Computer Modeling in Engineering & Sciences
DOI:10.32604/cmes.2020.09355
Article

Tech Science Press

3D Multilayered Turtle Shell Models for Image Steganography

Ji-Hwei Horng[1], Juan Lin[2,*], Yanjun Liu[3] and Chin-Chen Chang[3,4,*]

[1]National Quemoy University, Kinmen County, 892, Taiwan
[2]Engineering Research Center for ICH Digitalization and Multi-Source Information Fusion, Fuqing Branch of Fujian Normal University, Fujian Province University, Fuzhou, 350300, China
[3]Department of Information Engineering and Computer Science, Feng Chia University, Taichung, 40724, Taiwan
[4]School of Computer Science and Technology, Hangzhou Dianzi University, Hangzhou, 310018, China
[*]Corresponding Authors: Juan Lin. Email: lj2020229@gmail.com; Chin-Chen Chang. Email: alan3c@gmail.com
Received: 06 December 2019; Accepted: 09 March 2020

Abstract: By embedding secret data into cover images, image steganography can produce non-discriminable stego-images. The turtle shell model for data hiding is an excellent method that uses a reference matrix to make a good balance between image quality and embedding capacity. However, increasing the embedding capacity by extending the area of basic structures of the turtle shell model usually leads to severe degradation of image quality. In this research, we innovatively extend the basic structure of the turtle shell model into a three-dimensional (3D) space. Some intrinsic properties of the original turtle shell model are well preserved in the 3D version. Theoretic analysis shows that the new proposed models have good performance both in the image quality and in the complexity of the reference matrix. Our experimental results justify the theoretic conclusions.

Keywords: 3D turtle shell; data hiding; reference matrix; image quality

1 Introduction

Due to the rapid development of portable devices, wireless communications, and social networks, the transmission of huge amounts of digital information over the internet has become a daily routine. Because hiding data in an image and transmitting it over the internet or posting it to the social media will not get noticed, image steganography provides a good way for secret communication.

According to the processing domain, data hiding schemes for digital images can be classified into three categories, including the compressed [1–3], frequency [4–6], and spatial domain methods. Among them, the spatial domain methods can be further divided into the least significant bit substitution [7–9], the pixel value differencing [10–12], and the reference matrix (RM) methods [13–17].

In this research, we focus on the RM-based data hiding approach. The exploiting modification direction method [13] and its improved versions [14,18] produce high-quality stego-images; however, their RMs are too simple and regular in the aspect of security. The Sudoku-based methods [15,19] effectively increase the complexity of their RMs. But, the quality of the stego-images degrades significantly. The turtle shell model [16] provides a compromise solution between image quality and security level. Later, the turtle shell model is expanded to the octagon-shaped shell models [17,20] with various sizes. Although the octagon shape

effectively preserves the compactness property of the turtle shell model, the image quality degrades dramatically as the embedding capacity increased with the size of the fundamental octagons.

In order to expand the embedding capacity of the turtle shell model, in this paper, we extend the RM into a three-dimensional (3D) space. The rest of this paper is organized as follows. We will briefly introduce the turtle shell-based and the octagon-shaped shell-based data hiding schemes and discuss their defects in Section 2. In Section 3, we will propose a generalized embedding rule for determining the optimal associated set for the mentioned schemes above that can make the embedding process more efficient. Two 3D turtle shell models for data hiding based on the generalized embedding rule will be presented in Section 4. We will discuss some intrinsic properties of the two new models. Theoretic analysis and experimental results of the proposed models will be given in Sections 5 and 6, respectively. Finally, conclusions are made in Section 7.

2 Related Works

In this section, first, we will introduce two related data hiding schemes, including the turtle shell-based scheme and the octagon-shaped shell-based scheme. Some defects in their embedding rules are also discussed. Then, we try to derive a more general and efficient data embedding rule. In addition, inspired by these schemes, we will propose 3D multilayered turtle shell models to improve the quality of stego image and the embedding capacity of the RM-based approach.

2.1 The Turtle Shell-Based Scheme

In the turtle shell-based scheme proposed by Chang et al. [16], the pixels of the cover image P with size of $H \times W$ are rearranged to obtain the cover pixel set $P = \{p_i | i = 1, 2, \ldots, (H \times W)\}$. The binary stream of secret message is transformed into $S = \{s_j | j = 1, 2, \ldots, n\}$, where s_j is an 8-ary secret digit. Before embedding, a reference matrix M with size of 256×256 consisting of a number of hexagons, called turtle shells, is constructed as shown in Fig. 1. The elements in the matrix M are classified into two types, i.e., regular and special elements. The regular elements are located on the turtle shells and further divided into back digits and edge digits. Then, the following rules are applied to find the corresponding associated set G of each element $M(p_i, p_{i+1})$.

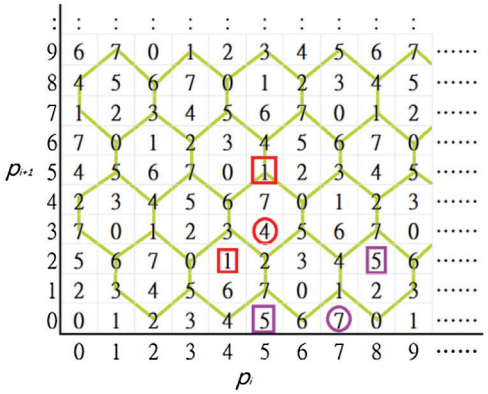


Figure 1: The reference matrix of the turtle shell-based scheme

Case 1: For regular elements

R1. If $M(p_i, p_{i+1})$ is a back digit within a turtle shell, the associated set G is the collection of all digits in this turtle shell.

R2. If $M(p_i,\ p_{i+1})$ is an edge digit involved in at least one turtle shell, the associated set G is the collection of all digits that belong to the involved turtle shells.

Case 1: For special elements

R3. If $M(p_i,\ p_{i+1})$ does not involve in any turtle shell, the associated set G is the collection of all digits in its nearest 3×3 sub-matrix.

To embed secret digit s_j, apply the gray level values of a cover pixel pair $(p_i,\ p_{i+1})$ as the coordinate to locate the currently processed element $M(p_i,\ p_{i+1})$ in M. By employing its corresponding rule, the associated set G is obtained and the cover pixel pair $(p_i,\ p_{i+1})$ is modified to $\left(p_i^{'}, p_{i+1}^{'}\right)$, where $M\left(p_i^{'}, p_{i+1}^{'}\right) \in G$ and $M\left(p_i^{'}, p_{i+1}^{'}\right) = s_j$. If there are multiple solutions, the shortest distant one which will lead to the least modification is selected.

However, the given rules will not produce optimal solutions for some cases. Two examples are shown in Fig. 1. When the currently processed element is (5, 3) and the secret digit is 1, its corresponding rule R1 will modify the pixel values to (5, 5). However, the optimal solution is (4, 2). The error vectors are (0, 2) and (−1, −1), respectively. Under the square error measurement, the solution (4, 2) is better than (5, 5). Another example is at position (7, 0). If the secret digit to be embedded is 5, the corresponding rule R3 will modify the pixel values to (8, 2). However, the optimal solution is (5, 0).

