IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received September 6, 2019, accepted September 22, 2019, date of publication September 25, 2019,
date of current version October 9, 2019.

Digital Object Identifier 10.1109/ACCESS.2019.2943616

A Mini-Sudoku Matrix-Based Data Embedding Scheme With High Payload

MINGZE HE[1], YANJUN LIU[2], CHIN-CHEN CHANG[2], (Fellow, IEEE), AND MINGXING HE[1]
[1]School of Computer and Software Engineering, Xihua University, Chengdu 610039, China
[2]Department of Information Engineering and Computer Science, Feng Chia University, Taichung 40724, Taiwan

Corresponding authors: Yanjun Liu (yjliu104@gmail.com) and Chin-Chen Chang (alan3c@gmail.com)

This work was supported in part by the Project of Chengdu Science and Technology Bureau under Grant 2016-XT00-00015-GX, in part by the Fund of Civil Aviation Administration of China under Grant PSDSA201802, in part by the Innovation Fund of Postgraduate of Xihua University under Grant ycjj2019022, in part by the Science and Technology Department of Sichuan Province under Grant 2017HH0083, and in part by the Project of Ministry of Science and Technology of Taiwan under Grant MOST 106-2221-E-035-013-MY3.

ABSTRACT Data hiding is a technology that generates meaningful stego media by embedding secret data in the cover media. In this paper, we propose a novel data hiding scheme using a mini-Sudoku matrix (MSM). A candidate block associated with the mapping value of a cover pixel pair on the MSM is used to indicate where the original pair would be modified to. According to the candidate block, each non-overlapping pixel pair can embed four bits of secret data. Among them, the first two bits are mapped from a specific element in the MSM while the other two are represented by the horizontal and vertical coordinates of this element with the modulus function. In addition, our MSM can be easily extended, and the proposed method can be used in other magic matrices to increase their embedding capacities. The experimental results indicate that our proposed scheme provides images with better visual quality than the previous methods. Furthermore, the security of the proposed scheme is verified by using pixel difference histogram (PDH) and RS steganalysis.

INDEX TERMS Data hiding, embedding capacity, mini-Sudoku matrix (MSM), visual quality.

I. INTRODUCTION

Digital information is transmitted on open networks, and increasing the security of digital information has become a major focus of research. One way to protect secret data and improve their security is to use encryption algorithms, such as the private key encryption technology DES [1] and the public key encryption technology RSA [2]. However, when an encryption algorithm is used, the data are encrypted in the form of scrambled code, which easily can be detected by malicious attackers, allowing them to intercept the secret data. Another way to protect secret data is to hide the data in an image to obtain a stego image. Then, the intended receiver can extract the secret information from the stego image. Stego images contain meaningful information but they can avoid being noticed by malicious attackers in the transmission process. Therefore, data hiding (DH) technology is used extensively in military and medical applications as well as in copyright protection and other fields. In recent decades, DH in images has attracted the attention of many researchers. In DH schemes, the embedding capacity (EC) and the visual quality of the stego image are two very important indicators that are used to measure the effectiveness of the schemes.

DH schemes for digital images are divided mainly into three types according to the domains, i.e., the compression domain, the frequency domain, and the spatial domain. For the compression domain, first, the cover image in the DH scheme is processed using compression algorithms, such as the Vector Quantization (VQ) [3], Absolute Moment Block Truncation Coding (AMBTC) [4], and the Joint Photographic Experts Group (JPEG) [5] algorithms. This allows the cover image to have more space to embed the secret data, but it does so at the cost of the visual quality of the stego image, which is a disadvantage of compression domain. In the frequency domain, the cover image is transformed either to Discrete Cosine Transform (DCT) coefficients [6], Discrete Wavelet Transform (DWT) coefficients [7], or Integer Wavelet Transform (IWT) coefficients [8]. The secret data can be embedded into the high frequency position because this location has a very small impact on the image. However, this limited position cannot embed much secret data, so the EC is relatively low compared with the DH scheme in the compression and spatial domains.

The associate editor coordinating the review of this manuscript and approving it for publication was Sudipta Roy.

Among the three domains, researchers have focused most of their work on the spatial domain, and the most commonly used DH scheme in this domain is based on the least significant bit (LSB) substitution [9]–[11]. The traditional LSB method was proposed first by Bender *et al.* [9], and it directly modifies the LSB of the pixels to embed the secret data. Later, in order to improve the quality of the image and the security of the secret data, Mielikainen [10] improved the LSB algorithm and embedded data in a pairwise manner by modifying their parity. Unfortunately, the LSB algorithms are exposed to RS steganalysis [11]. Some novel LSB-based DH schemes were proposed to further improve the embedding capacity and resist RS steganalysis [12]–[14]. Recently, certain DH schemes that combine pixel value differencing (PVD) and LSB to improve the visual quality of images have been proposed [15]–[17]. In 2003, Wu and Tsai [18] proposed a PVD-based DH scheme in which the amount of secret data that can be embedded is determined according to the difference between two adjacent pixels. In other words, the amount of embedded data will change dynamically with the pixel pair in the edge area or the smooth area, and the edge area can embed more secret bits. Subsequently, some optimizations of the DH schemes based on PVD were proposed [19]–[22]. In 2018, Sahu and Swain [23] proposed two PVD-based DH schemes using 1×5 pixel blocks.

In 2006, Zhang and Wang [24] presented the exploiting modification direction (EMD) DH scheme in order to improve the visual quality of the image in LSB substitution. The EMD algorithm can embed $\log_2(2n+1)$ bits into n cover pixels, and each embedding requires only one pixel to increase or decrease by one. Inspired by the EMD scheme, various extended DH schemes based on EMD also have been proposed [25], [26]. Subsequently, some research works have been conducted on magic matrix-based DH schemes that derived from the EMD method. In 2008, Chang *et al.* [27] proposed a novel DH scheme using a 256×256 Sudoku matrix that was composed of several 3×3 traditional Sudoku grids. In this scheme, the secret message is converted into secret digits in the base-9 numeral system, and the embedding rate (ER) can be as high as 1.5 bits per pixel (bpp). Subsequently, the Sudoku matrix was extended for use in various DH schemes, such as the 3D-Sudoku DH scheme [28] and the reversible DH scheme [29]. In 2014, a new DH scheme based on the turtle shell (TS) matrix was proposed by Chang *et al.* [30]. The TS matrix is composed of eight numbers ranging from 0 to 7, so the secret data can be processed by a string of 3-bit segments, without the need to transform it into a specific numeral system. In 2016, Liu *et al.* [31] used a location table to record the special location of elements in the TS matrix, and this improved the EC. Then, Jin *et al.* [32] applied the swarm optimization algorithm to the TS matrix to improve the visual quality of the image. Subsequently, many researchers have studied the DH schemes based on the TS matrix [33], [34].

Inspired by Liu *et al.*'s scheme [31], in this paper, we propose a novel DH scheme based on the mini-Sudoku matrix (MSM). The main contributions of the proposed scheme lie in: first, the MSM is used as the magic matrix for the first time to propose a data hiding scheme. The MSM is a 256×256 matrix consisting of a great number of quaternary digits, each of which is represented by two bits. Second, we use not only the digits in the MSM, but also the coordinate axes as part of the secret data for embedding. Thus, from the MSM, two more bits can be obtained by the horizontal and vertical coordinates modulo 2 in such a way that each non-overlapping pixel pair can embed four bits of secret data. Third, the proposed scheme is superior to other state-of-the-art schemes in visual quality of images under the same embedding capacity. Last, the universal principle of construction makes the proposed scheme extensible. In particular, the embedding method can be applied to other magic matrices, such as the TS matrix, the traditional Sudoku matrix, or other extended matrices, to increase the embedding capacity without redesigning a new magic matrix.

The rest of this paper is organized as follows. Three previous DH schemes are reviewed briefly in Section II, and our proposed scheme is explained in detail in Section III. The experimental results are discussed in Section IV, and Section V presents our conclusions.

II. RELATED WORK

In recent years, there have been many DH schemes based on magic matrix. DH schemes based on TS matrix and DH schemes based on EMD are research hotspots. In order to facilitate the understanding of our proposed scheme, first, we review Chang *et al.*'s [30] and Liu *et al.*'s [31] schemes based on the TS matrix. Subsequently, we briefly review Kim *et al.*'s [25] and Xie *et al.*'s [26] schemes based on EMD. These schemes are described briefly below.

