetic algorithm and OPAP have been used to obtain an optimal mapping function to reduce the error difference between the cover and the stego image. As a result, the hiding capacity improves with less damage. The genetic algorithm provides solutions based on the order of embedding a secret message in pixels of each block and evaluates each solution using PSNR criterion and finally provides the best embedding order for each block. This method has been very successful in im-

proving the quality of the stego image, but it also has a major drawback. This method requires the sender to transmit the order of embedding specified by the genetic algorithm as a key to the receiver, which is practically impossible. This issue will be discussed further in Section 4.

In [17], a new method was presented that used the idea of Pixel Value Differencing (PVD) to embed in IWT coefficients. In this method, the cover image is transformed using IWT to obtain all four $LL$, $LH$, $HL$, and $HH$ subbands. Then, the PVD approach is used to hide the secret information in the wavelet coefficients of these subbands. The proposed method first modifies the difference between two wavelet coefficients of a pair and then uses the modified difference to hide the information. The results showed that this technique outperformed other PVD-based techniques in terms of hiding capacity.

In [18], two similar methods based on spatial and frequency domains were proposed. In these methods, frequency coefficients were divided into $3\times3$ windows. The corner coefficients showed the edge strength of the whole window and could determine the number of secret bits that could be embedded in the other coefficients of the window. In this way, edge identifier coefficients were separated from data carrier coefficients, and the receiver could extract data without any error. However, in every window, four coefficients out of nine coefficients did not carry the data and were only used for edge detection. Therefore, the capacity obtained in this manner will decrease and if the length of the data is high, this method will have lower quality than it could have.

In [19], a novel approach for data hiding in the frequency domain was proposed using a genetic algorithm. Unlike other papers which selected regular frequency domains like IWT in the frequency domain selection phase, in this paper, the frequency space was not constant and was chosen according to the secret information and cover images. Cover images were mapped to a proper frequency domain using the concepts of adaptive wavelet transform and genetic algorithm. In the obtained space, encrypted information had been embedded in the frequency coefficients that represent edges of the image in the spatial domain. Hence, the cover image will change the least and have the most compatibility with the human visual system. Simulation results showed that this method outperformed previous works in the PSPNR criterion.

In [20], a new method was proposed with the idea of more embedding in edge coefficients. In this method, after applying IWT conversion on the cover image, the MSB part of the conversion coefficients was used for decision-making on the embedding process. The embedding was done in such a way that there was no change in the MSB part of the coefficients and by receiving this part of the coefficients, the receiver would be able to be aware of the sender's decisions and extract the embedded data. In this method, the coefficients were classified according to their size, and each group carried a certain amount of hidden data based on the group label. Although this method is not significantly superior to the compared methods in terms of PSNR criteria, it has a higher PSPNR.

In [21], to construct the technique, Particle Swarm Optimization (PSO), Bi-Orthogonal Wavelet Transform (BWT), and genetic algorithm were combined. PSO was used to take the enhanced version of the host image. The enhanced version of the host images was sharper and brighter. The aim of BWT was to choose the selective subbands of the host image. A genetic algorithm was included to select the fittest hidden image among a set of hidden images that were created after mutation. Later, the hidden image would produce a confidential password using an innovative technique. This combinational approach of image steganography was quite safe for confidential data transmission and difficult for attackers to retrieve the confidential data.

In [22], a novel IWT- and DWT-based steganography method was presented that used PSO to find the optimal substitution matrix for converting secret data into their substituted forms. In this method, an OPAP was used to improve perceptual transparency to obtain a stego image with low distortion. This method improved the security, imperceptibility, and robustness of the secret data by hiding them into the wavelet coefficients of an image. Evaluation of existing methods shows that one of the successful ideas for increasing the security of steganography algorithms in both spatial and transform domains is the use of optimization algorithms in the data embedding stage. Application of optimization algorithms to steganography methods is usually performed in three stages; before embedding, during embedding, and after embedding. In the pre-embedding category, the optimization algorithm is usually used to find the best place to embed or modify message bits [23–26]. In the second category, the optimization algorithm is used to determine the value of the stego image pixel and how data is stored [27, 28], and in the third category, the optimization algorithm helps to reduce the changes resulting from the embedding [29, 30]. The most common and successful optimization algorithm used in the literature is the genetic algorithm. There is limited use of other optimization algorithms, such as PSO [31], Ant Colony [32], and Artificial Bee Colony [33]. Consequently, in this paper, the genetic algorithm was used in the proposed method.

## 3 Proposed Method

This paper introduces a new algorithm for image steganography based on DWT that embeds and extracts message data in this domain. In the proposed method, a genetic algorithm is used as an optimization algorithm. Due to the ability of optimization algorithms in finding near-optimal points in different problems, using optimization algorithms in steganography methods can provide an appropriate fitness function and help improve the steganography algorithm.

The general algorithm of the proposed method is illustrated in Figure 2. Noteworthy, each steganography method has one or more keys, and the recipient must be aware of their values. These keys are used in the embedding process to increase security. Importantly, the keys are not long and can be exchanged via a secure channel between the transmitter and receiver, which is not possible with long keys. In the proposed method, there are two keys. The first key, $key_1$, is used as the block size, and the second key, $key_2$, as the kernel of the pseudo-random function of the embedding order selection.