The embedding rules are hard to design because there are too many different cases of hexagon in the reference matrix. In Fig. 1, we can't find any pair of hexagons having the same configuration of numbers, although their contents are the same. In the positive direction of p_i-axis, the value of each neighboring element is incremented by 1 with modulo-8 rightward, i.e., $M(p_i + 1,\ p_{i+1}) = M(p_i,\ p_{i+1}) + 1$. Therefore, the matrix will repeat itself with a period of 8 in the p_i-direction. On the other hand, the incremental value is an alternation of 2 and 3 in the positive p_{i+1}-direction. Thus, it requires a total number of $(2 + 3) \times 8 = 40$ incremental values to achieve a whole cycle. The period of repetition in the p_{i+1}-direction is, therefore, $2 \times 8 = 16$.

To simplify the analysis, we use an algebraic expression to represent the local relationships of a currently processed element with its nearest neighbors. For a pair of cover pixels $(p_i,\ p_{i+1})$, assume its corresponding value $M(p_i,\ p_{i+1})$ on the reference matrix M is x. We assign the matrix values around the element $M(p_i,\ p_{i+1})$ according to the construction rule of Chang et al. [16], as shown in Fig. 2. Due to the alternating increment of 2 and 3 in the p_{i+1}-direction, the general combinations of $(p_i,\ p_{i+1})$ could be classified into two cases: (1) $p_{i+1} = 1 + 2n$ and (2) $p_{i+1} = 2 + 2n$, where n denotes any proper integer

$x + 2$	$x + 3$	$x + 4$
$x - 1$	x	$x + 1$
$x - 3$	$x - 2$	$x - 1$

(a)

$x + 1$	$x + 2$	$x + 3$
$x - 1$	x	$x + 1$
$x - 4$	$x - 3$	$x - 2$

(b)

Figure 2: The associated set G for two different cases of regular elements in the turtle shell model. (a) The G of the element $M(p_i,\ p_{i+1}) = x$ for $p_{i+1} = 1 + 2n$, (b) The G of the element $M(p_i,\ p_{i+1}) = x$ for $p_{i+1} = 2 + 2n$

value that makes p_{i+1} within the defined domain. The associated set G of the element $M(p_i,\ p_{i+1}) = x$ is the painted area as shown in Fig. 2. Thus, we can simplify the associated set G to these two cases for regular elements.

To define the associated set G of the element $M(p_i,\ p_{i+1}) = x$, we seek to find a set of nearest elements that contains all distinct values of a secret digit. The precise meaning of the 'nearest' element is the one that produces minimum sum of square errors under embedding. The procedure is to sequentially include the distinct-valued neighboring elements in the ascending order of square error. Fig. 3 provides a fast look-up table of the sum of square errors $\Delta p_i^2 + \Delta p_{i+1}^2$ for deciding the including order. By mapping Figs. 3 to 2a with their center aligned, we sequentially include x, $x+1$, $x+3$, $x-1$, $x-2$, $x+4$, $x+2$ and $x-3$ by referring to their sum of square errors 0, 1, 1, 1, 1, 2, 2, and 2, respectively. In this way, we get the optimal associated set for embedding. The optimal associated set for the case of Fig. 2b can be obtained by including x, $x+1$, $x+2$, $x-1$, $x-3$, $x+3$, $x-4$ and $x-2$ sequentially with the same rule. The $x+1$ in the upper left corner is skipped because a same valued element with a lower square error has been included.

8	5	4	5	8
5	2	1	2	5
4	1	0	1	4
5	2	1	2	5
8	5	4	5	8

Figure 3: A look-up table for sum of square errors

For the cases of either p_i or $p_{i+1} = 0$ or 255, the associated set could be determined in a similar manner. An example is shown in Fig. 4. Assuming the currently processing element is the 'x' at the center of bottom, then, the optimal associated set G is the painted area.

By applying the modified associated set for regular and special elements, the turtle shell-based scheme could get its optimal solution and result in a slightly higher image quality.

$x+3$	$x+4$	$x+5$	$x+6$	$x+7$
x	$x+1$	$x+2$	$x+3$	$x+4$
$x-2$	$x-1$	x	$x+1$	$x+2$

Figure 4: The associated set for a special element in the turtle shell model

2.2 The Octagon-Shaped Shell-Based Scheme

In the octagon-shaped shell-based scheme [17], as shown in Fig. 5, the incremental values in the positive x_1-direction are periodic series of 4, 4 and 5, where x_1 and x_2 denote the values of the cover pixel pair. Kurup et al.'s embedding rules follow the same concept of Chang et al.'s rules [16]. An example is marked in Fig. 5 that the cover pixel pair is (4, 2) and the secret digit is '11'. Since $M(4,\ 2)$ is a back digit, the cover pixel pair will be modified to (6, 1) according to R1, instead of the optimal solution (2, 2).

To obtain the optimal solution with an efficient way, we apply the algebraic expression again to find the optimal associated set. As mentioned above, the incremental values in the x_1-direction are periodic series of

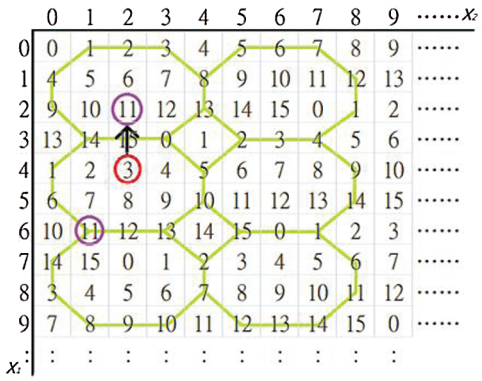


Figure 5: The reference matrix of the octagon-shaped shell-based scheme

$k-11$	$k-10$	$k-9$	$k-8$	$k-7$
$k-7$	$k-6$	$k-5$	$k-4$	$k-3$
$k-2$	$k-1$	k	$k+1$	$k+2$
$k+2$	$k+3$	$k+4$	$k+5$	$k+6$
$k+6$	$k+7$	$k+8$	$k+9$	$k+10$

(a)

$k-11$	$k-10$	$k-9$	$k-8$	$k-7$
$k-6$	$k-5$	$k-4$	$k-3$	$k-2$
$k-2$	$k-1$	k	$k+1$	$k+2$
$k+2$	$k+3$	$k+4$	$k+5$	$k+6$
$k+7$	$k+8$	$k+9$	$k+10$	$k+11$

(b)

$k-10$	$k-9$	$k-8$	$k-7$	$k-6$
$k-6$	$k-5$	$k-4$	$k-3$	$k-2$
$k-2$	$k-1$	k	$k+1$	$k+2$
$k+3$	$k+4$	$k+5$	$k+6$	$k+7$
$k+7$	$k+8$	$k+9$	$k+10$	$k+11$

(c)

Figure 6: The optimal associated set G for three different cases of regular elements in the octagon-shaped shell model. (a) $x_1 = 2 + 3n$, (b) $x_1 = 3 + 3n$ and (c) $x_1 = 4 + 3n$

4, 4 and 5. It results in three cases of neighboring relationships. Assume $M(x_1, x_2) = k$ and let n be any proper integer value that makes x_1 within the defined domain. Fig. 6 shows the three possible cases: (a) $x_1 = 2 + 3n$, (b) $x_1 = 3 + 3n$ and (c) $x_1 = 4 + 3n$. The painted area in each case corresponds to its optimal associated set.

