Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2013, Article ID 824239, 10 pages
http://dx.doi.org/10.1155/2013/824239

Research Article

Constructing HVS-Based Optimal Substitution Matrix Using Enhanced Differential Evolution

Shu-Fen Tu[1] and Ching-Sheng Hsu[2]

[1] *Chinese Culture University, No. 55 Huagang Road, Shihlin District, Taipei City, Taiwan*
[2] *Ming Chuan University, No. 5 Deming Road, Gueishan Township, Taoyuan County, Taiwan*

Correspondence should be addressed to Ching-Sheng Hsu; cshsu@mail.mcu.edu.tw

Received 11 July 2013; Revised 26 October 2013; Accepted 27 October 2013

Academic Editor: Gonzalo Pajares

Least significant bit (LSB) substitution is a method of information hiding. The secret message is embedded into the last k bits of a cover-image in order to evade the notice of hackers. The security and stego-image quality are two main limitations of the LSB substitution method. Therefore, some researchers have proposed an LSB substitution matrix to address these two issues. Finding the optimal LSB substitution matrix can be conceptualized as a problem of combinatorial optimization. In this paper, we adopt a different heuristic method based on other researchers' method, called enhanced differential evolution (EDE), to construct an optimal LSB substitution matrix. Differing from other researchers, we adopt an HVS-based measurement as a fitness function and embed the secret by modifying the pixel to a closest value rather than simply substituting the LSBs. Our scheme extracts the secret by modular operations as simple LSB substitution does. The experimental results show that the proposed embedding algorithm indeed improves imperceptibility of stego-images substantially.

1. Introduction

The internet provides an easy way to exchange information with others. However, information is also prone to eavesdropping from hackers. Several methods can be employed to protect secret information, such as cryptography, steganography, and secret sharing schemes. The spirit of these methods is basically varied. Cryptography scrambles the content with a private key. Without the appropriate key, unauthorized authors cannot decode the secret within limited time and resources. Unlike cryptography, steganography, also called information hiding, conceals the secret rather than scrambling it. That is, the secret is covered by innocent information that does not attract the attention of hackers, who thereby pass over it. Due to its simplicity and efficiency, steganography is still a popular method until now [1–4].

Among plenty of steganographic methods, simple least significant bit (LSB) substitution is the most general one [5]. The secret message is decomposed and embedded into the least significant bits of each pixel of the cover-image. The modified cover-image is called a stego-image. The secret message can be extracted by performing modular operation to each pixel of the stego-image. This method is very simple and easy to implement; however, sometimes the stego-image is not imperceptible enough when more least significant bits are substituted. Recently, Wang et al. proposed a novel idea about substitution matrix to improve the quality and security of the stego-image [6]. The substitution matrix can be seen as a mapping function, which maps each secret value into another value. Different substitution matrices represent different mappings and result in different stego-images. Among these different stego-images, some are closer to the original cover-image than others are. Obviously, the optimal substitution matrix is the one that produces the stego-image closest to the cover-image. Due to the huge number of possible substituion matrices, Wang et al. utilized genetic algorithms (GA) [7] to find the optimal substitution matrix. According to the patterns of chromosomes, GA can be classified into two types: one is binary GA, and the other is real-parameter GA. In the course of evolution, binary GA has to encode the original problem into binary chromosomes, and the encoding method may influence the

efficiency of problem-solving. However, some problems, such as combinatorial optimization, are not easy to be encoded into binary chromosomes. Furthermore, the length of chromosomes of such kind of problems may be too long to solve problems efficiently. Although real-parameter GA can encode a problem with shorter chromosomes, its efficiency and the quality of the final solution are not as good as those of binary GA. Therefore, each kind of GA has its limitation.

Later, other researchers adopt different heuristic methods, such as tabu search [8], ant colony algorithm [9], and cat swarm optimization [10], to construct optimal substitution matrix. All of these researchers adopt simple LSB substitution to embed the secret. However, even though an optimal substitution matrix is adopted, the modification to the pixels of the cover-image may be still large due to the intrinsic features of simple LSB substitution. Consequently, the improvement of the imperceptibility may be limited. Besides, these researchers adopt peak signal-to-noise ratio (PSNR) as a fitness function to measure how the stego-image is near the cover-image. The PSNR number represents the average differences between pixels; however, sometimes the difference between pixels cannot respond to human perception. Generally speaking, human eyes can tolerate modifications to texture areas more than those to smooth areas [11]. Images are viewed by human eyes after all; hence it is more suitable to use the measurement based on the human visual system (HVS) to evaluate the imperceptibility of the stego-image.

This paper proposed a method to construct optimal substitution matrix as well. Nevertheless, the proposed scheme has three aspects different from those in other similar researches. At first, the way to embed secrets is to change the pixel values rather than to substitute the least significant bits directly. However, the way of extracting is easy and the same as that of simple LSB substitution. Second, another heuristic method, called enhanced differential evolution (EDE) [12], is adopted to search for the optimal substitution matrix. Differential Evolution (DE) was first introduced by Storn and Price [13] and copes with problems whose feasible solutions are continuous values. Later, Onwubolu and Babu extend the capability of DE to handle problems with discrete solutions. Third, an HVS-based fitness function, called structural similarity SSIM [14], is employed to measure the difference between the stego-image and the cover-image. Therefore, our stego-image is not only physically near the cover-image but is also perceived similar to the cover-image by human eyes. The rest of this paper is organized as follows. In Section 2, some preliminary knowledge for our work is provided. In addition, some related literatures are reviewed as well. In Section 3, the way of constructing an optimal substitution matrix and the embedding and extraction methods are explained in detail. Then, the experimental results and comparisons with other researchers methods are presented in Section 4. Finally, we will give some conclusions in Section 5.