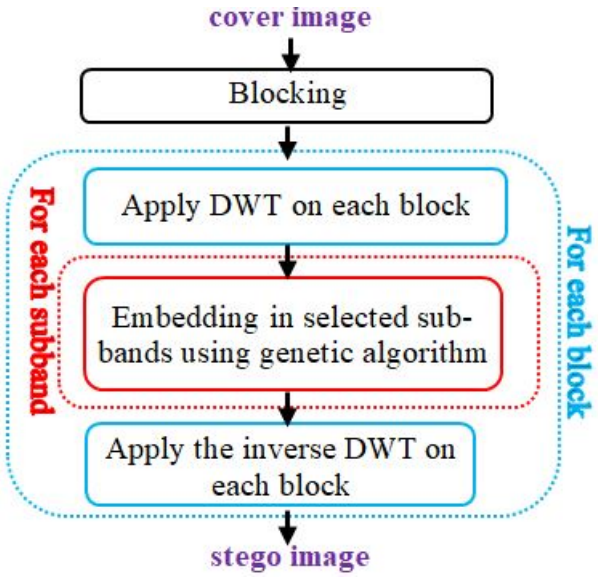



**Figure 2**. The general algorithm of the proposed method

The following are four steps of the embedding process in the proposed algorithm.
Step 1. Split the cover image into non-overlapping blocks (with $key_1$).
Step 2. Apply a $2-D$ DWT on each block.
Step 3. Embed the secret message into any selected subband in all image blocks (with $key_2$).
Step 4. Apply the inverse DWT to each block and obtain the final stego image.
In this algorithm, the first step is repeated once, the second and fourth steps are repeated the times equal to the number of image blocks, and the third step is repeated the times equal to the number of blocks multiplied by the number of selected subbands. Since the message embedding process in the LL approximation subband significantly affects image quality and drastically reduces image quality, this subband is not suitable for embedding. In contrast, embedding in the $HH$ subband causes the least damage to the image quality. The proposed method considers two modes: i) If low embedding capacity is needed, only the $HH$ subband will be used to embed the message, and ii) If the target data exceeds the capacity of this subband, all three subbands, $HH$, $LH$, and $HL$, will be used for message embedding.

Once the proposed algorithm is specified, the only obscure issue will be how to perform the third step of the algorithm. At this point, instead of directly embedding the data bits in the LSB of coefficients, an algorithm based on the XOR logical function is proposed. The steps are repeated in each subband of each block. For this, the coefficients in a subband are first considered as pairs. Each pair contains two neighboring coefficients. In each coefficient pair, two data bits are embedded. For embedding, two LSBs of each coefficient within a pair are considered. Suppose the coefficient pair $P$ in the cover image contains two coefficients, $C_1$ and $C_2$. $x_1x_0$ represents two LSBs of the coefficient $C_1$, $y_1y_0$ represents two LSBs of the coefficient $C_2$, and the sender decides to embed two data bits, $d_1d_0$, into the pair $P$. The sender changes these coefficients in the stego image to $C_1'$ and $C_2'$. $x_1'x_0'$, are two LSBs of the coefficient $C_1'$ and $y_1'y_0'$, are two LSBs of the coefficient $C_2'$. The sender makes this change so that the receiver can obtain the desired data from the XOR coefficients in the stego image. In other words, Formula (1) should be established after embedding the data into the stego image:

$$d_1d_0 = x_1'x_0' \oplus y_1'y_0' \tag{1}$$

To achieve this, it is clear that if these conditions are met for the primary $C_1$ and $C_2$ coefficients in the cover image, the same coefficients are placed on the stego image without any change. Therefore, if the conditions of Formula (2) are met, then: $C_1' = C_1$ and $C_2' = C_2$.

$$d_1d_0 = x_1x_0 \oplus y_1y_0 \tag{2}$$

However, if these conditions are not met, the sender must create these conditions by changing the values of $C_1$ and $C_2$ coefficients. Due to the sensitivity of the steganalysis methods to the changes made in the image for data embedding, in order to increase the security of the steganography method, we should make the least changes in the values of coefficients. It will be shown below that these conditions are caused

by increasing or decreasing one or both of the desired coefficients (depending on the case) by one unit.

The two LSBs of each coefficient have four different states: 00, 01, 10, and 11. Increasing or decreasing each coefficient by one unit, changes the value of these two bits. Figure 3 illustrates how this change occurs. For example, if two LSBs of a coefficient have the value of 00, by adding one unit to this coefficient, the value of two bits will become 01, and if one unit decreases, the value of two bits will become 11.

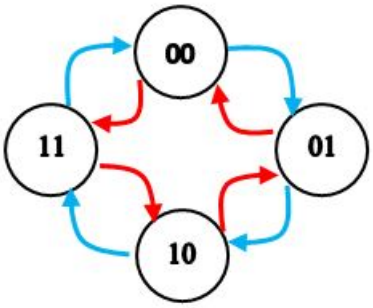



**Figure 3**. Change of the two LSB coefficients in the case of a single increase (blue path) or decrease (red path)

The decision on how to change the two coefficients depends on the original value of the two LSBs of each coefficient and the data to be embedded. Therefore, the values of $x_1x_0$, $y_1y_0$, and $d_1d_0$ are important in the decision-making process. The combination of the values $x_1x_0$, $y_1y_0$, has 16 different states. However, given the unimportance of the order of coefficients and the elimination of duplicate states, 10 main states remain for decision-making. There are three options for each coefficient: unchanged, one-unit increase, and one-unit decrease. So, there are nine options in each case for a pair. For better understanding, these nine options are presented in Table 1 for the coefficient pairs with initial values of 10 and 6. Operations 1 and 2 are the decisions made for the first and second coefficients, respectively. "No change" is shown by $\times$, "one-unit increase" by $++$, and "one-unit decrease" by $--$. After computing the new values of the coefficients and their two LSBs after the operation, the extracted data is shown in the last column. If the data to be sent is 00, the transmitter has three solutions, the least costly solution is not to change the coefficients. If the target data is 01, the transmitter has two solutions: keeping the first coefficient unchanged and adding one unit to the second coefficient, or adding one unit to the first coefficient and keeping the second coefficient unchanged. In both cases, the receiver can extract the 01 data from the new coefficients. If the target data is 10, the transmitter still has two choices: increase the first coefficient and decrease the second coefficient by one unit, or vice versa. The transmitter also has two options for sending data 11: keeping the first coefficient unchanged and subtracting one unit from the second coefficient, or subtracting one unit from the first coefficient and keeping the second coefficient unchanged.