For the special elements that $x_1, x_2 = 0$, 1, 254 or 255, the optimal associated set can be obtained in a manner like in Fig. 4. They are rare cases and with various types, so we can't give them all here. Another feasible way is to expand the associated set slightly to a rectangular shape that guarantees to contain the optimal set. However, under such situation, a secret digit s_j may match with more than one solution in the

associated set. Multiple solutions should be compared to obtain the least square error one. Details of this problem will be discussed in the next section.

3 A Generalized Embedding Rule

According to the discussions in the previous section, we can devise a generalized embedding rule for the turtle shell model, the octagon-shaped shell model or any enlarged octagon-shaped shell model. The proposed rule is summarized as follows.

Step 1: Construct the RM.

Step 2: According to the construction rule of the RM, determine the number of different cases for regular elements. Then, find the optimal associated set for each case.

Step 3: For special elements in the margin of the RM, a suitable sized rectangular matrix is applied to be its associated set.

Step 4: Embed a secret digit to each cover pixel pair based on the obtained associated set.

The number of different cases for regular elements depends on the construction rule of the RM. In the turtle shell model and the octagon-shaped shell model, different spacing of element values in any direction of axis will lead to different neighborhood configurations in the algebraic expression. Therefore, we should derive the associated set for each case.

The optimal associated set derived in Step 2 has a complete set of distinct numbers. Therefore, in the embedding process, just find the only matching element $M(x_1', x_2') = s_j$ and modify the pixel values to (x_1', x_2').

On the other hand, the rectangular matrix in Step 3 should have enough elements so that the optimal associated set belongs to the matrix. The example cases for the turtle shell model is 5×3 and the associated set for the octagon-shaped shell model could be a matrix of size 5×4, which has its long sides coincide with the direction of the margin.

In general, the design of associated set G for a special element can refer to Fig. 7, where d denotes the distance of the special element to the boundary of the RM. We choose proper values of m and n such that the rectangular matrix contains the optimal associated set. To embed the secret digit s_j to the cover pixel pair (x_1, x_2), we search in the associated set to find the nearest element, and thus minimize the square error satisfying $M(x_1', x_2') = s_j$ and modify the cover pixel values to (x_1', x_2').

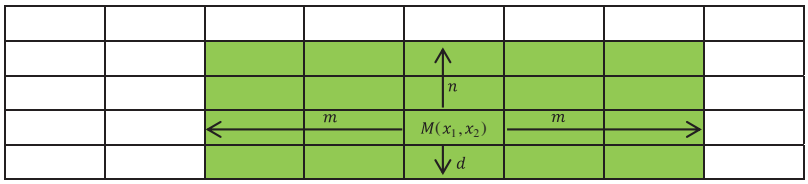


Figure 7: An example of rectangular associated set for a special element

4 The Proposed 3D Multilayered Turtle Shell Schemes

In this section, we will first introduce some intrinsic features of the turtle shell model. It is noted that we can treat the octagon-shaped shell model as a generalization of the turtle shell model even though there are still something different between them. Then, we try to devise two new 3D multilayered turtle shell models for data hiding that meet the same features of the original turtle shell model. For each model, we also show how the generalized embedding rule can be incorporated to embed secret data.

4.1 Intrinsic Features of the Turtle Shell Model

4.1.1 2^n-Sized Basic Structures

All the image data hiding schemes based on the turtle shell model or the octagon-shaped shell model share the same feature that all the basic structures of repetition are 2^n of size in order to reduce the computational complexity of the embedding process. Since the secret digits are usually stored in the binary format, the embedding process of a 2^n-sized basic structure does not require any conversion between number systems.

4.1.2 Compactness

A major index for the performance of image steganography is the peak signal-to-noise ratio (PSNR) value. To ensure a high PSNR value, the basic structures of repetition in the RM should be compactly connected. Both the turtle shell model and the octagon-shaped shell model meet the feature of compactness and can produce stego-images of high PSNR values.

4.1.3 Partition of Space

In the turtle shell model, the basic constructing hexagons are closely connected to each other without any redundant elements. That is, the hexagons constitute a perfect partition of the defined space, while the octagon-shaped shell model leaves some corner elements that do not belong to any octagon and, thus, do not form a space-partition.

4.1.4 Translation Invariance

In Chang et al.'s scheme [16], although they treated the turtle shell model as a fixed mesh on the RM, the mesh can be randomly translated in the p_i or p_{i+1} direction of axis without losing its properties. As discussed in Section 3, the proposed generalized embedding rule is irrelevant to the precise location of this virtual mesh. On the other hand, the octagon-shaped shell [17] model cannot be translated randomly in its x_1 and x_2 directions of axes. Only being translated with integer multiples of its repeating periods can preserve the required properties of an octagon.

4.2 The 3D Multilayered Turtle Shell Models

In this sub-section, we will propose two 3D multilayered turtle shell models for data hiding based on our generalized embedding rule, which share the same properties with the original model [16]. These properties are the 2^n-sized basic structures, the compactness, the partition of space, and the translation invariance.

4.2.1 The 2-Layered Model (Size of 2^4)

The first model we will introduce here is the 2-layered turtle shell model, whose RM M is a $256 \times 256 \times 256$ 3D array as shown in Fig. 8. The number system here is radix-2^4, that is, each number should be represented in a 16-ary format. For the base plane of $z = 0$, the value of the origin is zero, i.e., $M(0,\ 0,\ 0) = 0$. Then, each element value is incremented by 1 in the positive y direction of axis, while incremented by 7 in the positive x direction of axis, i.e., $M(x,\ y+1,\ 0) = M(x,\ y,\ 0) + 1$ and $M(x+1,\ y,\ 0) = M(x,\ y,\ 0) + 7$, respectively. After completing the plane of $z = 0$, we successively construct the plane of $z+1$ by simply adding 4 to its previous plane z, i.e., $M(x,\ y,\ z+1) = M(x,\ y,\ z) + 4$. Two examples of the basic structure in the 3D RM are painted with different colors for illustration. Two aligned hexagons of the consecutive layers constitute a basic structure which has distinct values of 0~15 in the number system. Since the proposed 2-layered model is translation-invariant, any basic structure can be randomly translated within the RM in any directions of axes without losing its properties. Two example basic structures are illustrated by different colors in the figure.