A. CHANG ET AL.'S SCHEME

In 2014, the first TS-based scheme for data hiding was introduced by Chang *et al.* [30]. Fig. 1 shows that the size of the TS matrix, M, is 256×256, and the elements in M from 0 to 7 indicate secret digits that will be embedded. Each turtle shell is defined as a hexagon that contains eight different digits, among which the two digits inside the turtle shell are called the back digits, and the other six ones on the boundary of the turtle shell are called the edge digits. To construct a TS matrix, the following two rules must be used, i.e., 1) adjacent elements in the same row are increased by "1" from left to right and defined as

$$M(p_i + 1, p_j) = (M(p_i, p_j) + 1) \bmod 8, \tag{1}$$

and 2) adjacent elements in the same column are increased cyclically by "2" and "3" from the bottom to the top and defined as

$$M(p_i, p_j + 1) = \begin{cases} (M(p_i, p_j) + 3) \bmod 8 \text{ if } p_j \text{ is an odd number,} \\ (M(p_i, p_j) + 2) \bmod 8 \text{ if } p_j \text{ is an even number.} \end{cases} \tag{2}$$

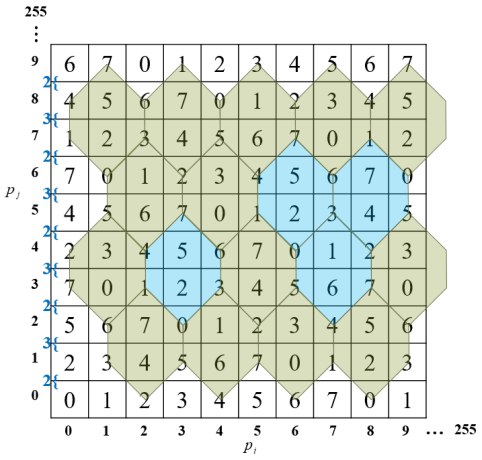


FIGURE 1. The architecture of the TS matrix.

The experimental results indicate that Chang *et al.*'s scheme has small distortion, low computational cost, and the maximum ER can reach 1.5 bpp.

B. LIU ET AL.'S SCHEME

Liu *et al.*'s scheme [31] is also based on the TS matrix. It improves the embedding capacity over that of Chang *et al.*'s scheme [30], and the distortion of the image is low. In Liu *et al.*'s scheme, a TS matrix M (see Fig. 1) and a location table T (see Fig. 2) are constructed in order to hide data. Each non-overlapping pixel pair in the cover image can embed 4 bits of the secret message, so the string of the secret message to be embedded is split into a number of pieces with 4 bits.

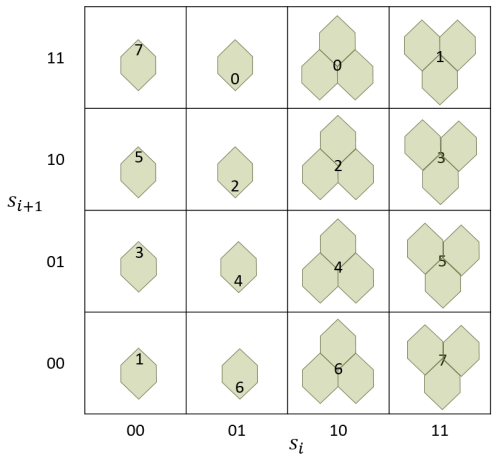


FIGURE 2. The location table T.

Fig. 2 shows that a location table T consists of 16 special locations of elements on the TS matrix which place all the to-be-embedded 4-bit secret message that will occur. The values of the elements on the upper back of a single turtle shell are always in the set of {1, 3, 5, 7}, as shown in the first column in Fig. 2; those on the lower back of a single turtle shell are always in the set of {0, 2, 4, 6}, as shown in the second column in Fig. 2. In addition, the elements on the intersection of three adjoining turtle shells are divided into two categories, i.e., those located in an upward rocket-like shape are always in the set of {0, 2, 4, 6}, and those located in a downward rocket-like form are always in the set of {1, 3, 5, 7}.

The location table T is generated by the positional characteristics of the turtle shell. Each location in T is denoted as $T(s_i, s_{i+1})$ with horizontal and vertical coordinates (s_i, s_{i+1}), where s_i and s_{i+1} belong to {00, 01, 10, 11}. This implies that a 4-bit secret message pair, (s_i, s_{i+1}), is hidden in the position of $T(s_i, s_{i+1})$ on the TS matrix. For example, if the 4-bit secret message is $(1100)_2$, first, we divide it into two groups, i.e., $s_1 = 11$ and $s_2 = 00$. Then, on the location table T we can get a digit "7" at the location $T(11, 00)$ that is in a downward rocket-like form of three turtle shells on the TS matrix. The embedding and extraction procedures are presented in detail below.

After the TS matrix M and the location table T have been generated, we divide the cover image with size of $W \times H$ into non-overlapping pixel pairs (p_j, p_{j+1}), where $j \in \{1, 3, \cdots, W \times H - 1\}$. To embed 4-bit secret, (s_i, s_{i+1}), into the pixel pair, (p_j, p_{j+1}), we must follow two embedding rules:

Rule 1: If $M(p_j, p_{j+1})$ is a regular element, determine the shape of 9 adjoining turtle shells in the TS matrix where it is located in the center, as shown in Fig. 3. Four potential positions of regular elements are illustrated by the blue dots. Then, find all candidate pixel pairs from this selected 9-TS shape that equal the secret message value, (s_i, s_{i+1}), in the location table T. Finally, select the pixel pair, (p'_j, p'_{j+1}), having the shortest Euclidean distance with (p_j, p_{j+1}) as the stego pixel pair.

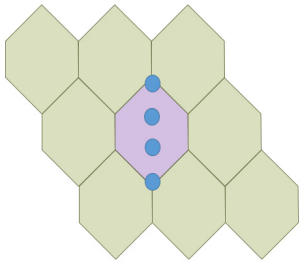


FIGURE 3. Nine candidate turtle shells for regular elements.

Rule 2: If $M(p_j, p_{j+1})$ is not a regular element, find the nearest 9 adjoining turtle shells in the TS matrix and execute Rule 1.

By the above embedding rules that combine the TS matrix and the location table, the secret message can be embedded successfully each time. The extraction procedure is quite simple, i.e., the secret message can be extracted by the location of the pixel pair (p'_j, p'_{j+1}) in the TS matrix.

Liu *et al.*'s scheme increases the capacity for embedding secret messages, and the maximum ER is 2 bpp. The quality

of the image is better than that by using Chang *et al.*'s scheme when the same amount of data are embedded.

C. ZHANG AND WANG'S SCHEME

The EMD-based scheme for data hiding was introduced by Zhang and Wang [24]. In the embedding phase, each secret digit in a $(2n+1)$-ary notational system is carried by n cover pixels such that at most one cover pixel is increased or decreased by 1. A cover image is first divided into a series of non-overlapping pixel groups. The gray values of pixels in a group are represented by $p_1, p_2, \ldots, p_n$ and mapped to a value f, which is defined as

$$f(p_1, p_2, \ldots p_n) = \left[\sum_{i=1}^{n} p_i \cdot i\right] \bmod (2n+1). \quad (3)$$

If the secret message d to be embedded is equal to the value f, there is no need to modify the pixel value; otherwise, it is necessary to calculate $s = d - f \bmod (2n+1)$. When $s \leq n$, the gray value of p_s is increased by 1; otherwise, the gray value of p_{2n+1-s} is decreased by 1. When the above steps are repeated until the last group is processed, all the secret messages are embedded in the cover image.

D. XIE ET AL.'S SCHEME

In 2019, an extended squared magic matrix (ESMM)-based DH scheme was introduced by Xie *et al.* [26]. Fig. 4 shows an arbitrarily drawn 9×9 block in ESMM that is composed of nine 3×3 sub-blocks. The aim of employing ESMM is to use 81 different numbers from 0 to 80 to fill a 9×9 block, as shown in the red square in Fig. 4.