To fully review the situation, the correct decisions for different states are provided in Table 2 according to the value of d1d0. In the proposed algorithm, the transmitter has two choices for embedding data in each state (Table 2). For example, if the first two low bits of the coefficients are 00 and 00, the possible solutions for sending different data are as follows:
Data 00: No need to change (marked with $\times$)
Data 01: Increasing the first coefficient or the second coefficient
Data 10: Decreasing the first coefficient and increasing the second coefficient, or increasing the first coefficient and decreasing the second coefficient
Data 11: Decreasing the first coefficient or the second coefficient

Although the embedding method is specified, the transmitter faces two solutions if the coefficients need to be changed. Which solution should he/she choose to make the change? If the solutions presented in Table 2 are examined, it will be observed that the magnitude of changes in both solutions is equal in all cases. In other words, since steganography methods try to minimize the changes in the image, one solution is superior to the other if it has fewer changes, but in all cases in Table 2, both solutions make either one change (increasing or decreasing one of the coefficients) or two changes (increasing or decreasing both coefficients). Hence, one of the possible strategies for selection can be a random selection. The advantage of using random selection is simple and fast. Although in single comparison mode it may not be possible to identify a particular superiority for a solution, it is possible to use an optimization algorithm to find a set of choices that together create better conditions for the stego image.

Given all the possible combinations of possible solutions for the coefficients of a subband, testing them all and selecting the best one is not possible. For this reason, using an optimization algorithm, one can hope for achieving a near-optimal combination. According to the description given in the previous section, the proposed algorithm uses the genetic algorithm for this purpose and it is necessary to note that the use of other optimization algorithms can also be considered. In the genetic algorithm, a solution with an answer vector is known as an individual or a "chromosome", and each chromosome is made up of separate segments called "gene". The definition

**Table 1**. All Choices for the coefficient pairs with initial values of 10 and 6

| $Operation1$ | $Operation2$ | $C_1'$ | $C_2'$ | $x_1'x_0'$ | $y_1'y_0'$ | $d_1d_0$ |
|---|---|---|---|---|---|---|
| × | × | 10 | 6 | 10 | 10 | 00 |
| × | ++ | 10 | 7 | 10 | 11 | 01 |
| × | −− | 10 | 5 | 10 | 01 | 11 |
| ++ | × | 11 | 6 | 11 | 10 | 01 |
| −− | × | 9 | 6 | 01 | 10 | 11 |
| ++ | ++ | 11 | 7 | 11 | 11 | 00 |
| ++ | −− | 11 | 5 | 11 | 01 | 10 |
| −− | ++ | 9 | 7 | 01 | 11 | 10 |
| −− | −− | 9 | 5 | 01 | 01 | 00 |

**Table 2**. Solutions for different data according to the current status of the coefficients

| $x_1x_0$ | $y_1y_0$ | $d_1d_0=00$ | | $d_1d_0=01$ | | $d_1d_0=10$ | | $d_1d_0=11$ | |
|---|---|---|---|---|---|---|---|---|---|
| | | Soultion1 | Solution2 | Soultion1 | Solution2 | Soultion1 | Solution2 | Soultion1 | Solution2 |
| 00 | 00 | × | × | $++C_1$ | $++C_2$ | $--C_1$<br>$++C_2$ | $++C_1$<br>$--C_2$ | $--C_1$ | $--C_2$ |
| 00 | 01 | $++C_1$ | $--C_2$ | × | × | $--C_1$ | $++C_2$ | $--C_1$<br>$--C_2$ | $++C_1$<br>$++C_2$ |
| 00 | 10 | $++C_1$<br>$--C_2$ | $--C_1$<br>$++C_2$ | $--C_1$ | $--C_2$ | × | × | $++C_1$ | $++C_2$ |
| 00 | 11 | $--C_1$ | $++C_2$ | $++C_1$<br>$++C_2$ | $--C_1$<br>$--C_2$ | $++C_1$ | $--C_2$ | × | × |
| 01 | 01 | × | × | $--C_1$ | $--C_2$ | $--C_1$<br>$++C_2$ | $++C_1$<br>$--C_2$ | $++C_1$ | $++C_2$ |
| 01 | 10 | $++C_1$ | $--C_2$ | $++C_1$<br>$++C_2$ | $--C_1$<br>$--C_2$ | $--C_1$ | $++C_2$ | × | × |
| 01 | 11 | $++C_1$<br>$--C_2$ | $--C_1$<br>$++C_2$ | $++C_2$ | $++C_1$ | × | × | $--C_1$ | $--C_2$ |
| 10 | 10 | × | × | $++C_1$ | $++C_2$ | $--C_1$<br>$++C_2$ | $++C_1$<br>$--C_2$ | $--C_1$ | $--C_2$ |
| 10 | 11 | $++C_1$ | $--C_2$ | × | × | $--C_1$ | $++C_2$ | $--C_1$<br>$--C_2$ | $++C_1$<br>$++C_2$ |
| 11 | 11 | × | × | $--C_1$ | $--C_2$ | $--C_1$<br>$++C_2$ | $++C_1$<br>$--C_2$ | $++C_1$ | $++C_2$ |