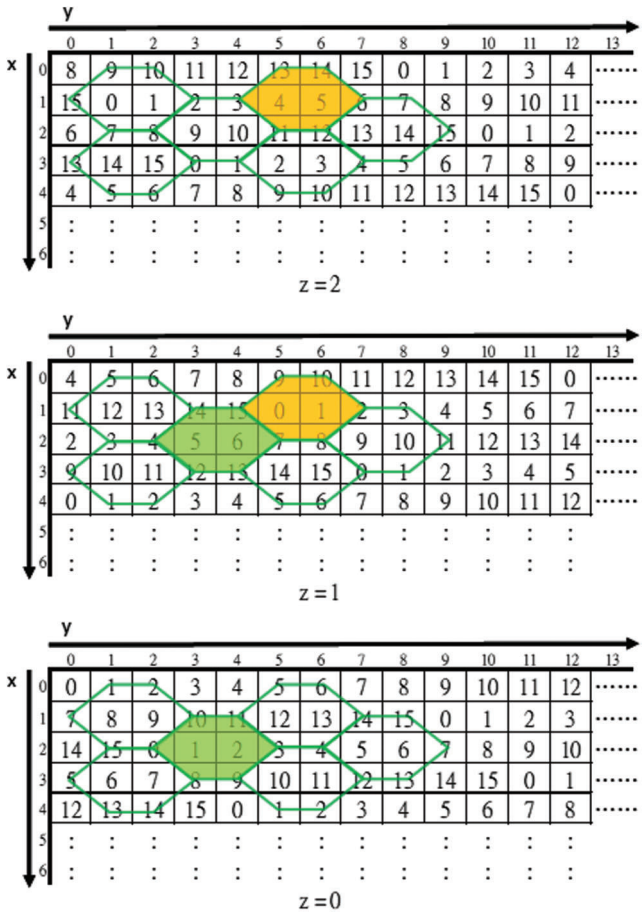


Figure 8: The 2-layered turtle shell model

According to the construction rule, the neighborhood of a regular element $M(x, y, z) = k$ is shown in Fig. 9a. The order for including elements to the optimal associated set G can refer to the 3D square error array in Fig. 10b, where the sum of square errors are given by $\Delta x^2 + \Delta y^2 + \Delta z^2$. Mapping Figs. 9b to 9a with their center aligned, we can successively include the values k, $k+1$, $k-7$, $k-1$, $k+7$, $k+4$, $k-4$, $k-6$, $k+6$, $k+8$, $k+5$, $k+3$, $k-3$, $k-5$, $k-2$, and $k+2$ with the corresponding square errors 0, six '1's, seven '2's, and two '3's to constitute the optimal associated set G.

4.2.2 The Data Embedding and Extraction Scheme for 2-Layered Model (Size of 2^4)

In this subsection, the secret data embedding and extraction scheme for the 2-layered 3D turtle shell model are provided. To further demonstrate the actual executoin of the embedding scheme, an example case is also given.

Data embedding scheme

Input: A cover image P with size of $H \times W$, the binary secret stream B with length L.

Output: A stego-image P'.

Before embedding, the pixels of the cover image P with size of $H \times W$ are rearranged to obtain the cover pixel set $P = \{p_i | i = 1, 2, \ldots, (H \times W)\}$. The binary stream B of secret message is transformed into $S = \{s_j | j = 1, 2, \ldots, L/4\}$, where s_j is an 16-ary secret digit. By applying the generalized embedding rule to this model, the data embedding scheme is provided as follows:

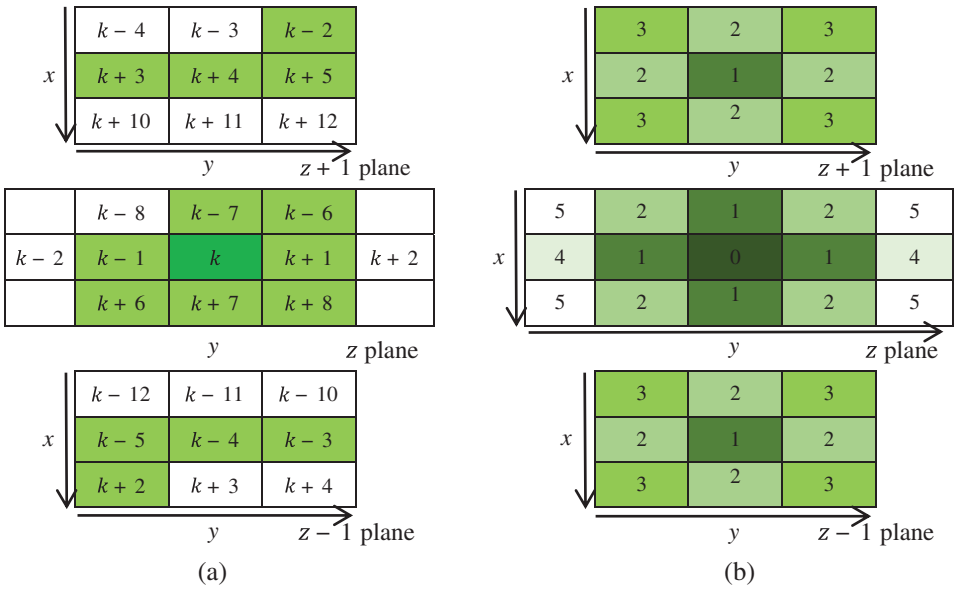


Figure 9: The optimal associated set G with its corresponding square error array for a regular element in the 2-layered model. (a) Optimal associated set and (b) Square error array

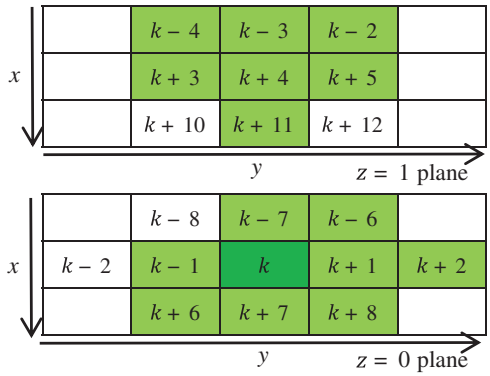


Figure 10: The optimal associated set G for a special element in the 2-layered model

Step 1: Construct the RM M according to the equations

$$M(0,0,0) = 0, \tag{1}$$

$$M(x+1,y,0) = M(x,y,0) + 7, \tag{2}$$

$$M(x,y+1,\ 0) = M(x,y,0) + 1, \tag{3}$$

$$M(x,y,z+1) = M(x,y,z) + 4. \tag{4}$$

Step 2: According to the construction rule of the RM, the unique neighborhood configuration for a regular element is shown in Fig. 9a. The corresponding optimal associated set G is the painted region.

Step 3: For special elements in the margin, i.e., $x, y, z = 0$ or 255, a 3D associated set of $3 \times 5 \times 2$ array with the special element at the center of the 3×5 plane aligned with the margin is suggested. A special case of $z = 0$ is shown in Fig. 10, where the painted region is its optimal associated set G.

Step 4: To embed secret digit s_j into the cover pixel set $(x,\ y,\ z) = (p_i, p_{i+1}, p_{i+2})$ by applying the 2-layered turtle shell model, use the pixel values $(x,\ y,\ z)$ to locate the currently processed element $M(x,\ y,\ z)$ in the RM M. Then, move the center of the 3D mask in Fig. 9a to $M(x,\ y,\ z)$ and let $k = M(x,\ y,\ z)$. Find the only element in the optimal associated set that satisfies $M(x',\ y',\ z') = s_j$, then modify the cover pixel set $(x,\ y,\ z)$ to $(x',\ y',\ z')$.

Data extraction scheme

Input: A stego-image P' with size of $H \times W$, the RM construction information.

Output: The secret binary stream B with length L.

Step 1: Construct the RM M according to the Eqs. (1)–(4).

Step 2: The pixels of the stego-image P' with size of $H \times W$ are rearranged to obtain the cover pixel set $P' = \{p_i' | i = 1,\ 2,\ \ldots,\ (H \times W)\}$.