2. Literature Review

2.1. Simple LSB Substitution. The simple LSB substitution is the earliest steganographic technique. The so-called least significant bit is the less important part of a pixel. Therefore, modifying LSBs of pixels cannot change an image too much. Embedding and extracting secrets are very simple and easy to implement. Suppose that p denotes a pixel of the cover-image and is expressed as

$$p = q \times 2^k + r. \tag{1}$$

That is, q and r are the quotient and the remainder when p is divided by 2^k. Suppose that s is a k-bit secret. The secret s can be embedded into p by means of the following:

$$p' = q \times 2^k + s. \tag{2}$$

Performing a modulo-operation on the stego-pixel p', as shown in (3), can extract the secret s:

$$s = p' \bmod 2^k. \tag{3}$$

Simply speaking, the secret s is embedded by directly substituting the last k bits of p and is retrieved from the last k bits of p'. Take a pixel $(00100000)_2$ and a 3-bit secret message $(111)_2$ as an example. Since the length of the secret message is three, we can substitute the last three bits of the pixel with the secret. Using (1) and (2), we can get the stego-pixel $(00100111)_2$. If the secret s is very large, it is divided into segments of fixed length and is evenly distributed into each pixel of the cover-image.

There are two problems of this method. First, the more secret messages there are, the more bits of the cover-image have to be modified. Hence the stego-image may become too different from the cover-image to give cover to the secret message inside. Second, the simplicity is a two-edged sword. The receiver can recover the secret easily, so do the hackers. Therefore, the security of this method has to be enhanced.

2.2. Substitution Matrix. In 2001, Wang et al. introduced the substitution matrix to improve the quality and security of the stego-image [6]. Briefly speaking, a substitution matrix is used to replace the secret value with another value. Wang et al.'s method can be summarized as follows. At first, the secret s is divided into segments of k-bit length, and then the order of each segment is randomly permuted. Suppose that E denotes the set of reordered segments of s and that

$$E = \left\{ e_t \mid 0 \le e_t \le 2^k - 1,\ 1 \le t \le n \right\}, \tag{4}$$

where n is the number of total segments of s. Let M denote a $2^k \times 2^k$ substitution matrix, and $M = [m_{i,j}]$, where $0 \le i, j \le 2^k - 1$, and $m_{i,j} \in \{0, 1\}$. According to M, every element of E is changed into another value as shown in

$$e_t = j \quad \text{if } m_{e_t,j} = 1. \tag{5}$$

Note that there is only a "1" in each row and each column. In short, the substitution matrix can be seen as a one-to-one mapping function from A to A, where A is the set of all possible integer values of e_t. Then, the mapping result is

embedded into the cover-image by means of simple LSB. The following serves as an example:

$$E = \{1, 3, 0, 2, 1\},$$
$$M = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (6)$$

The elements m_{00}, m_{12}, m_{21}, and m_{33} of M indicate that the possible values 0, 1, 2, and 3 of E are substituted with 0, 2, 1, and 3, respectively. Therefore, E is changed into another set E' as $\{2, 3, 0, 1, 2\}$.

Obviously, there are various possible substitution matrices. Different substitution matrix produces different E' and further produces different stego-image. Wang et al. defined an optimal substitution matrix as the one that produces a stego-image with maximal peak signal-to-noise ratio (PSNR), where

$$\text{PSNR} = 10 \times \log \frac{255^2}{\text{MSE}}, \quad (7)$$

$$\text{MSE} = \frac{\sum_{i=1}^{M} \sum_{j=1}^{N} \left(p_{i,j} - p'_{i,j}\right)^2}{M \times N}. \quad (8)$$

In (8), M and N denote the width and height of the cover-image, respectively. And $p_{i,j}$ and $p'_{i,j}$ denote the pixels of the cover-image and the stego-image, respectively. Essentially, finding an optimal substitution matrix is a kind of combinatorial optimization problem, and there are totally $2^k!$ possible solutions. Moreover, the solution space rapidly grows up with the number of k. If $k = 4$, for example, the number of possible solutions becomes 20, 922, 789, 888, 00. When solving optimization problems with large solution space, heuristic algorithms perform better than deterministic algorithms. Therefore, Wang et al. utilized genetic algorithm (GA) to find near-optimal substitution matrix.