of possible values for the gene depends on the problem. The first steps in the genetic algorithm are to determine the structure of the chromosome and produce the initial population. In the proposed method, the genetic algorithm is repeated for each subband of each block. In each iteration, the chromosome length is equal to the number of subband pairs that need to be changed. The order of embedding in a subband pair is determined by the $key_2$. In order to generate the initial population, pairs are examined by order of embedding. If a pair does not need to be altered, based on the desired data, it is ignored, otherwise, a gene is added to the chromosome and randomly assigned 1 or 2 in the gene, where the value 1 represents the choice of solution 1, and 2 represents the choice of solution 2 to change the corresponding coefficient.

Each optimization algorithm seeks to minimize (or maximize) a fitness function (objective function) that depends on the problem. According to the articles reviewed in Section 2, there are various fitness functions in steganography methods that are applicable at this stage. One of the most popular functions is the PSNR function. This function is used as a criterion for evaluating the quality of the stego image. Increasing the PSNR value means increasing the quality of the stego image, that is, reducing the difference between the stego and cover image. This criterion can be calculated using Formula (3):

$$PSNR = 10\log_{10}\frac{M\times N\times 255^2}{\sum_{ij}(y_{ij}-x_{ij})^2} \tag{3}$$

Where $M$ and $N$ are the block sizes, and $x_{ij}$ and $y_{ij}$ are the pixel intensities in row $i$ and column $j$ before and after embedding, respectively.

The next step in the genetic algorithm is the formation of subsequent generations of populations based on the selection process, crossover, and mutation. In this implementation, the tournament selection method to select parents, as well as single-point crossover and permutation mutation (by randomly choosing between swap, reversion, and insertion methods) were used. New offspring generated by crossover and mutation operations are added to the previous population, and chromosomes with the highest amount of fitness function are considered as the next generation population, and the extra chromosomes are deleted. By repeating the algorithm for a certain number of times, the best chromosome is selected, which is the best solution or output of the genetic algorithm. The embeddingprocess is based on this solution, and finally, by applying a $2-D$ inverse DWT on all blocks, the stego image is obtained.

## 4 Evaluation of the Proposed Method

There are different parameters for comparing steganography methods, which can be divided into three main categories; stego image quality, embedding capacity, and attack resistance. The experiments were performed on a 2.3 GHz processor with a RAM of 8 GB and using MATLAB R2017b. Two sets of images were used for comparisons: 60 images of the SIPI dataset with the size of $512 \times 512$, and 400 randomly selected images from the INRIA Holidays Dataset with dimensions of $400 \times 400$. Comparisons were made at two embedding levels: embedding in all $HH$ subband coefficients (low capacity), and embedding in all three $HH$, $LH$, and $HL$ subband coefficients (high capacity).

To make a comparison, the steganography method presented in [9], which is a genetic algorithm-based method on the wavelet transform domain, has also been implemented separately and compared with the proposed method for the three parameters mentioned above. Although this method is relatively old, it has been successful in improving the quality of the stego image. Its major drawback is the need for a very long and irrational embedding key. This method finds the best embedding order through the genetic algorithm. This order must be communicated with the receiver as a very long-term key, which is impossible in practice. Therefore, the achievements of this method can be considered as an ideal point that has not been achieved in practice. In this section, the approach of the proposed method to these achievements is evaluated, except that the proposed method does not have a long-term key problem and is applicable in practice.

In the evaluation process, the following naming is used:

- MYXORGA ($n$): This name is chosen for the proposed method and n represents the number of subbands used to embed, for which two values of 1 ($HH$ subband only) and 3 ($HH$, $HL$, and $LH$ subbands) are used in the tests.
- MYXOR ($n$): This is the same method as the proposed one but the genetic algorithm step has been removed from it. Thus, in this way the choice between alternative solutions is random. By this approach, the impact of the genetic algorithm on the proposed method will be determined.
- OPAPGA ($k$): This name is selected for the method presented in [9] and $k$ represents the number of bits embedded in each coefficient. The tests used two values of 1 and 3.

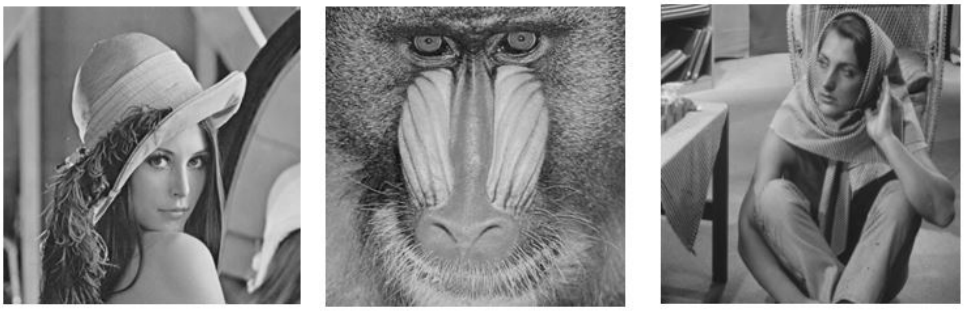
### 4.1 The Effect of Block Size on the Performance of the Proposed Method

One of the parameters of any method that affects its performance is the size of the selected blocks ($key_1$). In the first step, to determine the effect of block size on the performance of the proposed method, stego image quality criteria were measured for three test images including Lena, Baboon, and Barbara, with different block sizes. The test images are shown in Figure 4, and the test results for the execution of MYXORGA (1) are presented in Table 3. One of the criteria is the MSE, which represents the mean square error. Error is the difference between the cover and the stego image. This index has always a non-zero value, and the closer it is to zero, the lower is the error rate. The second criterion is the PSNR. Larger values of PSNR reflect better stego image quality. The SSIM shows the structural similarity of the stego image to the cover image. Structural information refers to the interdependence of pixels, especially in the case of pixels that are very close to each other. Ideally, this value is 1.