Step 3: Consecutively utilize each triplet pixels of P' as the coordinates $(x',\ y',\ z') = (p_i', p_{i+1}', p_{i+2}')$ and find the secret digit $s_j = M(x',\ y',\ z')$.

Step 4: Convert the 16-ary secret stream $S = \{s_j | j = 1,\ 2,\ \ldots,\ L/4\}$ back into the binary secret stream B with length L.

The processing diagrams for data embedding and extraction are shown in Figs. 11a and 11b, respectively. An example of data embedding is shown in Fig. 12, where the RM M in Fig. 8 is replotted. As discussed in Sections 2 and 3, although the 2^4-ary distinct numbers are compactly arranged in the two-layered turtle shells, the embedding does not necessarily execute within the basic structure. Instead, the optimal associated set G is the actual processing region. Let the triplet cover pixels are $(x,\ y,\ z) = (2, 7, 1)$. We move the mask in Fig. 9a to be centered at the currently processing element. If the secret digit to be embed is $s_j = 7$. According to the embedding rule, we seek the optimal associated set to find the matching element $M(1,\ 8,\ 2) = 7$ and modify the gray level values from $(2, 7, 1)$ to $(1,\ 8,\ 2)$.

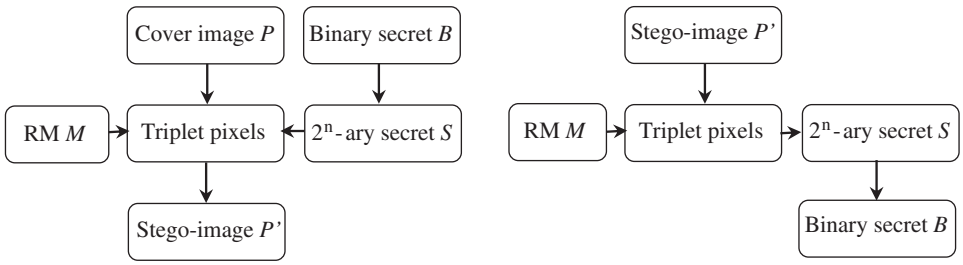


Figure 11: Flow diagram of data embedding and extraction. (a) Data embedding diagram and (b) Data extraction diagram

4.2.3 The 4-Layered Model (Size of 2^5)

The 4-layered turtle shell model can be constructed in a similar way with the 2-layered model. This time, the number system is radix-2^5 and each number is represented in a 32-ary format. Refer to Fig. 13, the incremental value for the positive x, y, and z directions are 12, 4 and 1, respectively. Two examples of the basic structure are also illustrated in the figure. The basic structure has 4 consecutive layers of aligned hexagons and contains distinct values of 0~31. In addition, the 4-layered turtle shell model is also translation-invariant. We can move the basic structure through the defined space of the RM without losing its intrinsic properties.

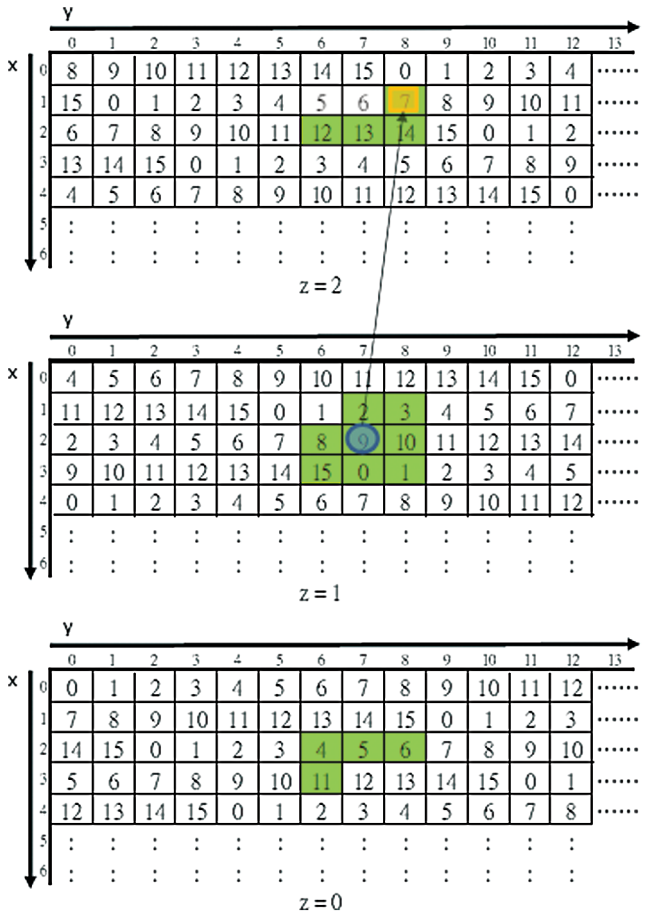


Figure 12: An embedding example for the 2-layered model

By applying the algebraic expression, the neighborhood configuration of a regular element can be uniquely represented as shown in Fig. 14a. Its corresponding look-up table of square errors is shown in Fig. 14b. By sequentially including procedure, we can obtain the optimal associated G set as illustrated by painted region.

4.2.4 The Data Embedding Scheme for 4-Layered Model (Size of 2^5)

In this subsection, the secret data embedding and extraction scheme for the 4-layered 3D turtle shell model are provided. To further demonstrate the actual executoin of the embedding scheme, an example case is also given.

Data embedding scheme

Input: A cover image P with size of $H \times W$, the binary secret stream B with length L.

Output: A stego-image P'.

Before embedding, the pixels of the cover image P with size of $H \times W$ are rearranged to obtain the cover pixel set $P = \{p_i | i = 1, 2, \ldots, (H \times W)\}$. The binary stream of secret message is transformed into $S = \{s_j | j = 1, 2, \ldots, L/5\}$, where s_j is an 32-ary secret digit. By applying the generalized embedding rule to this model, the data embedding scheme is provided as follows:

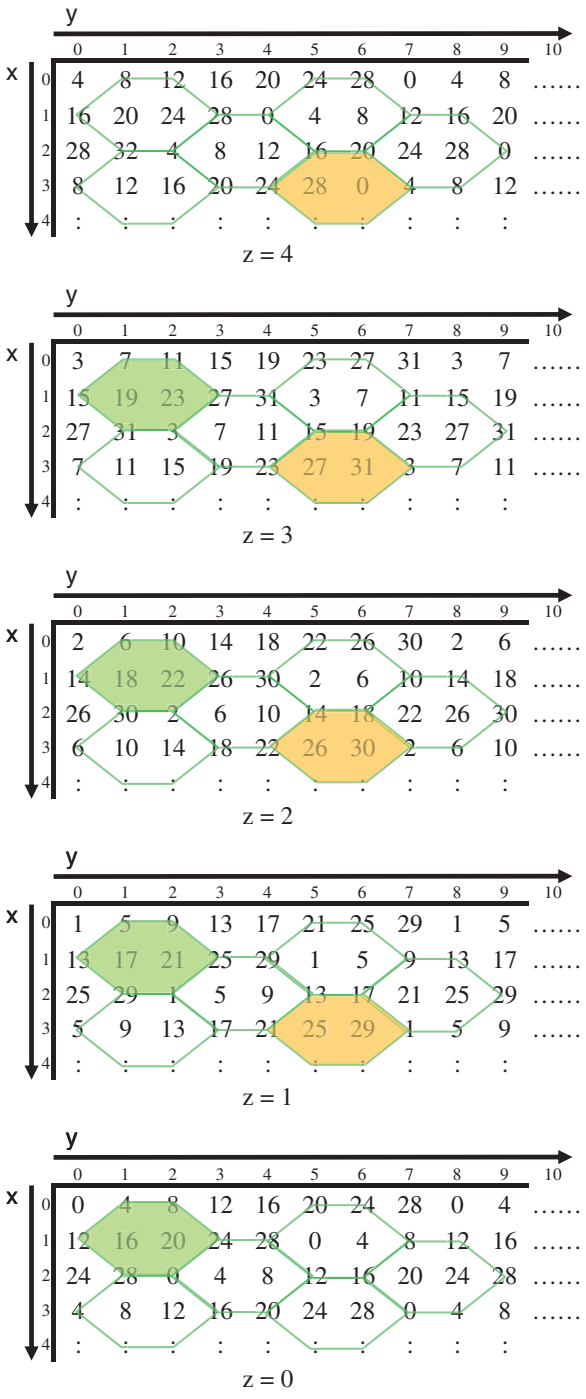


Figure 13: The 4-layered turtle shell model

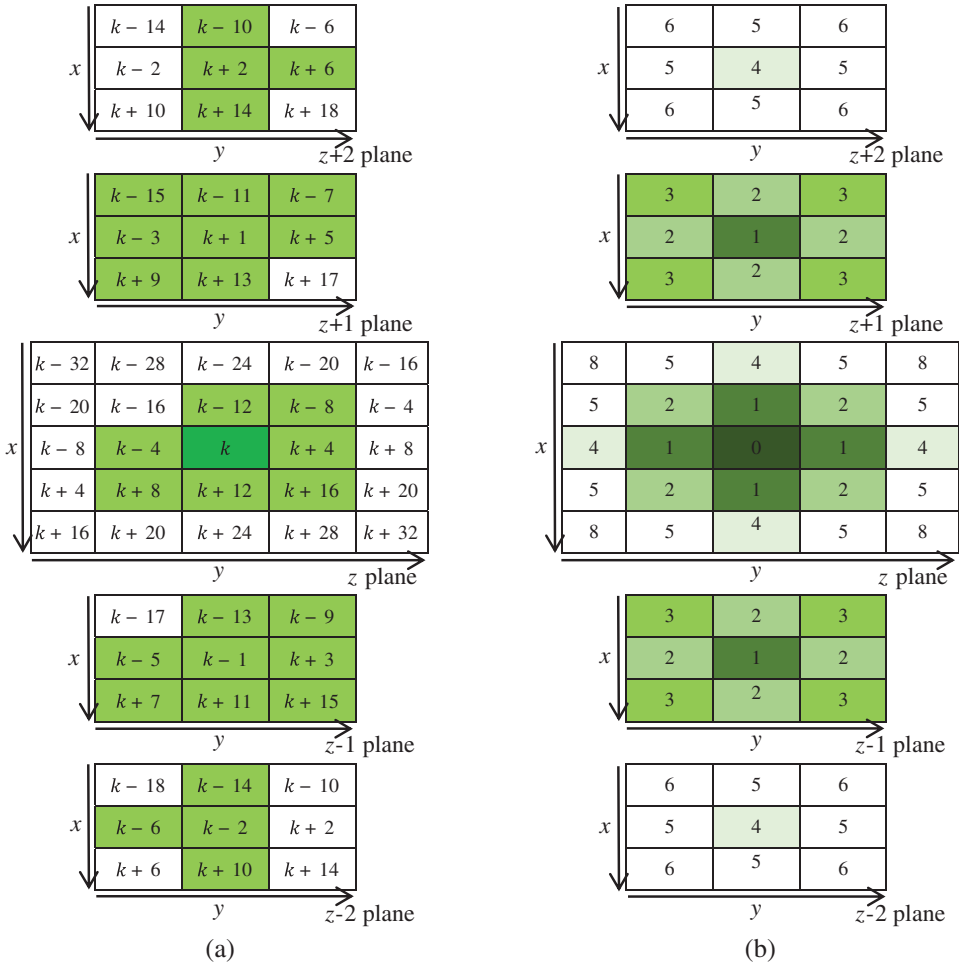


Figure 14: The optimal associated set with its corresponding square error array for a regular element in the 4-layered model. (a) Optimal associated set and (b) Square error array

Step 1: Construct the RM M according to the equations

$$M(0,0,0) = 0, \tag{5}$$

$$M(x+1,y,0) = M(x,y,0) + 12, \tag{6}$$

$$M(x,y+1,\ 0) = M(x,y,0) + 4, \tag{7}$$

$$M(x,y,z+1) = M(x,y,z) + 1. \tag{8}$$

Step 2: According to the construction rule of the RM, the unique neighborhood configuration for a regular element is shown in Fig. 14a. The corresponding optimal associated set G is the painted region.

Step 3: For special elements in the margin, i.e., x, y, $z = 0$, 1, 254 or 255, a suitable sized 3D associated set is applied. Special cases of $z = 0$ and $z = 1$ are shown in Figs. 15a and 15b, respectively, where the painted regions are their optimal associated sets. The cases of $y = 0$ and $y = 1$ are given in Figs. 16a and 16b. The general rule is that any $3 \times 3 \times 5$ array contains the whole set of distinct numbers, the optimal associated set is a sub-set of this array.