2.3. Optimal LSB-Based Steganography. Some other researchers apply Wang et al.'s substitution matrix to improve the quality of the stego-images. The main difference is that they adopt different optimization algorithms, especially bioinspired algorithms. In 1992, Dorigo proposed an ant colony optimization (ACO) algorithm [15] in his Ph.D. thesis, which is very suitable for solving combinatorial optimization problems. Since finding an optimal substitution matrix is a kind of combinatorial optimization problem, Hsu and Tu [9] adopted ACO to find optimal substitution matrix. In 2007, Chu and Tsai introduced a cat swarm optimization (CSO) algorithm [16], which is derived from the behavior of cats. Valuing the performance on finding the global best solutions, Wang et al. [10] gave some revisions to CSO to generate optimal substitution matrix. When using bioinspired algorithms, one needs to provide fitness function to evaluate a solution so that the algorithm can guide those virtual creatures, such as cats or ants, toward the optimal solution. Hsu and Tu and Wang et al. utilized the pixel difference between the cover-image and the stego-image as the fitness of a solution.

Some researchers adopt different embedding strategies to make the distortion to the cover-image as little as possible. Xu et al. [17] adopted Mielikaines' pairwise LSB matching method [18] and changed the matching order between the secret bits and cover pixels to decrease the distortion to the cover-image. They designed a three-tiered score system to evaluate the performance of a matching order and utilized an immune programming to find the best matching order. Considering that the difference measured in pixels is not necessary the same as that measured by human eyes, a few of researchers take human visual system into consideration. Lee and Tsai [19] determined the number of bits used to carry secret in a pixel according to the principle of just noticeable difference (JND). Further, they utilized dynamic programming to divide the secret data into segments to minimize the modification to the cover-image when embedding secret data. Instead of using every pixel of a block, Bedi et al. [20] chose a part of the pixels in a block to carry secret data. In view of the image quality, the choice is not made at random. For each block of 8×8 pixels, they utilized particel swarm optimization (PSO) algorithm [21] to determine the best pixels to embed secret data sequentially. The distortion error between the cover-image and the stego-image is measured with a quality index based on human visual system. Since the pixels used to embed secret data vary from block to block, the pixel positions have to be recorded as the key to extract secret data successfully. If the size of the cover-image is $M \times N$ pixels, the minimal required space for the key is $M \times N$ bits. Obviously, the required space grows up with the size of the cover-image. Another worry about Bedi et al.'s scheme is about hiding capacity. Not all of the pixels in a block will be used to embed secret data; or else, Bedi et al.'s scheme becomes meaningless. In fact, in their experiments, only eight pixels of a block are used to embed secret data. The highest possible payload is only 0.5 bits per pixel if the last four bits of a pixel are used to embed data.

2.4. HVS-Based Measurement. Human eyes are complex biological organs. The way human eyes perceive the difference between two images is not the same as that of PSNR. Sometimes, a sensible difference for human eyes does not necessarily mean a large difference between pixels. After some observations, Barni and Bartolini [11] listed the following three rules of thumb.

(1) Disturbs are much less visible on highly textured regions than on smooth areas.

(2) Contours are more sensible to noise addition than highly textured regions but less than flat areas.

(3) Disturbs are less visible over dark and bright regions.

Based on the characteristics of human visual system (HVS), some researchers proposed different methods to evaluate image quality or to estimate the acceptable change to an image [22–26].

Combining the three components of luminance, contrast, and structure, Wang et al. proposed a structural similarity

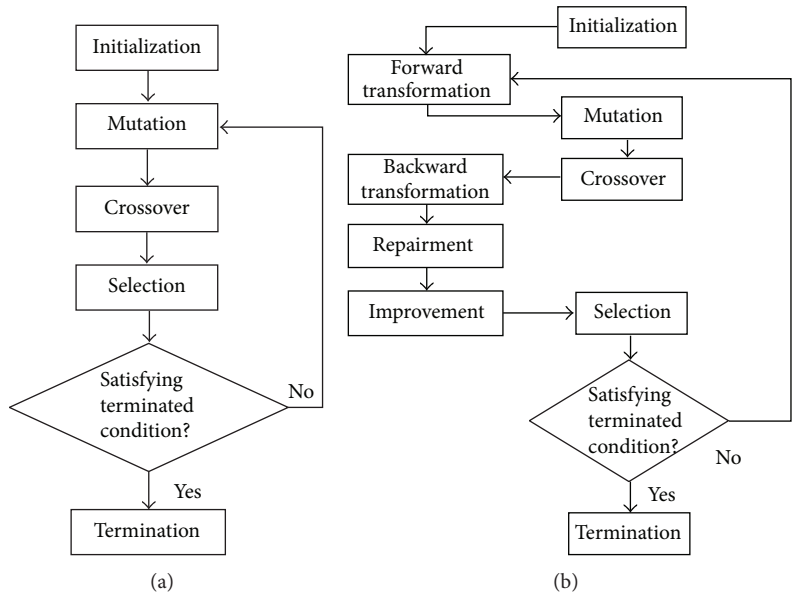


Figure 1: Flowcharts of (a) canonical DE and (b) enhanced DE.