Examination of the presented results in Table 3 showed that, as expected, all criteria were improved with smaller blocks. In smaller blocks, due to the smaller search space for the genetic algorithm, the convergence speed of this algorithm was higher, and thus, better results were obtained. Therefore, in the next tests, the proposed method was performed with a block size of 8.

### 4.2 Quality of the Stego Image

There are several criteria to compare the quality of stego images. The MSE, PSNR, and SSIM criteria were explained in the previous subsection. The aver-



(a) (b) (c)

**Figure 4**. Three test images. (a) Lena, (b) Baboon, (c) Barbara

**Table 3**. PSNR, MSE, and SSIM results for test images with different block sizes

| | Lena | | | Baboon | | | Barbera | | |
|---|---|---|---|---|---|---|---|---|---|
| $key_1$ | PSNR | MSE | SSIM | PSNR | MSE | SSIM | PSNR | MSE | SSIM |
| 512 | 51.77 | 0.4326 | 0.9976 | 51.75 | 0.4338 | 0.9992 | 51.74 | 0.4347 | 0.9976 |
| 256 | 51.82 | 0.4275 | 0.9976 | 51.78 | 0.4306 | 0.9992 | 51.78 | 0.4312 | 0.9976 |
| 128 | 51.89 | 0.4200 | 0.9976 | 51.85 | 0.4238 | 0.9992 | 51.86 | 0.4231 | 0.9977 |
| 64 | 52.05 | 0.4051 | 0.9977 | 52.00 | 0.4095 | 0.9992 | 52.01 | 0.4091 | 0.9977 |
| 32 | 52.30 | 0.3820 | 0.9978 | 52.25 | 0.3868 | 0.9993 | 52.25 | 0.3872 | 0.9978 |
| 16 | 52.50 | 0.3651 | 0.9979 | 52.52 | 0.3634 | 0.9993 | 52.52 | 0.3633 | 0.9979 |
| 8 | 52.54 | 0.3620 | 0.9979 | 52.52 | 0.3634 | 0.9993 | 52.52 | 0.3633 | 0.9979 |

age values of MSE, PSNR, and SSIM for 60 images of the SIPI dataset are provided in Table 4 and Table 5. Examination of these results revealed three points:

1. Adding the genetic algorithm step to the proposed method had a great impact on improving the quality of the stego image.
2. As expected, the OPAPGA method produced higher-quality stego images than the proposed method because it was able to find the best embedding sequence. This superiority is higher in terms of PSNR and MSE criteria, and lower in the capacity embedding. In high-capacity embedding, the performance of the two methods is very close, but due to the impracticability of this method, the proposed method was able to produce high-quality images close to the quality of the images produced by the OPAPGA method and had no problem in practice.
3. At high capacity, the stego image quality in the proposed method using all three subbands was better than the OPAPGA method with more than one bit embedded in each coefficient.

In the next step, the goal is to compare the proposed method with a number of new methods. To perform this test at a low embedding level, methods proposed in [18], [19], and [20] were selected. Since methods in

**Table 4**. MSE, PSNR, and SSIM results for SIPI images (low capacity)

| | PSNR | MSE | SSIM |
|---|---|---|---|
| MYXOR(1) | 51.12 | 0.5023 | 0.9990 |
| MYXORGA(1) | 52.53 | 0.3620 | 0.9996 |
| OPAPGA(1) | 53.98 | 0.2589 | 0.9999 |

**Table 5**. MSE, PSNR, and SSIM results for SIPI images (high capacity)

| | PSNR | MSE | SSIM |
|---|---|---|---|
| MYXORGA(3) | 51.00 | 0.5203 | 0.9992 |
| OPAPGA(3) | 44.96 | 2.0717 | 0.9963 |
| OPAPGA(1) | 51.14 | 0.4993 | 0.9993 |

[19] and [20] claim their superiority in the PSPNR criterion, this criterion was also used to compare the proposed method with these methods. It is equal if the carrier coefficients are edge points or not, as PSNR criteria are pixel-based. Consequently, a parameter that considers the adjacent pixel to the intended one is introduced in [34]. In case the intended pixel is on the edge of the image, it can be changed to somewhat (jnd), and the remaining changes are taken into account in the calculation of the varia-

tion. Permissible range of change for each pixel can be determined according to the pixel value and its surrounding variance. This factor is referred to as PSPNR and obtained via Formula (4), as follows:

$$PSPNR = 10\log_{10}\frac{255^2}{MSEP} \tag{4}$$

$$MSEP = \frac{1}{MN}\sum_{i=1}^{M}\sum_{j=1}^{N}(|x_{ij}-y_{ij}|-jnd(i,j))^2 \times \delta(i,j) \tag{5}$$

$$\delta(i,j) = \begin{cases} 1 & \text{if } |x_{ij}-y_{ij}| > jnd(i,j) \\ 0 & \text{if } |x_{ij}-y_{ij}| \le jnd(i,j) \end{cases} \tag{6}$$

$$jnd(i,j) = \frac{\delta^2(local)}{5+x_{ij}} \tag{7}$$

Where $M$ and $N$ are the block sizes, and $x_{ij}$ and $y_{ij}$ are the pixel intensities in row $i$ and column $j$ of cover and stego image. The word local that was used in (7) means that we used a window with a size of 5. The value of variation in this window shows the strength of edges.