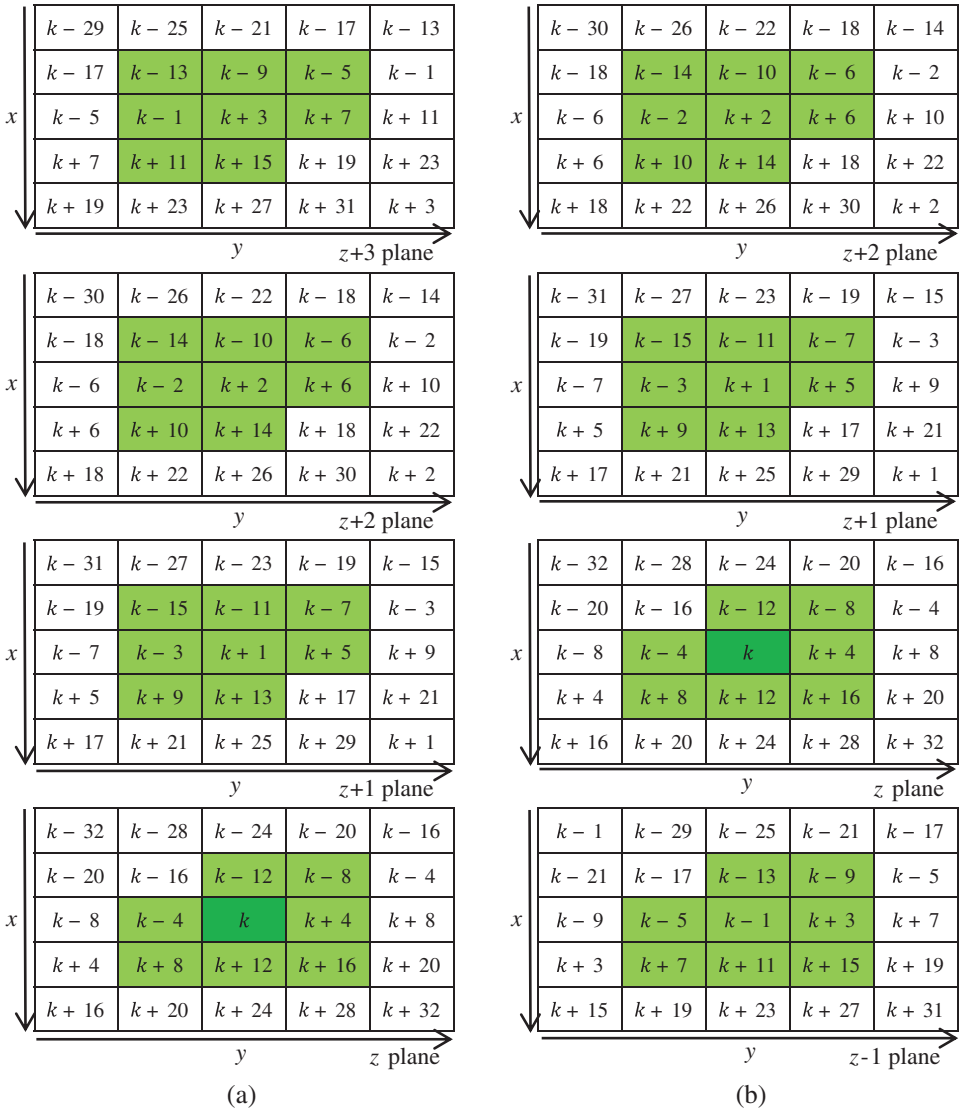


Figure 15: The optimal associated set for two different types of special elements in the 4-layered model. (a) $z = 0$ and (b) $z = 1$

Step 4: To embed secret digit s_j into the cover pixel set $(x,\ y,\ z) = (p_i,\ p_{i+1},\ p_{i+2})$ by applying the 4-layered turtle shell model, use the pixel values $(x,\ y,\ z)$ to locate the currently processed element $M(x,\ y,\ z)$ in the RM M. Then, move the center of the 3D mask in Fig. 9a to $M(x,\ y,\ z)$ and let $k = M(x,\ y,\ z)$. Find the only element in the optimal associated set that satisfies $M(x',\ y',\ z') = s_j$, then modify the cover pixel set $(x,\ y,\ z)$ to $(x',\ y',\ z')$.

Data extraction scheme

Input: A stego-image P' with size of $H \times W$, the RM construction information.

Output: The secret binary stream B with length L.

Step 1: Construct the RM M according to the Eqs. (5)–(8).

Step 2: The pixels of the stego-image P' with size of $H \times W$ are rearranged to obtain the cover pixel set $P' = \{p_i' | i = 1,\ 2,\ \ldots,\ (H \times W)\}$.

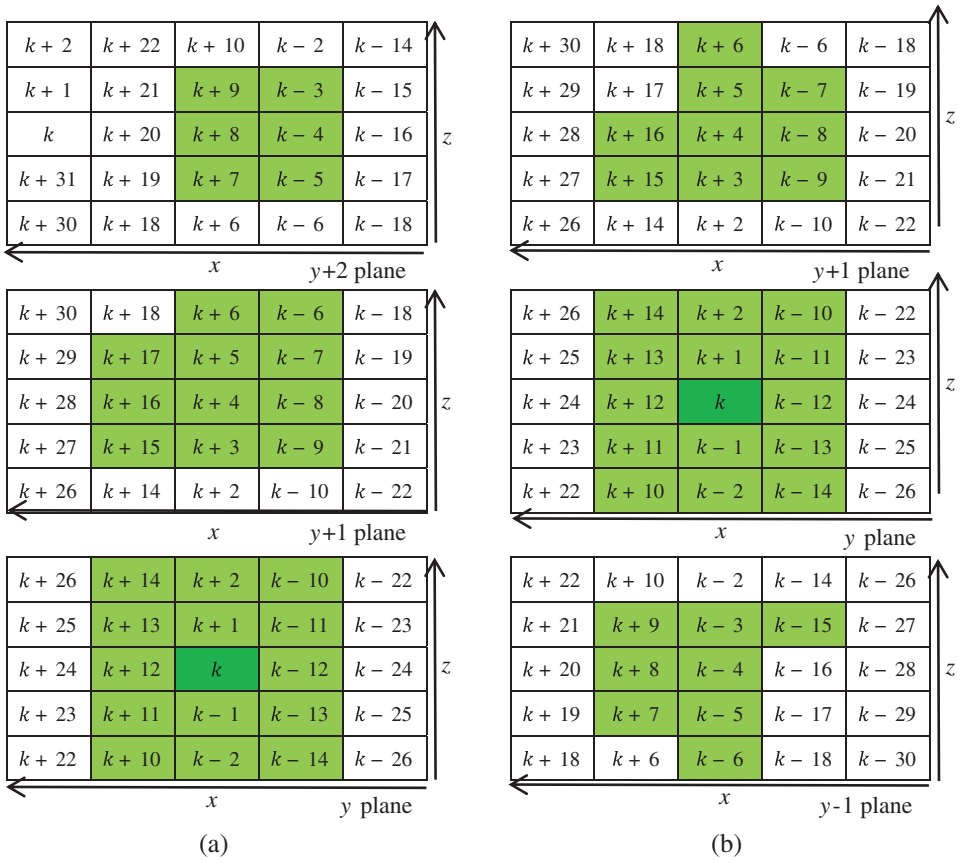


Figure 16: The optimal associated set for two different types of special elements in the 4-layered model. (a) $y = 0$ and (b) $y = 1$

Step 3: Consecutively utilize each triplet pixels of P' as the coordinates $(x',\ y',\ z') = \left(p_i^{'}, p_{i+1}^{'}, p_{i+2}^{'}\right)$ and find the secret digit $s_j = M(x',\ y',\ z')$.

Step 4: Convert the 32-ary secret stream $S = \left\{s_j | j = 1,\ 2,\ \ldots,\ L/5\right\}$ back into the binary secret stream B with length L.

The processing diagrams for data embedding and extraction are the same as Fig. 11. An example of data embedding is shown in Fig. 17, where the RM M in Fig. 13 is replotted. Let the triplet cover pixels are $(x,\ y,\ z) = (2, 6, 2)$. We move the mask in Fig. 14a to be centered at the currently processing element. If the secret digit to be embed is $s_j = 12$. According to the embedding rule, we seek the optimal associated set to find the matching element $M(2,\ 5, 0) = 12$ and modify the gray level values from $(2, 6, 2)$ to $(2,\ 5,\ 0)$.

5 Theoretic Analysis

Before applying the proposed models to real images, we will analyze their theoretic measures of image quality and security level in the section. The theoretic PSNR of stego images and complexity of RMs for the proposed models are compared with the related works as follows.