(SSIM) index to measure the similarity between two images in light of HVS. Suppose that x and y denote two gray-level images, respectively. The luminance comparison function is

$$l(x, y) = \frac{2\mu_x\mu_y + C_1}{\mu_x^2 + \mu_y^2 + C_1}, \tag{9}$$

where μ_x and μ_y denote the average pixel values of images x and y, respectively, and $C_1 = (K_1L)^2$, where L is the dynamic range of pixel values and $K_1 \ll 1$. The contrast comparison function is

$$c(x, y) = \frac{2\sigma_x\sigma_y + C_2}{\sigma_x^2 + \sigma_y^2 + C_2}, \tag{10}$$

where σ_x and σ_y denote the standard deviations of pixel values of images x and y, respectively, and $C_2 = (K_2L)^2$ where $K_2 \ll 1$. The structure comparison function is

$$s(x, y) = \frac{\sigma_{xy} + C_3}{\sigma_x\sigma_y + C_3}, \tag{11}$$

where σ_{xy} is the covariant of pixel values of images x and y, respectively. Combining (9), (10), and (11), we can get the following SSIM index:

$$\mathrm{SSIM}(x, y) = [l(x, y)]^\alpha \bullet [c(x, y)]^\beta \bullet [s(x, y)]^\gamma. \tag{12}$$

For simplicity, Wang et al. set $\alpha = \beta = \gamma = 1$ and $C_3 = C_2/2$. Consequently, SSIM can be transformed into a specific form as follows:

$$\mathrm{SSIM}(x, y) = \frac{(2\mu_x\mu_y + C_1)(2\sigma_{xy} + C_2)}{(\mu_x^2 + \mu_y^2 + C_1)(\sigma_x^2 + \sigma_y^2 + C_2)}. \tag{13}$$

The SSIM metric is calculated on various windows of the image, and hence we can use the following mean SSIM (MSSIM) index to evaluate the overall image quality:

$$\mathrm{MSSIM}(X, Y) = \frac{1}{M}\sum_{j=1}^{M} \mathrm{SSIM}(x_j, y_j). \tag{14}$$

2.5. Enhanced Differential Evolution. Differential evolution (DE) was first introduced by Storn and Price [13]. DE is a population-based optimization method, and candidate solutions are represented as vectors. For each individual (called a target vector) in the current population, offspring (called a trial vector) is generated by adding a scaled, random vector difference to a randomly selected population vector. The trial vector competes with its corresponding target vector on their fitness. The winner can live to the next generation. Although simple, DE performs well on a wide variety of test problems [12, 13, 27, 28]. Figure 1(a) is the flowchart of DE. Initially, DE is invented for solving continuous space optimization problems. Later, some researchers modified DE to attack permutative-based combinatorial optimization problem. The so-called permutative-based combinatorial optimization problem is that its candidate solution is a permutation of a sequence of integers. Among these modifications, Onwubolu and Babu's approach, called enhanced differential evolution (EDE), is intuitive and easy to implement [12]. The main idea of this approach is to transform the permutative population into continuous population. The forward transformation formula is as follows:

$$x_i' = -1 + \left(\frac{f \times 5}{10^3 - 1}\right) \times x_i, \tag{15}$$

where x_i is a discrete parameter of some vector and f is a scaling factor. After being transformed into continuous form, the population can be handled by canonical DE strategy

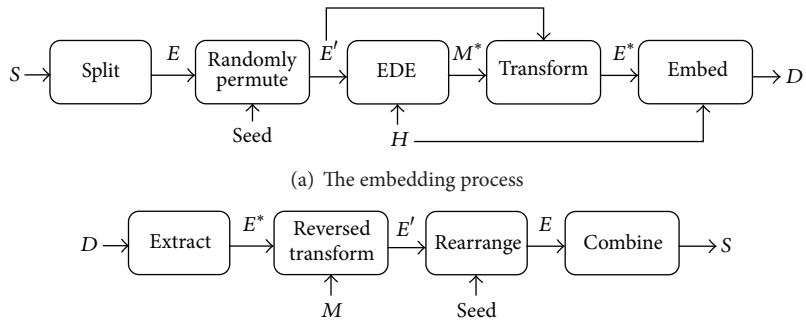


Figure 2: The flowchart of the proposed scheme.

to generate the child population. However, the individuals of the child population are continuous values and cannot be evaluated by the fitness function. Therefore, they have to be backward transformed into discrete solutions by the following equation:

$$x_i = \text{int}\left[\frac{\left(1 + x_i'\right) \times \left(10^3 - 1\right)}{5 \times f}\right], \tag{16}$$

where int[·] denotes a function that rounds a real value to the nearest integer. Backward transformation may produce infeasible solutions, so the offspring population has to be repaired. To generate better offspring, Onwubolu and Babu proposed two improvement strategies for the repaired offspring: one is swap mutation and the other is insertion mutation. The final offspring will compete with the parents. Figure 1(b) shows the flowchart of EDE.

3. The Proposed Method

With the help of a substitution matrix, the imperceptibility can be improved, but the embedding way of simple LSB substitution may limit the improvement. Several researches have been devoted to the study of constructing an optimal substitution matrix. Some utilized deterministic algorithms [29, 30], while some utilized heuristic algorithms [6, 8–10] to search for the optimal matrix. No matter what algorithms they employed, it is PSNR that they adopted as the objective function (also called fitness function in heuristic algorithms) to guide the search direction. As we have mentioned above, PSNR measures the absolute difference of pixel values, but not the difference perceived by human eyes. Therefore, we adopt MSSIM (14) as the fitness function. The heuristic algorithm we utilized to search for the optimal substitution matrix is EDE. In addition, to break through the intrinsic limitation of simple LSB substitution, we design a *ModEmbedding* algorithm, which can make the cover-image and stego-image as close as possible.