255	6	7	8	15	16	17	24	25	26	33	34	35	…	35
…	…	…	…	…	…	…	…	…	…	…	…	…	…	…
11	6	7	8	15	16	17	24	25	26	33	34	35	…	8
10	3	4	5	12	13	14	21	22	23	30	31	32	…	5
9	0	1	2	9	10	11	18	19	20	27	28	29	…	2
8	60	61	62	69	70	71	78	79	80	6	7	8	…	62
7	57	58	59	66	67	68	75	76	77	3	4	5	…	59
6	54	55	56	63	64	65	72	73	74	0	1	2	…	56
5	33	34	35	42	43	44	51	52	53	60	61	62	…	35
4	30	31	32	39	40	41	48	49	50	57	58	59	…	32
3	27	28	29	36	37	38	45	46	47	54	55	56	…	29
2	6	7	8	15	16	17	24	25	26	33	34	35	…	8
1	3	4	5	12	13	14	21	22	23	30	31	32	…	5
0	0	1	2	9	10	11	18	19	20	27	28	29	…	2
	0	1	2	3	4	5	6	7	8	9	10	11	…	255

FIGURE 4. The extended squared magic matrix.

Before embedding, the ESMM is generated and the binary secret stream S is converted into an 81-ary stream S'. In the embedding phase, a pixel pair (p_j, p_{j+1}) is replaced by (p'_j, p'_{j+1}) to embed one 81-ary digit when $ESMM\ (p'_j, p'_{j+1})$ is equal to this digit.

Xie *et al.*'s scheme is a generalized data hiding method based on magic matrix to enlarge embedding capacity by enlarging the digit range of the magic matrix. Although the maximum embedded capacity of Xie *et al.*'s scheme is large, experiments show that its PSNR value is much lower than those of the previous magic matrix-based schemes under the same embedding capacity.

Our scheme proposed later in Section III is different from Xie *et al.*'s scheme. In our proposed scheme, the secret data is divided into two parts, one part is hidden in the digits of the magic matrix, and the other part is hidden in the coordinates of the matrix. In addition, our embedding method can steadily increase the ER value by 1 bpp without enlarging the digit range of magic matrix. If our embedding method is applied to Xie *et al.*'s ESMM, it would not need 81 different numbers to construct the matrix, but only 16 different numbers to achieve nearly the same embedding capacity. In the following, our scheme is presented in detail.

III. THE PROPOSED SCHEME

In order to increase the embedding capacity of the data hiding schemes based on the magic matrix, most scholars have enlarged the values of the elements in the magic matrix, that is, the base of the numeral system for the to-be-embedded secret digits is increased. Unfortunately, this is done at the expense of visual quality, such as Xie *et al.*'s scheme [26]. In this section, we propose a data hiding scheme based on the mini-Sudoku matrix (MSM) to improve the visual quality of stego image further while guaranteeing satisfactory embedding capacity. In the proposed scheme, a candidate block associated with the mapping value of a cover pixel pair on the MSM is used to indicate where the original pair would be modified to. According to the candidate block, each non-overlapping pixel pair can embed four bits of secret data, two bits of which are mapped from a specific element in the MSM, with the other two bits represented by the horizontal and vertical coordinates of this element with the modulus function.

Assume that a binary secret message stream S with n bits is denoted as $S = \{s_i | i = 1, 2, \cdots, n\}$ and assume that a grayscale cover image P with the size of $H \times W$ is denoted as $P = \{p_j | j = 1, 2, \cdots, H \times W\}$. For security considerations, before embedding, secret messages are always encrypted with private-key encryptions, such as DES or AES [35], or with public-key encryptions, such as RSA. The proposed scheme consists of two phases, i.e., (1) the secret message embedding phase and (2) the secret message extraction phase. Before addressing the detailed phases, we shall introduce the mini-Sudoku matrix that is used as a main building block in the proposed scheme.

A. CONSTRUCTION OF THE MINI-SUDOKU MATRIX

Sudoku [27] is a logical number placement puzzle that was created by Garns in 1979 and it became a household puzzle game in 2005. The aim of Sudoku is to use nine different numbers from 1 to 9 to fill a 9×9 grid. Inspired by the conventional Sudoku, we now use the mini-Sudoku [36] to

design a novel magic matrix that will guide the embedding process of our proposed data hiding scheme.

The objective of the mini-Sudoku is to fill a 4×4 grid through digits from 0 to 3. The red square frame in Fig. 5 illustrates a solution of such a mini-Sudoku grid, which has the following characteristics. (1) The mini-Sudoku grid is a 4×4 block that contains four 2×2 sub-grids. (2) Each sub-grid consists of different digits from 0 to 3. (3) The digits from 0 to 3 must occur just once in each row and each column of the mini-Sudoku grid.

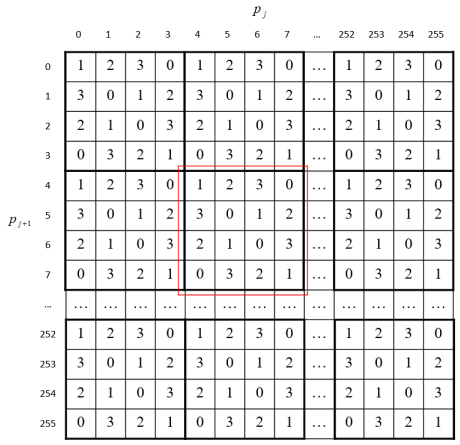


FIGURE 5. An example of the mini-Sudoku matrix (MSM).

As shown in Fig. 5, a pre-selected mini-Sudoku solution is used to construct a mini-Sudoku matrix (MSM). The coordinates on the horizontal and vertical axes of the MSM are the values of pixel pairs ranging from 0 to 255. Thus, the MSM is a 256×256 matrix that contains 4096 tiled mini-Sudoku grids. It is noted that each element of the MSM indicates a quaternary secret digit to be embedded. Therefore, the functionality of the MSM is to make slight modifications on the currently processed cover pixel pair to embed a quaternary secret digit with minimum distortions.

However, if we do not specify the MSM solution to be used, each pixel pair can embed only a 2-bit secret message, leading to an embedding rate at just 1 bpp. Since the elements in the MSM are highly repetitive, it is better to select those solutions that consider the horizontal and vertical coordinates of nearby elements with the same value. Assume that the element in MSM located at the horizontal coordinate of p_j and the vertical coordinate of p_{j+1} is denoted as $MSM\left(p_j, p_{j+1}\right)$. The selected solutions must satisfy an extra requirement such that the values of the digits, $MSM\left(p_j, p_{j+1}\right)$, $MSM\left(p_j+2, p_{j+1}\right)$, $MSM\left(p_j, p_{j+1}+2\right)$, and $MSM\left(p_j+2, p_{j+1}+2\right)$, cannot be the same in a 4×4 grid, where p_j and p_{j+1} range from 0 to 253. By this strategy, the embedding rate of the proposed scheme can reach up to 2 bpp.

B. SECRET MESSAGE EMBEDDING PHASE

The core idea of the embedding phase of the proposed scheme is to change the coordinate positions in the MSM of the cover pixel pairs according to the to-be-embedded secret messages for minimum distortions. Suppose that the currently processed cover pixel pair is denoted as $\left(p_j, p_{j+1}\right)$, which can embed four secret bits by the way below: first, two bits are concealed through the candidate block associated with the mapping value of this pair on the MSM, and then, the third bit is hidden by the horizontal coordinate p_j modulo 2 and the fourth bit by the vertical coordinate p_{j+1} modulo 2. The binary secret stream, denoted as S, is divided into a sequence of 4-bit segments, each is represented by $sg_d = \{s_{4d-3}, s_{4d-2}, s_{4d-1}, s_{4d}\}$, where $d \in \left\{1, 2, \cdots, \left\lceil \frac{n}{4} \right\rceil\right\}$. The details of the embedding phase are elaborated as follows:

Step 1: Construct an MSM with the size of 256×256 according to the rules described in Subsection III.A.

Step 2: Divide the grayscale cover image P with the size of $H \times W$ into non-overlapping pixel pairs $\left(p_j, p_{j+1}\right)$, where $j \in \{1, 3, \cdots, H \times W - 1\}$.

Step 3: For a cover pixel pair, $\left(p_j, p_{j+1}\right)$, read a 4-bit segment, $sg_d = \{s_{4d-3}, s_{4d-2}, s_{4d-1}, s_{4d}\}$ for $d = (j+1)/2$, from the binary secret stream S. Divide sg_d into three groups, i.e., $B_1 = s_{4d-3}s_{4d-2}$, $B_2 = s_{4d-1}$, and $B_3 = s_{4d}$. Thus, B_1 is a quaternary digit from 0 to 3 and B_2/B_3 is a bit of 0 or 1.