The average values of MSE, PSNR, and PSPNR for SIPI database images are presented in Table 4. It can be observed that our proposed method has better MSE and PSNR, compared to [18] and [19], and better PSPNR for all 3 methods. Since the proposed method changes each HH coefficient by a maximum of one unit, the difference between many pixels in the stego image relative to the cover is less than the jnd threshold. For this reason, the proposed method is much more successful than other methods in the PSPNR criterion. It should be noted that in some cover images, the PSPNR criterion calculated for the proposed method is the INF value. Since the new PSO-based method [22] has a high embedding capacity, this method was selected to compare the proposed method in the case of high capacity. For this comparison, the test images of Figure 4 were used. In [22], several methods have been proposed and tested, the most successful of which were PSO-LSB, PSO-IWT, and PSO-DWT. Table 7 presents the PSNR and SSIM results of these methods and the proposed method for the three sample images. Examination of these results shows that the proposed method for both Baboon and Barbara images is more successful than all the methods presented in [22], and in the case of Lena image, the PSO-DWT method is superior in terms of PSNR criterion.

**Table 6**. MSE, PSNR, and PSPNR results for SIPI images (low capacity)

| | MSE | PSNR | PSPNR |
|---|---|---|---|
| [18] | 0.6094 | 50.28 | 54.94 |
| [19] | 0.4176 | 51.92 | 55.07 |
| [20] | 0.2788 | 53.68 | 57.35 |
| MYXORGA(1) | 0.3620 | 52.53 | 71.77 |

**Table 7**. PSNR and SSIM results for test images (high capacity)

| | Lena | | Baboon | | Barbera | |
|---|---|---|---|---|---|---|
| | PSNR | SSIM | PSNR | SSIM | PSNR | SSIM |
| PSO-LSB [22] | 51.14 | 0.9982 | 51.14 | 0.9987 | 51.14 | 0.9972 |
| PSO-IWT[22] | 50.48 | 0.9977 | 46.44 | 0.9985 | 49.85 | 0.9972 |
| PSO-DWT[22] | 51.64 | 0.9981 | 51.13 | 0.9974 | 51.11 | 0.9974 |
| MYXORGA (3) | 51.23 | 0.9980 | 51.21 | 0.9989 | 51.21 | 0.9974 |

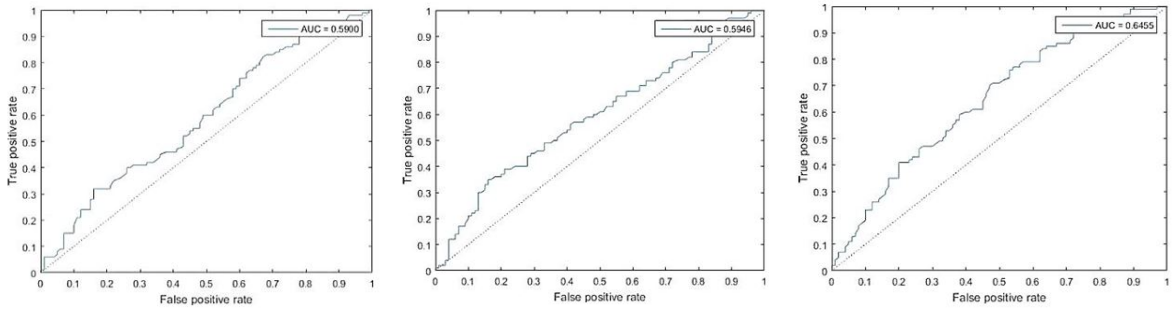
### 4.3 Security Analysis

The security test of a steganography method is a very important issue for the feasibility of the method in practice but is often overlooked in most articles. A steganography method produces a stego image by modifying the cover image. Steganalysis attacks use these changes to discover the existence of hidden data. With the discovery of the existence of data, the method of steganography has failed. With the increase in the number of these changes, the steganography method is more likely to be failed. One of the numerical parameters indicating the accuracy of each attack is the AUC value, which is the area under the ROC diagram. This area is normalized to 1 for a fully successful discovery method. If the AUC is close to 0.5, the attack will be unsuccessful, and therefore, the embedding method is safer. SPAM attack with a 686-dimensional feature vector [35] was used to compare the security of the proposed method. To perform this test, images of the second dataset are used. The ROC diagram of this attack for different methods in two embedding levels is shown in Figure 5 and Figure 6. The AUC values of the attack for these two embedding levels are also presented in Table 8 and Table 9.