Before moving on to the main task, it is helpful to give an overview of our scheme. Figure 2 is the flowchart of the proposed scheme. In the embedding process, a secret S is split into segments, each of which is of k-bit length. Let E denote the set of segments. Then, all elements of E are randomly permuted using a pseudorandom number generator. According to E' and the cover-image H, EDE constructs near-optimal substitution matrix M^*. With M^*, E' is transformed into E^* and embedded into H. And finally, we can get the stego-image D with the secret inside. In the extraction process, E^* is extracted from the stego-image D and is reversely transformed into E' with the same substitution matrix M. Using a pseudorandom number generator seeded by the same key, we can rearrange the elements of E' in the original order of the elements of E. Finally, combining the elements of E, we can recover the secret S.

With the overview in mind, we can now look deeper into the details of the proposed scheme.

3.1. EDE Subroutine. In this paper, EDE is employed to construct a near-optimal substitution matrix. Since initialization and selection are problem-dependent parts of EDE, we will concentrate on these two parts.

Initialization. EDE is a population-based evolutaionary algorithm. Therefore, a population size N_p has to be predefined at first. Initially, users have to randomly generate a set of N_p distinct candidate solutions. Starting from the initial population, EDE will generate offspring and evolve continuously to find the optimal solution until the terminated condition is satisfied. In order to use EDE to solve problems, we first need to represent solutions in form of vectors. As regards the problem of the proposed scheme, a solution is in the form of a $2^k \times 2^k$ matrix. Precisely speaking, as we have mentioned before, it is a one-to-one mapping from the set A to the set A, where $A = \{a_i \mid i = 0, \ldots, (2^k - 1), 0 \le a_i \le 2^k - 1\}$. Consequently, we can simply represent a substitution matrix as a permutation of the set A. Therefore, it is obvious that a substitution matrix can be represented as a permutation of 0 to $2^k - 1$. We will now explain more definitely how a substitution matrix is encoded into a vector in EDE. Suppose that the substitution matrix $M = [m_{i,j}]$, where $0 \le i, j \le 2^k - 1$, and $m_{i,j} \in \{0, 1\}$. Then, the corresponding vector $\mathbf{X} = [x_0, x_1, \ldots, x_{2^k-1}]$, where $x_i = j$ if $m_{i,j} = 1$. Take the following

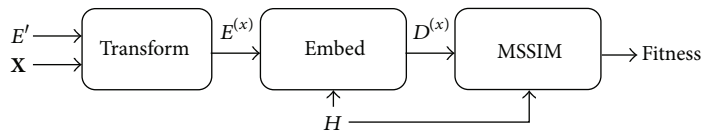


Figure 3: The process of computing fitness.

substitution matrix as an example. The corresponding vector is $[0\ \ 2\ \ 1\ \ 3]$. Consider

$$M = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{17}$$

Selection. In the selection phase, each individual (i.e., vector) of the current population has to compete with its offspring. The competition is based on their quality; hence users have to provide a fitness function to score vectors. Here we adopt (14) as our fitness function. Figure 3 illustrates the detailed process of computing a fitness of a vector $\mathbf{X}$. At first, E' is transformed into the substitution matrix corresponding to the vector X. Next, the transformed result $E^{(x)}$ is embedded into the cover-image H to get the stego-image $D^{(x)}$. Therefore, the fitness of a vector is MSSIM $(H, D^{(x)})$.

3.2. Embedding. Before explaining the way of the proposed embedding, let us consider the following example. Suppose that a cover-pixel is 60 and a 2-bit secret is 3. According to (1), 60 is expressed as follows:

$$60 = 15 \times 2^2 + 0. \tag{18}$$

In light of simple LSB substitution, the secret substitutes the remainder directly and hence results in the following stego-pixel:

$$63 = 15 \times 2^2 + 3. \tag{19}$$

Performing 63 **mod** 2^2, we can extract the secret. Let us consider another situation. If we change the cover-pixel to 59, instead of 63, we still can extract the secret by performing the same modulo-operation (i.e., 59 **mod** 2^2) because $59 = 14 \times 2^2 + 3$. It is clear that 59 is closer to the cover-pixel 60 than 63 is. This example makes it clear that we can test (2) on three quotients $(q - 1)$, q, and $(q + 1)$ to see which one can result in a stego-pixel closest to the cover-pixel. The complex embedding algorithm is as Algorithm 1.

3.3. Extraction. Extracting secret is very simple. Algorithm 2 illustrates the extracting algorithm. Each k-bit secret e is extracted concatenated to the whole secret E'. However, E^* is not the original secret. We have to convert each k-bit value of E^* to the original value according to the substitution matrix M.

Algorithm *ModEmbedding* (H, E^*)
  $D \leftarrow \varnothing$
  for each pixel $p \in H$ and each k-bit element $e \in E^*$
    $q \leftarrow \lfloor p/2^k \rfloor$
    $r \leftarrow p$ **mod** 2^k
    $a^* \leftarrow \arg\min_{a \in \{-1,0,1\}} |a \times 2^k + r - e|$
    $p' \leftarrow (q + a^*) \times 2^k + e$
    $D \leftarrow D \cup \{p'\}$
  return D

Algorithm 1: The proposed embedding algorithm.