Step 4: Map the pair, $\left(p_j, p_{j+1}\right)$, onto the element, $MSM\left(p_j, p_{j+1}\right)$, in the MSM.

Step 5: Embed the secret segment sg_d into the pair $\left(p_j, p_{j+1}\right)$. To embed sg_d, a 4×4 candidate block G for $MSM\left(p_j, p_{j+1}\right)$ is used to indicate where $\left(p_j, p_{j+1}\right)$ would be modified to. First, determine the scope of G by (4), as shown at the top of the next page. For example, the blue square shown in Fig. 6 is the candidate block G derived from (4) for the mapping value, $MSM\,(1, 3)$, of the pixel pair, (1, 3). Then, find the element, $MSM\left(p'_j, p'_{j+1}\right)$, within G satisfying (1) $MSM\left(p'_j, p'_{j+1}\right) = B_1$, (2) $p'_j \bmod 2 = B_2$, and (3) $p'_{j+1} \bmod 2 = B_3$. Ultimately, modify the pair $\left(p_j, p_{j+1}\right)$ to $\left(p'_j, p'_{j+1}\right)$.

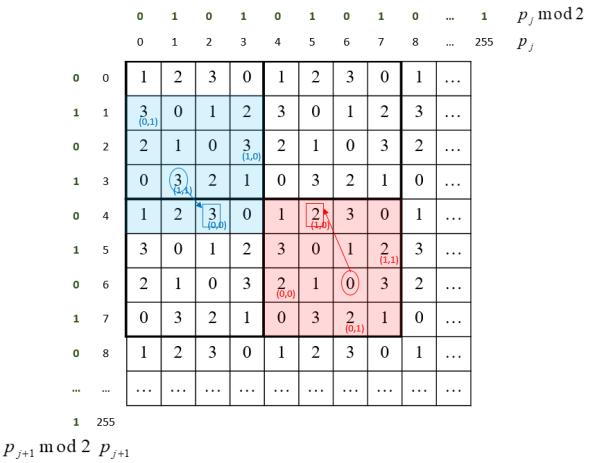


FIGURE 6. Examples of the embedding process in the proposed scheme.

Step 6: Repeat Steps 3-5 until all of the secret messages are embedded or the last cover pixel pair is executed. Afterwards, a stego image P' is generated and sent to the receiver. It is

$$G = \begin{cases} MSM\ (0:3, 0:3) & \text{if } p_j \le 1 \ \&\ p_{j+1} \le 1 \\ MSM\ \left(0:3, p_{j+1}-2 : p_{j+1}+1\right) & \text{if } p_j \le 1 \ \&\ 1 < p_{j+1} < 255 \\ MSM\ (0:3, 252:255) & \text{if } p_j \le 1 \ \&\ p_{j+1} = 255 \\ MSM\ \left(p_j-2 : p_j+1, 0:3\right) & \text{if } 1 < p_j < 255 \ \&\ p_{j+1} \le 1 \\ MSM\ \left(p_j-2 : p_j+1, p_{j+1}-2 : p_{j+1}+1\right) & \text{if } 1 < p_j < 255 \ \&\ 1 < p_{j+1} < 255 \\ MSM\ \left(p_j-2 : p_j+1, 252:255\right) & \text{if } 1 < p_j < 255 \ \&\ p_{j+1} = 255 \\ MSM\ (252:255, 0:3) & \text{if } p_j = 255 \ \&\ p_{j+1} \le 1 \\ MSM\ \left(252:255, p_{j+1}-2 : p_{j+1}+1\right) & \text{if } p_j = 255 \ \&\ 1 < p_{j+1} < 255 \\ MSM\ (252:255, 252:255) & \text{otherwise.} \end{cases} \tag{4}$$

noted that there is no need to deliver the MSM as auxiliary information to the receiver; the receiver should establish the same MSM on his side to retrieve the embedded secret messages.

In the following, two examples are illustrated to explain the embedding process of our proposed scheme further. Given two cover pixel pairs (6, 6) and (1, 3), we will address how to use them to embed the binary secret stream $(1010\,1100)_2$. As shown in Fig. 6, the circles and the squares represent the locations of cover pixel pairs and stego pixel pairs in the MSM, respectively. The MSM is generated by Step 1 before embedding and the secret stream is split into two 4-bit pieces $(1010)_2$ and $(1100)_2$.

Example 1: Embed the first secret piece $(1010)_2$ in the pixel pair (6, 6)

First, divide the secret piece $(1010)_2$ into three groups, i.e., $B_1 = (10)_2 = 2$, $B_2 = 1$, and $B_3 = 0$. Second, map the pair (6, 6) onto the value of $MSM\ (6, 6)$ in the MSM. Then, determine the candidate block G (see the 4×4 red square in Fig. 6) for $MSM\ (6, 6)$ according to (4), in which G is set to $G = MSM\ (4:7, 4:7)$ since the fifth condition is met. Find an element $MSM\ (5, 4)$ within G that satisfies $MSM\ (5, 4) = 2 = B_1$, $5 \bmod 2 = 1 = B_2$, and $4 \bmod 2 = 0 = B_3$, simultaneously. As a result, the pixel pair (6, 6) is modified to (5, 4) for concealing the secret piece $(1010)_2$.

Example 2: Embed the second secret piece $(1100)_2$ in the pixel pair (1, 3)

First, divide the secret piece $(1100)_2$ into three groups, i.e., $B_1 = (11)_2 = 3$, $B_2 = 0$, and $B_3 = 0$. Second, map the pair (1, 3) onto the value of $MSM\ (1, 3)$ in the MSM. Then, determine the candidate block G (see the 4×4 blue square in Fig. 6) for $MSM\ (1, 3)$ according to (4). Here, the second condition in (4) is satisfied to get $G = MSM\ (0:3, 1:4)$. Find an element $MSM\ (2, 4)$ within G that satisfies $MSM\ (2, 4) = 3 = B_1$, $2 \bmod 2 = 0 = B_2$, and $4 \bmod 2 = 0 = B_3$. Consequently, the pixel pair (1, 3) is modified to (2, 4) for embedding the secret piece $(1100)_2$.

C. SECRET MESSAGE EXTRACTION PHASE

Upon obtaining the stego image P', the receiver first divides it into non-overlapping pixel pairs $\left(p'_j, p'_{j+1}\right)$, where $j \in \{1, 3, \cdots, H \times W - 1\}$. Afterwards, a 4-bit secret piece is extracted by the steps as follows: (1) get the mapping value $B'_1 = MSM\left(p'_j, p'_{j+1}\right)$ of $\left(p'_j, p'_{j+1}\right)$ in the MSM and convert it into two binary bits; (2) retrieve one bit B' by $B'_2 = p'_j \bmod 2$; (3) obtain another bit B'_3 by $B'_3 = p'_{j+1} \bmod 2$; and (4) concatenate B'_1, B'_2, and B'_3 to form the embedded secret piece. Subsequently, the whole secret message is retrieved exactly when all of the pieces are put together.

In the following, the extraction process of Example 1 that was described in Section III-B is given. Given that the current stego pixel pair $\left(p'_j, p'_{j+1}\right)$ is (5, 4), we can immediately get that $B'_1 = MSM\ (5, 4) = 2 = (10)_2$, $B'_2 = 5 \bmod 2 = 1$, and $B'_3 = 4 \bmod 2 = 0$.Then, the embedded 4-bit secret piece $(1010)_2$ is concatenated by B'_1, B'_2, and B'_3.

IV. EXPERIMENTAL RESULTS

In this section, we demonstrate the results of the experiments that are conducted by our proposed scheme, and we compare the results with those of several previous schemes. The 8 standard 512×512 grayscale images in Fig. 7, i.e., Lena, Boat, Airplane, Elaine, Goldhill, Peppers, Baboon, and Sailboat, are tested. The binary secret message S is generated pseudo-randomly and encrypted by the encryption algorithm before the embedding process. All experiments are implemented by MATLAB R2012b. Fig. 8 shows the 8 stego images after the embedding process.