A review of the SPAM attack results shows that the security of the proposed method is improved by using the genetic algorithm. On the other hand, the performance of the proposed low-capacity method is

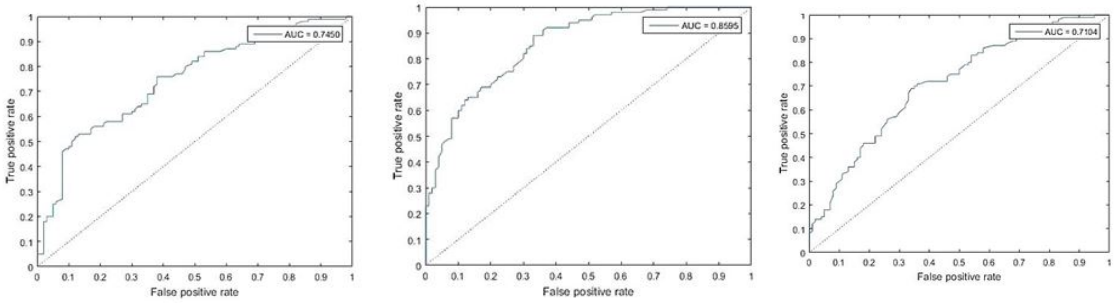
**Table 8**. AUC results for SPAM attack for the images of INRIA Holidays Dataset (low embedding)

| | AUC |
|---|---|
| MYXOR(1) | 0.6455 |
| MYXORGA(1) | 0.5964 |
| OPAPGA(1) | 0.5900 |




(a) (b) (c)

**Figure 5**. ROC diagram for SPAM attack against (a) MYXOR (1), (b) MYXORGA (1), and (c) OPAPGA (1) methods (low capacity)




(a) (b) (c)

**Figure 6**. ROC diagram for SPAM attack against (a) MYXOR (3), (b) OPAPGA (3), and (c) OPAPGA (1) methods (high capacity)

**Table 9**. AUC results for SPAM attack for the images of INRIA Holidays Dataset (high embedding)

| | AUC |
|---|---|
| MYXORGA(3) | 0.7104 |
| OPAPGA(3) | 0.8595 |
| OPAPGA(1) | 0.7450 |

similar to the OPAPGA method. The accuracy of the attack to detect the proposed method is about 0.59 when embedding is done in the $HH$ subband (low capacity). Another important point is the higher security of the proposed method at a higher capacity (embedding is done in three subbands) than the OPAPGA method, which demonstrates the success of the proposed method over the OPAPGA method. Finally, the method is operational in practice.

## 5 Conclusion

Image steganography with data embedding in DWT coefficients is one of the most attractive and successful areas of information hiding. The proposed method in this paper uses optimization algorithms to embed in this domain. In this way, the image is divided into $8 \times 8$ non-overlapping blocks. Then, a $2 - D$ DWT is applied to each block. No change in the coefficients will occur if the XOR result of the two LSBs of the two adjacent coefficients is identical to the two bits of secret data. Otherwise, a one-unit increase or decrease of one or both coefficients will produce the desired XOR result. The proposed method uses a genetic algorithm to select a new value for the pair of adjacent coefficients that need to be changed. Comparison results at both embedding levels (high and low) show that the proposed method produces stego images with very high PSPNR and in most cases is more successful than existing methods in producing high-quality stego images. The discovery accuracy of the proposed method by SPAM attack at a low and high capacity is about 59% and 75%, respectively. Therefore, the proposed method in this paper, in addition to having an appropriate stego image quality and capacity, has good security against existing attacks. One of the most important advantages of the proposed method is providing a solution for image steganography with a short embedding key so that it is practically feasible.

## References

[1] Inas Jawad Kadhim, Prashan Premaratne, Peter James Vial, and Brendan Halloran. Comprehensive survey of image steganography: Techniques, evaluations, and trends in future research. *Neurocomputing*, 335:299–326, 2019.

[2] Abbas Cheddad, Joan Condell, Kevin Curran, and Paul Mc Kevitt. Digital image steganography: Survey and analysis of current methods.

*Signal processing*, 90(3):727–752, 2010.

[3] Mehdi Hussain, Ainuddin Wahid Abdul Wahab, Yamani Idna Bin Idris, Anthony TS Ho, and Ki-Hyun Jung. Image steganography in spatial domain: A survey. *Signal Processing: Image Communication*, 65:46–66, 2018.

[4] Eugene T Lin and Edward J Delp. A review of data hiding in digital images. In *PICS*, volume 299, pages 274–278, 1999.

[5] Gyan Singh Yadav and Aparajita Ojha. Hamiltonian path based image steganography scheme with improved imperceptibility and undetectability. *Applied Soft Computing*, 73:497–507, 2018.

[6] Gandharba Swain and Saroj Kumar Lenka. Classification of image steganography techniques in spatial domain: a study. *Journal of Computer Science & Engineering Technology (IJCSET)*, 5(03):219–232, 2014.

[7] S Arunkumar, V Subramaniyaswamy, V Vijayakumar, Naveen Chilamkurti, and R Logesh. Svd-based robust image steganographic scheme using riwt and dct for secure transmission of medical images. *Measurement*, 139:426–437, 2019.

[8] Manashee Kalita and Themrichon Tuithung. A comparative study of steganography algorithms of spatial and transform domain. *International Journal of Computer Applications*, 975:8887, 2016.

[9] Katzenbeisser Stefan, Petitcolas Fabien AP, et al. Information hiding techniques for steganography and digital watermarking. 2000.

[10] Shiv K Sahu, Shachi Sahu, Vahid Nourani, Chief Advisory Board, Uma Shanker, Rama Shanker, Vinita Kumari, Kapil Kumar Bansal, Deepak Garg, Vijay Anant Athavale, et al. Untitled-international journal of engineering and advanced.

[11] Stéphane Mallat. *A wavelet tour of signal processing*. Elsevier, 1999.

[12] Ingrid Daubechies. *Ten lectures on wavelets*. SIAM, 1992.

[13] Aparna Vyas and Joonki Paik. Review of the application of wavelet theory to image processing. *IEIE Transactions on Smart Processing and Computing*, 5(6):403–417, 2016.

[14] Inas Jawad Kadhim, Prashan Premaratne, and Peter James Vial. High capacity adaptive image steganography with cover region selection using dual-tree complex wavelet transform. *Cognitive Systems Research*, 60:20–32, 2020.