Algorithm *Extracting* (D)
  $E^* \leftarrow \varnothing$
  for each pixel $p' \in D$
    $e \leftarrow p'$ **mod** 2^k
    $E^* \leftarrow E^* \cup \{e\}$
  return E^*

Algorithm 2: The proposed extracting algorithm.

4. Experimental Results and Discussions

This section demonstrates some experimental results of the proposed method. In addition, the proposed scheme was compared with some simulated experiments. The experiments in this section are carried out on a PC with Intel Core 2 Duo CPU at 2.8 GHz, 4 GB RAM, Windows 7 Professional Operating System, NetBeans IDE, and JDK 6. Before turning to a closer examination of the experimental results, we will outline our assumptions here.

(1) The cover-image is a gray-level and uncompressed image.

(2) The stego-image cannot be modified by any form of signal processing.

(3) The key is preserved secretly.

(4) The size of a secret image is of $(w \times h)/2$ pixels, where w and h are the width and the height of the cover-image, respectively.

Having clarified the assumptions, we may now go into details about our experiments. Here we have three distinct types of simulationsas follows.

(a) *Experiment I.* The secret was embedded and extracted by means of simple LSB substitution.

Table 1: Summarizations of the simulated experiments and the proposed scheme.

	(a) Experiment I	(b) Experiment II	(c) Experiment III	(d) Ours
With substitution matrix	No	Yes	Yes	Yes
Fitness function	—	PSNR	MSSIM	MSSIM
Embedding way	Equation (2)	Equation (2)	Equation (2)	ModEmbedding
Extracting way	Equation (3)	Equation (3)	Equation (3)	Equation (3)

(a) Secret image (b) Sailboat (c) Boat

(d) Pepper (e) Bridge (f) Gold

Figure 4: The experimental images.

(b) *Experiment II.* The secret was transformed with an optimal substitution matrix and then embedded and extracted by means of simple LSB substitution. The optimal substitution matrix was constructed by EDE with PSNR as the fitness function.

(c) *Experiment III.* The secret was transformed with an optimal substitution matrix and then embedded and extracted by means of simple LSB substitution. The optimal substitution matrix was constructed by EDE with MSSIM as the fitness function.

As regards the proposed scheme, we transformed the secret with an optimal substitution matrix and then embedded it by the proposed ModEmbedding algorithm. The way to extract the secret is the same as that of simple LSB substitution. The optimal substitution matrix was constructed by EDE with MSSIM as the fitness function. For clarity, we use Table 1 to summarize the similarities and dissimilarities between the proposed scheme and the above simulations.

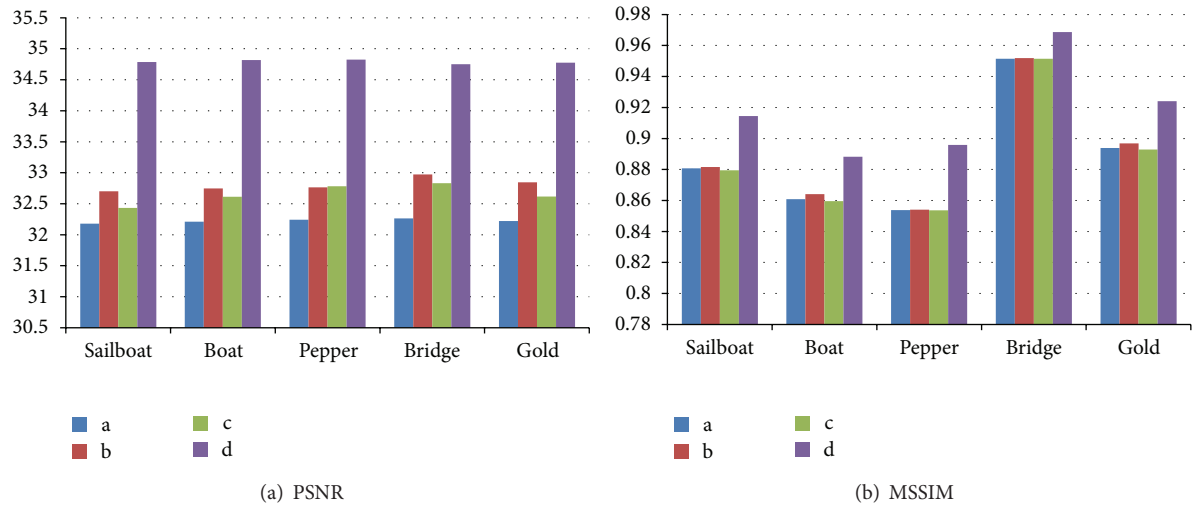
Table 2: Parameters of EDE.

Parameter	Value
Np	20
CR	0.8
F	0.7
DE strategy	DE/best/1/bin
Termination condition	Max generation > 50

Figure 4(a) is our secret image of 256 512 pixels, and Figures 4(b) to 4(f) are our cover-images of 512 × 512 pixels. The secret image is embedded into the last four significant bits of pixels of the cover-image (i.e., $k = 4$). The window size of MSSIM is 11 × 11. Table 2 lists parameters of EDE, and Table 3

Table 3: The PSNR and MSSIM of the three experiments and our method.