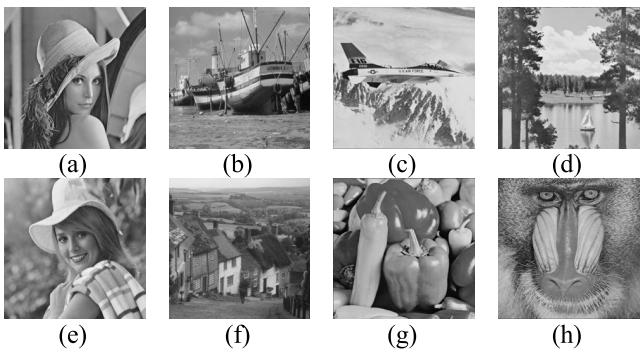


FIGURE 7. Eight test images with the sizes of 512×512. (a) Lena. (b) Boat. (c) Airplane. (d) Sailboat. (e) Elaine. (f) Goldhill. (g) Peppers. (h) Baboon.

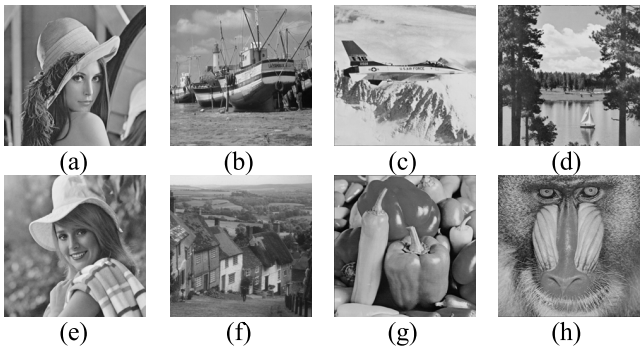


FIGURE 8. Eight resulted stego images with the sizes of 512 × 512. (a) Lena. (b) Boat. (c) Airplane. (d) Sailboat. (e) Elaine. (f) Goldhill. (g) Peppers. (h) Baboon.

The peak signal-to-noise ratio (PSNR) is used to evaluate the quality of the image. The definition of PSNR is given by

$$PSNR = 10\log_{10}\left(\frac{255^2}{MSE}\right), \tag{5}$$

where MSE is the mean square error between the grayscale cover image and the stego image. MSE is defined as

$$MSE = \frac{1}{H \times W}\sum_{i=1}^{H}\sum_{j=1}^{W}\left(P(i,j) - P'(i,j)\right)^2, \tag{6}$$

where $P(i,j)$ is the pixel value of the cover image, $P'(i,j)$ is the pixel value of the corresponding stego image, and H and W are the height and the width of the image.

In addition to the PSNR, the structural similarity (SSIM) and the quality index (QI) are used as two criteria to measure the quality of the images. The SSIM values ranged from 0 to 1, where the value of 1 indicates that the two images are the same. The SSIM is calculated between the block x of the cover image and the corresponding block y of the stego image by

$$SSIM(x,y) = \frac{\left(2\mu_x\mu_y + c_1\right)\left(2\sigma_{xy} + c_2\right)}{\left(\mu_x^2 + u_y^2 + c_1\right)\left(\sigma_x^2 + \sigma_y^2 + c_2\right)}, \tag{7}$$

where μ_x and μ_y are the averages of x and y, respectively; σ_x^2 and σ_y^2 are the variances of x and y, respectively; and σ_{xy} is the variance of x and y. Here, c_1 and c_2 are calculated by (8) and (9), respectively.

$$c_1 = (k_1 L)^2, \tag{8}$$
$$c_2 = (k_2 L)^2, \tag{9}$$

where L is the dynamic range of the pixel values, $k_1 = 0.01$, and $k_2 = 0.03$.

The QI is calculated as

$$QI = \frac{4\mu_1\mu_2\alpha_{12}}{\left(\alpha_1^2 + \alpha_2^2\right)\left(\mu_1^2 + \mu_2^2\right)}, \tag{10}$$

where μ_1 and μ_2 are the averages of P and P', respectively, and other parameters are defined as

$$\alpha_{12} = \frac{1}{H \times W - 1}\sum_{i=1}^{H}\sum_{j=1}^{W}\left(P(i,j) - \mu_1\right)\left(P'(i,j) - \mu_2\right),$$

$$\alpha_1^2 = \frac{1}{H \times W - 1}\sum_{i=1}^{H}\sum_{j=1}^{W}\left[P(i,j) - \mu_1\right]^2,$$

$$\alpha_2^2 = \frac{1}{H \times W - 1}\sum_{i=1}^{H}\sum_{j=1}^{W}\left[P'(i,j) - \mu_2\right]^2.$$

A higher value of QI means lower distortions of the image.

In the following, the embedding capacity (EC) is defined as the total secret bits that can be embedded into the cover image, while the embedding rate (ER) indicates bit-per-pixel (bpp), i.e., the number of secret bits that can be embedded in a single pixel. The ER is defined as

$$ER = \frac{\| S \|}{H \times W}, \tag{11}$$

where $\| S \|$ is the number of embedded secret bits.

Table 1 shows the performance of our proposed scheme. The average PSNR value is still higher than 46 dB when ER reaches the maximum value of 2 bpp. Since the size of the image is 512×512 and ER is 2 bpp, the total EC of each image is calculated by $512 \times 512 \times 2 = 524,288$. The values from the columns of bit error rate (BER) implies that no bit error occurs in a unit time after embedding. The average of SSIM is 0.9919 and the average of QI is 0.9997 in our proposed scheme. The average of the embedding efficiency (EF) is 25% in our proposed scheme. In addition, by the given EC value the PSNR values of all of the images that are tested remained basically stable, i.e., there are no excessive changes. The reason for this is that the change range is determined by the MSM and the embedding algorithm, so the PSNR value is irrelevant to the cover image.

TABLE 1. Experimental results of the proposed scheme.

Image	EC(bits)	PSNR(dB)	BER	SSIM	QI	EF (%)
Lena	524,288	46.37	0	0.9902	0.9997	25
Boat	524,288	46.36	0	0.9930	0.9997	25
Airplane	524,288	46.37	0	0.9889	0.9997	25
Elaine	524,288	46.38	0	0.9924	0.9997	25
Goldhill	524,288	46.37	0	0.9933	0.9997	25
Peppers	524,288	46.37	0	0.9906	0.9998	25
Baboon	524,288	46.36	0	0.9968	0.9996	25
Sailboat	524,288	46.35	0	0.9926	0.9998	25
Average	524,288	46.37	0	0.992	0.9997	25

In order to clearly illustrate the advantages of our scheme, Table 2 compares our scheme and several previous schemes, and it indicates that both the EC and PSNR of our proposed scheme are higher than those of the other three schemes [19], [20], [31]. More specifically, the proposed scheme has the

TABLE 2. Comparisons on EC and PSNR of the proposed scheme and other schemes.

Image	Shen and Huang [20]		Chen [19]		Liu et al. [31]		Proposed	
	EC	PSNR	EC	PSNR	EC	PSNR	EC	PSNR
Lena	402,485	42.46	512,384	43.01	524,288	45.55	524,288	46.37
Peppers	404,226	41.25	512,392	43.63	524,288	45.54	524,288	46.37
Baboon	443,472	38.88	512,516	43.58	524,288	45.55	524,288	46.36
Sailboat	411,306	41.29	524,508	42.86	524,288	45.55	524,288	46.35
Elaine	398,250	42.98	493,520	43.41	524,288	45.54	524,288	46.38
Airplane	413,556	42.33	469,320	43.66	524,288	45.58	524,288	46.37
Boat	408,777	41.60	483,516	43.63	524,288	45.55	524,288	46.36
Goldhill	405,956	41.80	523,882	42.12	524,288	45.49	524,288	46.37
Average	411,003	41.57	504,004	43.24	524,288	45.54	524,288	46.37

best average PSNR, followed by Liu *et al.*'s scheme [31] with the average PSNR of 45.54 dB. The average PSNR obtained by Shen and Huang's scheme [20] and Chen's scheme [19] are just 43.24 dB and 41.57 dB, respectively.

Table 3 compares the PSNR values of our proposed scheme with three other magic matrix-based schemes [26], [31], [32] under maximum EC. In particular, Xie *et al.*'s scheme has the lowest average PSNR value of 41.87 dB. The average PSNR value of our proposed scheme is more than 4 dB higher than that of Xie *et al.*'s scheme [26]. It can be inferred that the proposed scheme has the highest PSNR value when all of the four schemes have the same maximum EC at 524,288 bits. This is because different magic matrix-based schemes have different sizes of candidate areas in the embedding phase, and smaller areas can lead to lower visual distortions. The candidate areas used in our proposed scheme are limited to 4×4 squares as shown in Fig. 6, so higher PSNR values can be obtained.