[15] KB Raja, Kiran Kumar, Satish Kumar, MS Lakshmi, H Preeti, KR Venugopal, and Lalit M Patnaik. Genetic algorithm based steganography using wavelets. In *International Conference on Information Systems Security*, pages 51–63. Springer, 2007.

[16] Elham Ghasemi, Jamshid Shanbehzadeh, and Nima Fassihi. High capacity image steganography based on genetic algorithm and wavelet transform. In *Intelligent Control and Innovative Computing*, pages 395–404. Springer, 2012.

[17] Avinash K Gulve and Madhuri S Joshi. An image steganography method hiding secret data into coefficients of integer wavelet transform using pixel value differencing approach. *Mathematical Problems in Engineering*, 2015, 2015.

[18] Hayat Al-Dmour and Ahmed Al-Ani. A steganography embedding method based on edge identification and xor coding. *Expert systems with Applications*, 46:293–306, 2016.

[19] Aref Miri and Karim Faez. Adaptive image steganography based on transform domain via genetic algorithm. *Optik*, 145:158–168, 2017.

[20] Aref Miri and Karim Faez. An image steganography method based on integer wavelet transform. *Multimedia Tools and Applications*, 77(11):13133–13144, 2018.

[21] Sabyasachi Pramanik, RP Singh, and Ramkrishna Ghosh. Application of bi-orthogonal wavelet transform and genetic algorithm in image steganography. *Multimedia Tools & Applications*, 79, 2020.

[22] Pranab K Muhuri, Zubair Ashraf, and Swati Goel. A novel image steganographic method based on integer wavelet transformation and particle swarm optimization. *Applied Soft Computing*, 92:106257, 2020.

[23] Hamidreza Rashidy Kanan and Bahram Nazeri. A novel image steganography scheme with high embedding capacity and tunable visual image quality based on a genetic algorithm. *Expert systems with applications*, 41(14):6123–6130, 2014.

[24] Ran-Zan Wang, Chi-Fang Lin, and Ja-Chen Lin. Image hiding by optimal lsb substitution and genetic algorithm. *Pattern recognition*, 34(3):671–683, 2001.

[25] Pratik D Shah and RS Bichkar. A secure spatial domain image steganography using genetic algorithm and linear congruential generator. In *International Conference on Intelligent Computing and Applications*, pages 119–129. Springer, 2018.

[26] Ranyiah Wazirali, Waleed Alasmary, Mohamed MEA Mahmoud, and Ahmad Alhindi. An optimized steganography hiding capacity and imperceptibly using genetic algorithms. *IEEE Access*, 7:133496–133508, 2019.

[27] Lifang Yu, Yao Zhao, Rongrong Ni, and Zhenfeng Zhu. Pm1 steganography in jpeg images using genetic algorithm. *Soft Computing*, 13(4):393–400, 2009.

[28] V Sabeti, S Faiazi, and H Shirinkhah. Improving security of lsbm steganography using of genetic

algorithm, mmulti-key and blocking. 2020.

[29] Rinita Roy and Sumit Laha. Optimization of stego image retaining secret information using genetic algorithm with 8-connected psnr. *Procedia Computer Science*, 60:468–477, 2015.

[30] Amrita Khamrui, Diotima Dutta Gupta, Shatadal Ghosh, and Sambhunath Nandy. A spatial domain image authentication technique using genetic algorithm. In *International Conference on Computational Intelligence, Communications, and Business Analytics*, pages 577–584. Springer, 2017.

[31] SI Nipanikar, V Hima Deepthi, and Nikita Kulkarni. A sparse representation based image steganography using particle swarm optimization and wavelet transform. *Alexandria engineering journal*, 57(4):2343–2356, 2018.

[32] Sahib Khan and Tiziano Bianchi. Ant colony optimization (aco) based data hiding in image complex region. *International Journal of Electrical & Computer Engineering (2088-8708)*, 8(1), 2018.

[33] Anan Banharnsakun. Artificial bee colony approach for enhancing lsb based image steganography. *Multimedia Tools and Applications*, 77(20):27491–27504, 2018.

[34] Chun-Hsien Chou and Yun-Chin Li. A perceptually tuned subband image coder based on the measure of just-noticeable-distortion profile. *IEEE Transactions on circuits and systems for video technology*, 5(6):467–476, 1995.

[35] Tomáš Pevny, Patrick Bas, and Jessica Fridrich. Steganalysis by subtractive pixel adjacency matrix. *IEEE Transactions on information Forensics and Security*, 5(2):215–224, 2010.

**Vajiheh Sabetiis** is an Assistant Professor of Engineering and Technology department at Alzahra University. She received her B.Sc. degree in Software Engineering in 2004 and her M.Sc. degree in Computer Architecture in 2007 and her Ph.D. degree in Computer Engineering in 2012 from the Electrical and Computer Engineering department of Isfahan University of Technology (IUT), Isfahan, Iran. Her research interests are softcomputing, image processing, and information hiding (steganography, watermarking).

**Mahsa Amerehei** received the B.Sc. degree from the computer engineering department, the university of Arak, Iran, in 2014 and the M.Sc. degree in computer software engineering from Alzahra University, Tehran, Iran, in 2017. She was Professor Assistant for software engineering courses in Alzahra University, from Februry 2015 until July 2015. After spending some time working on software system analysis, she became a system analyst at EIT Company, Tehran, Iran, in 2016. Her main research is in the field of jpeg image steganography in wavelet domains, image steganalysis and genethic algorithms.