Image	(a) Experiment I		(b) Experiment II		(c) Experiment III		(d) Ours	
	PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM	PSNR	MSSIM
Sailboat	32.18033	0.88080	32.69998	0.88160	32.43443	0.87941	34.78432	0.91447
Boat	32.21223	0.86088	32.74649	0.86403	32.61401	0.85963	34.81854	0.88820
Pepper	32.24308	0.85380	32.76629	0.85406	32.78340	0.85370	34.82431	0.89590
Bridge	32.26443	0.95144	32.97276	0.95192	32.83314	0.95141	34.75165	0.96867
Gold	32.22283	0.89393	32.84528	0.89678	32.61744	0.89288	34.77397	0.92409



(a) PSNR (b) MSSIM

Figure 5: The bar charts of Table 3.

lists the PSNR and MSSIM values of the stego-images of the three simulated experiments and our method. To summarize, we sketch the bar chart of these values in Figure 5.

Several observations from these experimental results are discussed as follows.

(1) The proposed scheme outperforms simple LSB substitution (i.e., Experiment I) in both PSNR and MSSIM.

(2) As we have mentioned before, there are some researchers using heuristic algorithms to construct the near-optimal substitution matrix. Because we are not able to acquire their experimental results, we use Experiment II to simulate and compare. The MSSIM values indicate that our stego-images are more visually imperceptible than those of other researchers adopting PSNR as a fitness function. The PSNR values indicate that the improvement of the absolute difference between the cover-image and the stego-image is limited if the way of embedding is simple LSB substitution. Therefore, the proposed embedding algorithm indeed breaks through the limitation.

(3) We wonder what the result is if other researchers change their fitness function from PSNR to MSSIM. Therefore, we use Experiment III to simulate that situation. The MSSIM values indicate that our method performs better. We may, therefore, reasonably conclude that simple LSB substitution also limits the improvement of visual imperceptibility.

As a whole, the merits of our work are summarized as follows.

(1) The extra space for the substitution matrix is small.

The extra space for the substitution matrix is related to the number of bits, that is, the parameter k in our scheme, used to carry the secret. If $k = 4$, the required space is 16×16 bits. In Bedi et al.'s scheme [20], the extra space is related to the size of the cover-image. If the size of the cover-image is 512×512 pixels, the required space is 512×512 bits, which is 1024 times that of our scheme.

(2) The payload is high, but the image quality is not destroyed too much.

Generally speaking, the last four bits of a pixel can be modified at most; or else, the image quality is not acceptable. Hence the highest possible payload of a steganographic scheme is four bits per pixel. The experimental results show that our scheme achieves the highest payload, which is eight times that of Bedi et al.'s scheme. In addition, the average MSSIM of our scheme as shown in the experiments is 0.9183, while that of Bedi et al.' is 0.9124 according to their experimental results.

(3) We give consideration to image quality at pixel level and at visual level simultaneously.

The pervious researches related to optimal substitution matrix only consider the image quality at pixel level [6, 9, 10]. Our scheme takes the human visual system into account and adopts the measurement MSSIM as our fitness function. Besides, we elaborate the embedding algorithm so that the difference between the cover- and stego-images at pixel level is as small as possible. Though Bedi et al. also adopt MSSIM as their fitness function, the required space for the key is too large.

(4) Our extracting method is as simple as the simple LSB.

One of the merits of the simple LSB is its simple way of extracting the secret, that is, the modular operation. Like simple LSB, we extract the secret only through the modular operation.

5. Conclusions

As we have mentioned in Section 2.2, the number of possible solutions becomes 20, 922, 789, 888, 000 when $k = 4$. In this paper, we adopt EDE to construct a near-optimal substitution matrix. It follows from the experiment results that EDE can construct a good substitution matrix within a few iterations. Considering the features of human eyes, we adopt an HVS-based measurement MSSIM, instead of PSNR, as the fitness function. We can see from the experimental results that adopting MSSIM as the fitness function indeed improves imperceptibility visually. Besides, the proposed embedding algorithm improves the stego-image quality largely; at the same time, the extraction is as simple as by the traditional LSB substitution method. Many researchers utilize different methods to solve the problem of constructing an optimal substitution matrix, so we believe that this is an interesting problem. So far as we know, no one has attempted to apply discrete DE to solve this problem until now. Therefore, this paper provides an efficient method to construct a substitution matrix and extends the applications of the DE algorithm successfully.

In future work, we intend to address the issue of steganalysis [31]. We will design a sophisticated embedding strategy against statistical steganalysis. In addition, we may compare the results obtained from different bioinspired algorithms.

Acknowledgment

This work was supported in part by a Grant from the National Science Council of the Republic of China under Project NSC 102-2221-E-034-011-.

References

[1] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding," *IBM Systems Journal*, vol. 35, no. 3-4, pp. 313–336, 1996.

[2] N. F. Johnson and S. Jajodia, "Exploring steganography: seeing the unseen," *Computer*, vol. 31, no. 2, pp. 26–34, 1998.

[3] S. Katzenbeisser and F. A. P. Petitcolas, *Information Hiding Techniques for Steganography and Digital Watermarking*, Artech House, Boston, Mass, USA, 2000.

[4] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding—a survey," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1062–1078, 1999.