TABLE 3. Comparisons among four magic matrix-based schemes.

Image	Jin et al. [32]		Liu et al. [31]		Xie et al. [26]		Proposed	
	EC	PSNR	EC	PSNR	EC	PSNR	EC	PSNR
Lena	524,288	45.57	524,288	45.55	524,288	41.87	524,288	46.37
Peppers	524,288	45.56	524,288	45.54	524,288	41.86	524,288	46.37
Baboon	524,288	45.57	524,288	45.55	524,288	41.87	524,288	46.36
Sailboat	524,288	45.58	524,288	45.55	524,288	41.86	524,288	46.35
Elaine	524,288	45.56	524,288	45.54	524,288	41.87	524,288	46.38
Airplane	524,288	45.57	524,288	45.58	524,288	41.87	524,288	46.37
Boat	524,288	45.58	524,288	45.54	524,288	41.86	524,288	46.36
Goldhill	524,288	45.49	524,288	45.49	524,288	41.87	524,288	46.37
Average	524,288	45.56	524,288	45.54	524,288	41.87	524,288	46.37

Table 4 shows the simulation performance of Chang *et al.*'s scheme [27], Xia *et al.*'s scheme [28], and our proposed scheme in terms of EC and PSNR. Although all of the three schemes use the Sudoku matrix, our proposed scheme can carry more secret messages than Chang *et al.*'s scheme.

TABLE 4. Comparisons among Sudoku-based schemes.

Image	Chang et al. [27]		Xia et al. [28]		Proposed	
	EC	PSNR	EC	PSNR	EC	PSNR
Lena	393,216	44.96	524,288	41.31	524,288	46.37
Peppers	393,216	44.67	524,288	41.30	524,288	46.37
Baboon	393,216	44.68	524,288	41.25	524,288	46.36
Sailboat	393,216	44.67	524,288	41.32	524,288	46.35
Elaine	393,216	44.92	524,288	41.31	524,288	46.38
Airplane	393,216	44.99	524,288	41.28	524,288	46.37
Boat	393,216	44.90	524,288	41.23	524,288	46.36
Goldhill	393,216	44.85	524,288	41.29	524,288	46.37
Average	393,216	44.83	524,288	41.29	524,288	46.37

In addition, the visual quality of the stego images in our scheme is better than those of both Chang *et al.*'s and Xia *et al.*'s schemes. Intuitively, our proposed method has less matrix construction cost than Xia *et al.*'s scheme since the Sudoku matrix used in our proposed method is of two dimensions rather than that of three dimensions used in Xia *et al.*'s scheme. Thus, we can say that our proposed scheme significantly outperforms the two scheme based on Sudoku matrix.

Fig. 9 shows the PSNR comparison results for the proposed scheme and three other magic matrix-based schemes [26], [27], [31] at different ER values. It is noticed that these schemes are based on turtle shell matrix [26], extended squared magic matrix [27], and extended turtle shell matrix [31], respectively. From Fig. 9 we can obviously observe that the proposed scheme can maintain higher image quality with different ERs from 0.2 bpp to 2 bpp. We can also see that the curve of Liu *et al.*'s scheme [31] is closest to that of the proposed scheme. On the other hand, Xie *et al.*'s scheme [26] is the worst because a 9×9 square candidate area is used in the embedding phase. This larger area could bring a higher distortion.

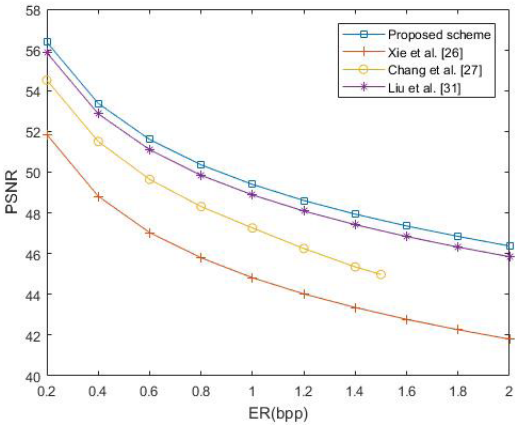


FIGURE 9. PSNR comparisons on different ER values.

To estimate the security of images after embedding by our proposed scheme, we apply pixel difference histogram (PDH) analysis technique [37] to eight cover images and their corresponding stego images under the maximum EC. The PDH illustrates the distribution for the differences between the two pixels in a pair of an image, and it would become more abnormal as the embedded amount of secret message increases. Fig. 10 shows that the histograms of eight stego images remain well preserved when the value of ER is as high as 2 bpp.

The RS steganalysis [11] is applied to the safety evaluation of experiments. It is widely used in the field of data hiding [12]–[14], [22], [23] because of its effectiveness, accuracy and well-known features in detecting LSB embedding. In the RS steganalysis, we first divide the image into non-overlapping blocks. Then, a discrimination function f, a flipping function F and a mask $M = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}$ are used to

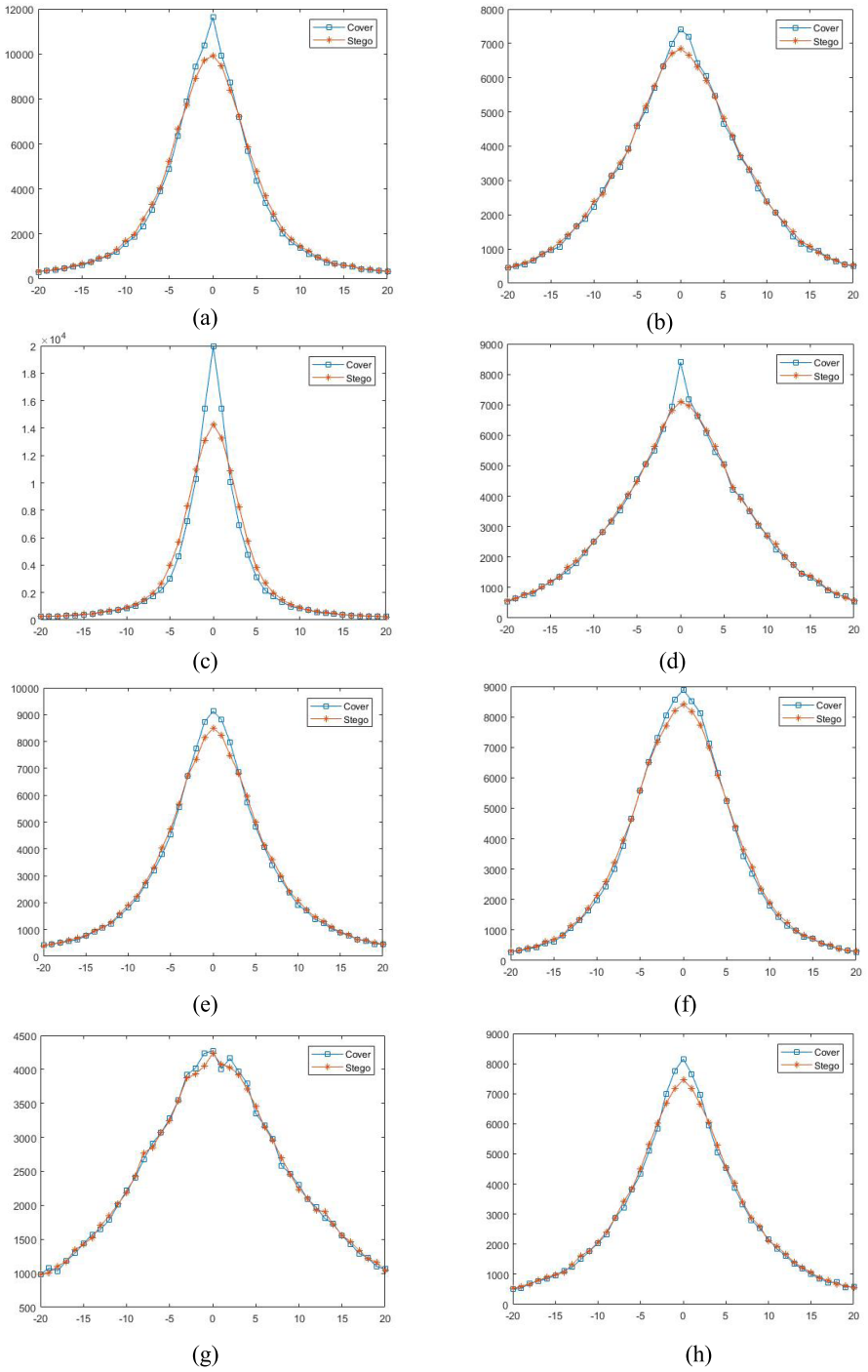


FIGURE 10. PDH by the proposed scheme. (a) Lena. (b) Boat. (c) Airplane. (d) Elaine. (e) Goldhill. (f) Peppers. (g) Baboon. (h) Sailboat.

classify them into three groups: i.e., regular groups, singular groups and unchanged groups. The percentages of regular and singular groups for mask M are denoted as R_M and S_M, respectively. Let F_{-1} be the dual concept of flipping function F. R_{-M} and S_{-M} can be denoted in a similar way when the shifted LSB flipping F_{-1} is applied. For a typical cover image, it must satisfy the following requirement:

$$R_M \cong R_{-M} \quad \text{and } S_M \cong S_{-M}. \tag{12}$$

Fig. 11 shows the RS steganalysis diagrams of eight stego images by our proposed scheme. These well preserved diagrams indicate that the proposed scheme can withstand RS analysis.

It is noted that the proposed scheme is suitable not only for grayscale images, but also for color images. In the case of the color image, we just separate it into R, G, and B layers before embedding, and then apply each layer to the proposed scheme.

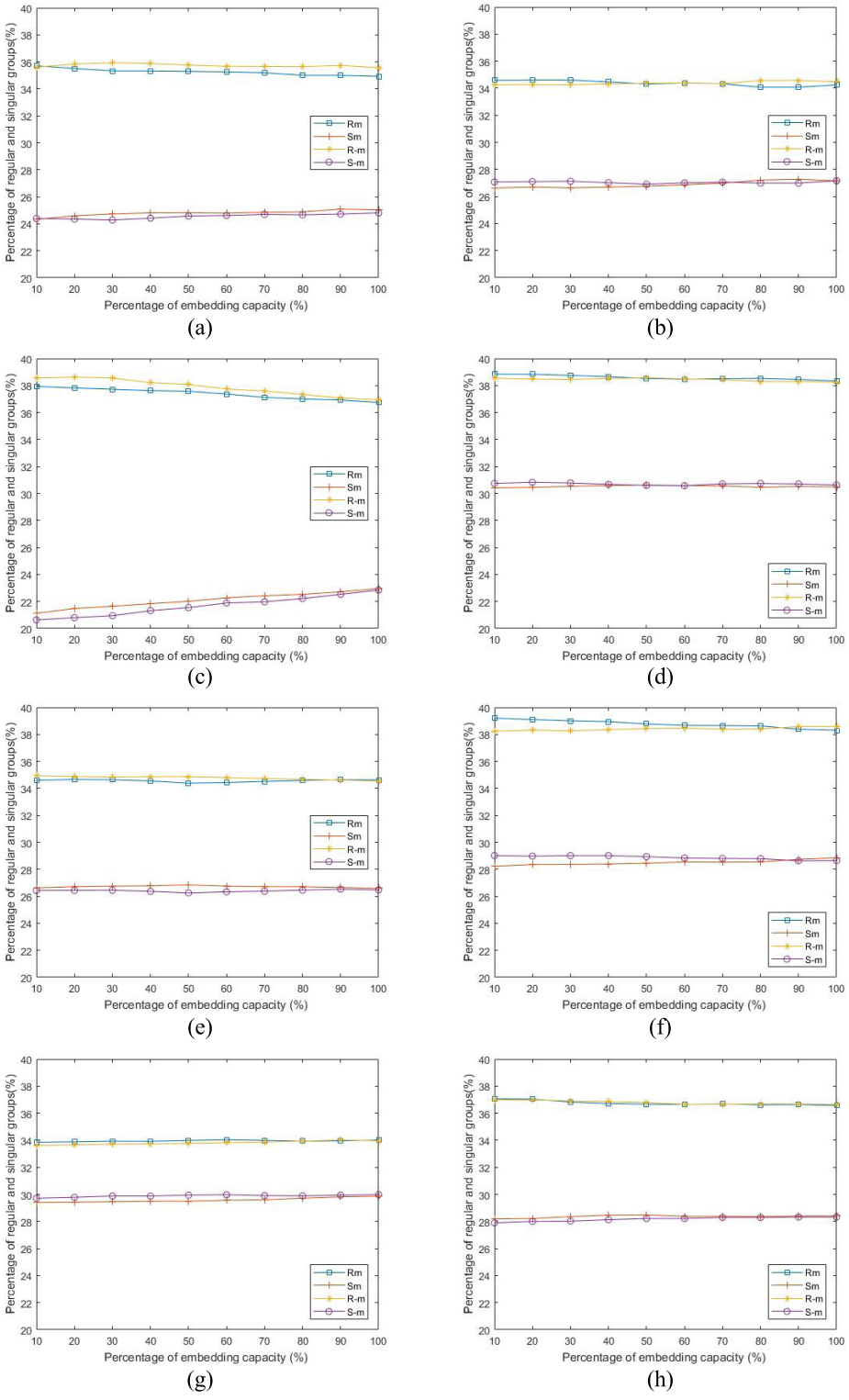


FIGURE 11. RS steganalysis by the proposed scheme with different embedding capacities. (a) Lena. (b) Boat. (c) Airplane. (d) Elaine. (e) Goldhill. (f) Peppers. (g) Baboon. (h) Sailboat.

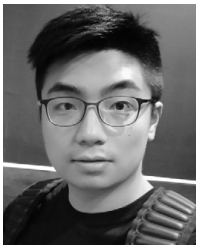
V. CONCLUSION

In this paper, we proposed a novel data hiding scheme based on the mini-Sudoku matrix (MSM). In most of the magic matrix-based data hiding schemes, it is possible to embed more secret data by expanding the values of the elements in the magic matrix, but this would lead to a large increase in the Euclidean distance between the original pixel and the stego pixel, thereby causing the value of PSNR to decrease

rapidly. In our proposed scheme, we make full use of the characteristics of the MSM elements that have high repeatability. The embedding process for a pixel pair in the cover image contains three main steps. First, a 4×4 candidate block in the MSM for the pixel pair is determined. Second, the mapping value in the candidate block embeds 2-bit secret information. Third, the horizontal and vertical coordinates of this mapping value in the MSM carry out modulo operation, making it to embed two more bits to increase the embedding capacity. According to the experimental results, the embedding rate of our proposed scheme reached 2 bpp, and the PSNR was greater than 46 dB. Another appealing feature of our proposed scheme is that the MSM can be extended to enhance the embedding capacity further. Also, the proposed scheme is applicable to other magic matrices, such as the turtle shell matrix, the traditional Sudoku matrix, and others.