[5] C.-K. Chan and L. M. Cheng, "Hiding data in images by simple LSB substitution," *Pattern Recognition*, vol. 37, no. 3, pp. 469–474, 2004.

[6] R.-Z. Wang, C.-F. Lin, and J.-C. Lin, "Image hiding by optimal LSB substitution and genetic algorithm," *Pattern Recognition*, vol. 34, no. 3, pp. 671–683, 2001.

[7] J. H. Holland, *Adaptation in Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, Mich, USA, 1975.

[8] C. F. Tsai and C. W. Chen, "A new image hiding technique by optimal LSB substitution and Tabu search," in *Proceedings of National Computer Symposium*, pp. 161–171, Taichung City, Taiwan, 2007.

[9] C.-S. Hsu and S.-F. Tu, "Finding optimal LSB substitution using ant colony optimization algorithm," in *Proceedings of the 2nd International Conference on Communication Software and Networks (ICCSN '10)*, pp. 293–297, Singapore, February 2010.

[10] Z.-H. Wang, C.-C. Chang, and M.-C. Li, "Optimizing least-significant-bit substitution using cat swarm optimization strategy," *Information Sciences*, vol. 192, pp. 98–108, 2012.

[11] M. Barni and F. Bartolini, *Watermarking Systems Engineering Enabling Digital Assets Security and Other Applictions*, CRC Press, Boca Raton, Fla, USA, 2004.

[12] G. C. Onwubolu and D. Davendra, *Differential Evolution: A Handbook For Global Permutation-Based Combinatorial Optimization*, vol. 175 of *Studies in Computational Intelligence*, Springer, Berlin, Germany, 2009.

[13] R. Storn and K. Price, "Differential evolution-A simple efficient adaptive scheme for global optimization over continuous spaces," Tech. Rep., International Computer Science Institute, Berkeley, Calif, USA, 1995.

[14] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.

[15] M. Dorigo, *Optimization, learning and natural algorithms [Ph.D. thesis]*, Politecnico di Milano, Milan, Italy, 1992.

[16] S.-C. Chu and P.-W. Tsai, "Computational intelligence based on the behavior of cats," *International Journal of Innovative Computing, Information and Control*, vol. 3, no. 1, pp. 163–173, 2007.

[17] H. Xu, J. Wang, and H. J. Kim, "Near-optimal solution to pair-wise LSB matching via an immune programming strategy," *Information Sciences*, vol. 180, no. 8, pp. 1201–1217, 2010.

[18] J. Mielikainen, "LSB matching revisited," *IEEE Signal Processing Letters*, vol. 13, no. 5, pp. 285–287, 2006.

[19] I.-S. Lee and W.-H. Tsai, "Data hiding in grayscale images by dynamic programming based on a human visual model," *Pattern Recognition*, vol. 42, no. 7, pp. 1604–1611, 2009.

[20] P. Bedi, R. Banssal, and P. Sehgal, "Using PSO in a spatial domain based image hiding scheme with distortion tolerance," *Computers and Electrical Engineering*, vol. 39, pp. 640–654, 2013.

[21] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, pp. 1942–1948, December 1995.

[22] C.-H. Chou and Y.-C. Li, "Perceptually tuned subband image coder based on the measure of just-noticeable-distortion profile," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 5, no. 6, pp. 467–476, 1995.

[23] J. F. Delaigle, C. De Vleeschouwer, and B. Macq, "Watermarking algorithm based on a human visual model," *Signal Processing*, vol. 66, no. 3, pp. 319–335, 1998.

[24] J. F. Delaigle, C. D. Vleeschouwer, B. Macq, and I. Langendijk, "Human visual system features enabling watermarking," in *Proceedings IEEE International Conference on Multimedia and Expo (ICME '02)*, vol. 2, pp. 489–492, 2002.

[25] M. Kutter and S. Winkler, "A vision-based masking model for spread-spectrum image watermarking," *IEEE Transactions on Image Processing*, vol. 11, no. 1, pp. 16–25, 2002.

[26] W.-N. Lie and L.-C. Chang, "Data hiding in images with adaptive numbers of least significant bits based on the human visual system," in *Proceedings of the International Conference on Image Processing (ICIP '99)*, pp. 286–290, October 1999.

[27] D. G. Mayer, B. P. Kinghorn, and A. A. Archer, "Differential evolution—an easy and efficient evolutionary algorithm for model optimisation," *Agricultural Systems*, vol. 83, no. 3, pp. 315–328, 2005.

[28] G. C. Onwubolu, *New Optimization Techniques in Engineering*, Studies in Fuzziness and Soft Computing, Springer, Berlin, Germany, 2004.

[29] C.-C. Chang, M.-H. Lin, and Y.-C. Hu, "A fast and secure image hiding scheme based on LSB substitution," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 16, no. 4, pp. 399–416, 2002.

[30] C.-C. Chang, J.-Y. Hsiaob;, and C.-S. Chan, "Finding optimal least-significant-bit substitution in image hiding by dynamic programming strategy," *Pattern Recognition*, vol. 36, no. 7, pp. 1583–1595, 2003.

[31] B. Li, J. He, J. Huang, and Y. Q. Shi, "A survey on image steganography and steganalysis," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 2, no. 2, pp. 142–172, 2011.