REFERENCES

[1] M. Matsui, "Linear cryptanalysis method for DES cipher," in *Proc. Workshop Theory Appl. Cryptograph. Techn.* Berlin, Germany: Springer, 1993, pp. 386–397.
[2] R. L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," *Commun. ACM*, vol. 21, no. 2, pp. 120–126, Feb. 1978.
[3] C. Qin and Y.-C. Hu, "Reversible data hiding in VQ index table with lossless coding and adaptive switching mechanism," *Signal Process.*, vol. 129, pp. 48–55, Dec. 2016.
[4] W. Hong, Y.-B. Ma, H.-C. Wu, and T.-S. Chen, "An efficient reversible data hiding method for AMBTC compressed images," *Multimedia Tools Appl.*, vol. 76, no. 4, pp. 5441–5460, Feb. 2017.
[5] F. Huang, J. Huang, and Y. Q. Shi, "New channel selection rule for JPEG steganography," *IEEE Trans. Inf. Forensics Security*, vol. 7, no. 4, pp. 1181–1191, Aug. 2012.
[6] G. C. Langelaar and R. L. Lagendijk, "Optimal differential energy watermarking of DCT encoded images and video," *IEEE Trans. Image Process.*, vol. 10, no. 1, pp. 148–158, Jan. 2001.
[7] H. Liu, J. Liu, J. Huang, D. Huang, and Y. Q. Shi, "A robust DWT-based blind data hiding algorithm," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, Phoenix-Scottsdale, AZ, USA, vol. 2, May 2002, pp. 672–675.
[8] S. Lee, C. D. Yoo, and T. Kalker, "Reversible image watermarking based on integer-to-integer wavelet transform," *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 3, pp. 321–330, Sep. 2007.
[9] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding," *IBM Syst. J.*, vol. 35, nos. 3–4, pp. 313–336, 1996.
[10] J. Mielikainen, "LSB matching revisited," *IEEE Signal Process. Lett.*, vol. 13, no. 5, pp. 285–287, May 2006.
[11] J. Fridrich and M. Goljan, "Practical steganalysis of digital images: State of the art," *Proc. SPIE*, vol. 4675, pp. 1–13, Apr. 2002.
[12] A. K. Sahu and G. Swain, "A novel n-rightmost bit replacement image steganography technique," *3D Res.*, vol. 10, no. 1, pp. 1–18, 2019.
[13] A. K. Sahu and G. Swain, "A novel multi stego-image based data hiding method for gray scale image," *Pertanika J. Sci. Technol.*, vol. 29, no. 2, pp. 753–768, May 2019.
[14] A. K. Sahu, G. Swain, and E. S. Babu, "Digital image steganography using bit flipping," *Cybern. Inf. Techn.*, vol. 18, no. 1, pp. 59–80, Apr. 2018.
[15] C.-H. Yang, C.-Y. Weng, S.-J. Wang, and H.-M. Sun, "Varied PVD + LSB evading detection programs to spatial domain in data embedding systems," *J. Syst. Softw.*, vol. 83, no. 10, pp. 1635–1643, Oct. 2010.
[16] K. A. Darabkh, A. K. AI-Dhamari, and I. F. Jafar, "A new steganographic algorithm based on multi directional PVD and modified LSB," *J. Inf. Technol. Control*, vol. 46, no. 1, pp. 16–36, 2017.
[17] K.-H. Jung, "Data hiding scheme improving embedding capacity using mixed PVD and LSB on bit plane," *J. Real-Time Image Process.*, vol. 14, no. 1, pp. 127–136, Jan. 2018.
[18] D.-C. Wu and W.-H. Tsai, "A steganographic method for images by pixel-value differencing," *Pattern Recognit. Lett.*, vol. 24, pp. 1613–1626, Jun. 2003.
[19] J. Chen, "A PVD-based data hiding method with histogram preserving using pixel pair matching," *Signal Process., Image Commun.*, vol. 29, no. 3, pp. 375–384, Mar. 2014.
[20] S.-Y. Shen and L.-H. Huang, "A data hiding scheme using pixel value differencing and improving exploiting modification directions," *Comput. Secur.*, vol. 48, pp. 131–141, Feb. 2015.
[21] M. Hussain, A. W. A. Wahab, A. T. S. Ho, N. Javed, and K. H. Jung, "A data hiding scheme using parity-bit pixel value differencing and improved rightmost digit replacement," *Signal Process., Image Commun.*, vol. 50, pp. 44–57, Feb. 2017.
[22] A. K. Sahu and G. Swain, "An optimal information hiding approach based on pixel value differencing and modulus function," *Wireless Pers. Commun.*, vol. 108, no. 1, pp. 159–174, 2019. doi: 10.1007/s11277-019-06393-z.
[23] A. K. Sahu and G. Swain, "Pixel overlapping image steganography using PVD and modulus function," *3D Res.*, vol. 9, no. 3, p. 40, Sep. 2018.
[24] X. Zhang and S. Wang, "Efficient steganographic embedding by exploiting modification direction," *IEEE Commun. Lett.*, vol. 10, no. 11, pp. 781–783, Nov. 2006.
[25] H. J. Kim, C. Kim, Y. Choi, S. Wang, and X. Zhang, "Improved modification direction methods," *Comput. Math. Appl.*, vol. 60, no. 2, pp. 319–325, Jul. 2010.
[26] X.-Z. Xie, Y. Liu, and C.-C. Chang, "Extended squared magic matrix for embedding secret information with large payload," *Multimedia Tools Appl.*, vol. 78, no. 14, pp. 19045–19059, Jul. 2019.
[27] C.-C. Chang, Y.-C. Chou, and T. D. Kieu, "An information hiding scheme using Sudoku," in *Proc. 3rd Int. Conf. Innov. Comput. Inf. Control (ICICIC)*, Dalian, China, Jun. 2008, pp. 17–22.
[28] B.-B. Xia, A.-H. Wang, C.-C. Chang, and L. Liu, "An image steganography scheme using 3D-Sudoku," *J. Inf. Hiding Multimedia Signal Process.*, vol. 7, no. 4, pp. 836–845, Jul. 2016.
[29] T.-S. Nguyen and C.-C. Chang, "A reversible data hiding scheme based on the Sudoku technique," *Displays*, vol. 39, pp. 109–116, Oct. 2015.
[30] C. C. Chang, Y. Liu, and T. S. Nguyen, "A novel turtle shell based scheme for data hiding," in *Proc. 10th Int. Conf. Intell. Inf. Hiding Multimedia Signal Process. (IIHMSP)*, Kitakyushu, Japan, Aug. 2014, pp. 89–93.
[31] Y. Liu, C.-C. Chang, and T.-S. Nguyen, "High capacity turtle shell-based data hiding," *IET Image Process.*, vol. 10, no. 2, pp. 130–137, Feb. 2016.
[32] Q. Jin, Z. Li, C.-C. Chang, A. Wang, and L. Liu, "Minimizing turtle-shell matrix based stego image distortion using particle swarm optimization," *Int. J. Netw. Secur.*, vol. 19, no. 1, pp. 154–162, Jan. 2017.
[33] L. Liu, C.-C. Chang, and A. Wang, "Data hiding based on extended turtle shell matrix construction method," *Multimedia Tools Appl.*, vol. 76, no. 10, pp. 12233–12250, May 2017.
[34] X.-Z. Xie, C.-C. Lin, and C.-C. Chang, "Data hiding based on a two-layer turtle shell matrix," *Symmetry*, vol. 10, no. 2, p. 47, Feb. 2018. doi: 10.3390/sym10020047.
[35] J. Daemen and V. Rijmen, *The Design of Rijndael: AES—The Advanced Encryption Standard*. Berlin, Germany: Springer-Verlag, 2002.
[36] *Mathematics of Sudoku*. Accessed: Aug. 6, 2019. [Online]. Available: https://en.wikipedia.org/wiki/Mathematics_of_Sudoku#Enumeration_results
[37] X. Zhang and S. Wang, "Vulnerability of pixel-value differencing steganography to histogram analysis and modification for enhanced security," *Pattern Recognit. Lett.*, vol. 25, no. 3, pp. 331–339, Feb. 2004.



MINGZE HE received the B.S. degree in computer science and technology from Xihua University, Chengdu, China, in 2017, where he is currently pursuing the M.S. degree in computer science and technology. His current research interests include cryptography, information security, and information hiding.

YANJUN LIU received the Ph.D. degree from the School of Computer Science and Technology, University of Science and Technology of China (USTC), Hefei, China, in 2010. She was an Assistant Professor with Anhui University, China, from 2010 to 2015. She has been a Senior Research Fellow with Feng Chia University, Taiwan, since 2015. Her specialties include E-business security and electronic imaging techniques.

CHIN-CHEN CHANG received the Ph.D. degree in computer engineering from National Chiao Tung University. On numerous occasions, he was invited to serve as a Visiting Professor, the Chair Professor, an Honorary Professor, an Honorary Director, an Honorary Chairman, a Distinguished Alumnus, a Distinguished Researcher, and a Research Fellow by universities and research institutes. His current title is the Chair Professor with the Department of Information Engineering and Computer Science, Feng Chia University, since February 2005. His current research interests include database design, computer cryptography, image compression, and data structures. He is also a Fellow of IEE, U.K.

MINGXING HE received the M.S. degree from Chongqing University and the Ph.D. degree from Southwest Jiaotong University, in 1990 and 2003, respectively. He is currently a Full Professor with the School of Computer and Software Engineering, Xihua University, Chengdu, China. He has coauthored five books and has published more than 100 articles in refereed professional journals and international conferences. His current research interests include cryptography and information security. He is a member of the ACM and a Senior Member of CACR. He received the DAAD Scholarship Reward of Germany, in 2002, the Excellent Ph.D. Dissertation Award in Southwest Jiaotong University, in 2003, and the Grant of National Science Foundation of China (NSFC), in 2004, 2007, and 2015.