5.1 PSNR

In Section 3, we proposed a generalized embedding rule. Based on this rule, we give a new definition to the associated set of a currently processed element for each model under discussion. The possible cases of the

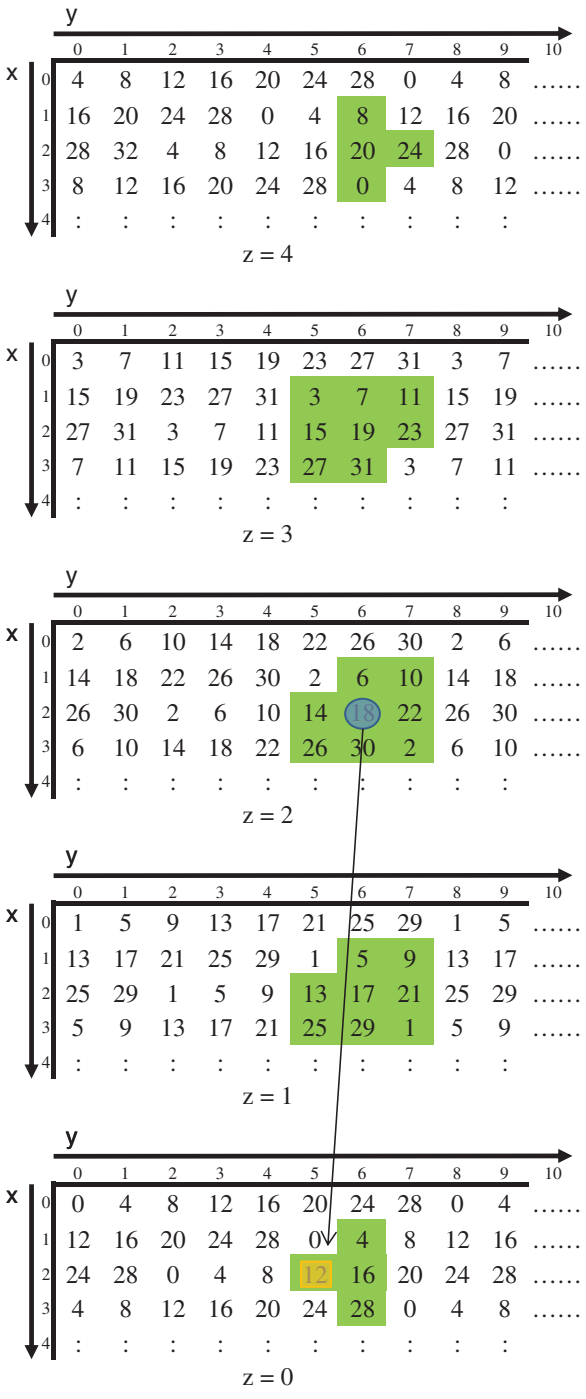


Figure 17: An embedding example for the 4-layered model

associated set G for the regular elements are simplified in a translation-invariant model. This makes the theoretic value of PSNR possible to analyze. In the following, we calculate theoretic PSNR values of Chang et al.'s model, Kurup et al.'s model, and the two proposed models, respectively.

5.1.1 Chang et al.'s Model

In the turtle shell model, two cases of associated set G for regular elements are shown in Fig. 2. Assume all elements in the RM is equally likely to be located and the 8 possible values of the secret digit s_j are also equally likely to be encountered. Then, the theoretic value of MSE can be computed by mapping Figs. 2a–3. The 8 possible values of the secret digit are mapped to one '0', four '1's, and three '2's. MSE is therefore

$$MSE = \frac{1}{2}\left(\frac{1}{8}\times 0 + \frac{4}{8}\times 1 + \frac{3}{8}\times 2\right) = \frac{5}{8}. \tag{9}$$

The number is divided by 2 because the square error in Fig. 3 is the sum of two cover pixels. Although the configuration of Fig. 2b is different from 2a, mapping 2b to 3 leads to the same MSE equation as in Eq. (9). Substitute MSE into the definition of the PSNR will get its theoretic value:

$$PSNR = 10\log_{10}\frac{255^2}{MSE}(\text{dB}) = 50.172. \tag{10}$$

The special elements are more complex to analyze. However, the occasions of applying the special elements are very rare. Therefore, the theoretic value is very close to real experimental value by our experience.

To further explain, an example case is shown in Fig. 18. If the cover pixel pair is $(5, 4)$, its optimal associated set is the painted region. Assuming all outcomes of the secret digit to be embedded are equally likely to occur. Modifying the pixel values to embed 7, 0, 1, 6, 4, 2, 3, and 5 will result in the sum of square errors 0, 1, 1, 1, 1, 2, 2, and 2, respectively. Then, its *MSE* and *PSNR* are coincide with Eqs. (9) and (10). The *PSNR* of all regular elements can be derived in a similar manner and will lead to the same conclusion.

:	:	:	:	:	:	:	:	:	:	:	
9	6	7	0	1	2	3	4	5	6	7	.....
8	4	5	6	7	0	1	2	3	4	5	.....
7	1	2	3	4	5	6	7	0	1	2	.....
6	7	0	1	2	3	4	5	6	7	0	.....
5	4	5	6	7	0	1	2	3	4	5	.....
4	2	3	4	5	6	7	0	1	2	3	.....
3	7	0	1	2	3	4	5	6	7	0	.....
2	5	6	7	0	1	2	3	4	5	6	.....
1	2	3	4	5	6	7	0	1	2	3	.....
0	0	1	2	3	4	5	6	7	0	1	.....
	0	1	2	3	4	5	6	7	8	9	.....

Figure 18: An example to illustrate the theoretic analysis of the *PSNR* for turtle shell model

5.1.2 Kurup et al.'s Model

Based on the same approach and referring to Fig. 6, MSE of the Kurup et al.'s model can be calculated by

$$MSE = \frac{1}{2}\left(\frac{1}{16}\times 0 + \frac{4}{16}\times 1 + \frac{4}{16}\times 2 + \frac{4}{16}\times 4 + \frac{3}{16}\times 5\right) = \frac{43}{32}. \tag{11}$$

It's corresponding theoretic PSNR value is given by

$$PSNR = 46.8476. \tag{12}$$

5.1.3 The 2-Layered Turtle Shell Model (Size 2^4)

The associated set with its corresponding square error array for a regular element in the 2-layered turtle shell model is shown in Fig. 10. By mapping the optimal associated set into the square error array, MSE and PSNR can be conducted as follows.

$$MSE = \frac{1}{3}\left(\frac{1}{16}\times 0+\frac{6}{16}\times 1+\frac{7}{16}\times 2+\frac{2}{16}\times 3\right)=\frac{26}{48}. \tag{13}$$

$$PSNR = 50.7935. \tag{14}$$

Since the square error is contributed by 3 cover pixels for the 3D model, the sum of square errors is divided by 3 to get MSE.

Without loss of generality, the example triple cover pixels $(2, 7, 1)$ with its optimal associated set is shown in Fig. 19. Modifying the pixel values to embed 9, 10, 2, 8, 0, 13, 5, 3, 15, 1, 12, 14, 4, 6, 7 and 11 will result in the sum of square errors '0', six '1's, seven '2's, and two '3's, respectively. Then, its *MSE* and *PSNR* are coincide with Eqs. (13) and (14). The *PSNR* of all regular elements can be derived in a similar manner and will lead to the same conclusion. Comparing with the Kurup et al.'s model, piling the 16 basic structural elements to a 3D space efficiently improves the *PSNR* value from 46.85 dB to 50.79 dB with an expense of using an additional cover pixel.

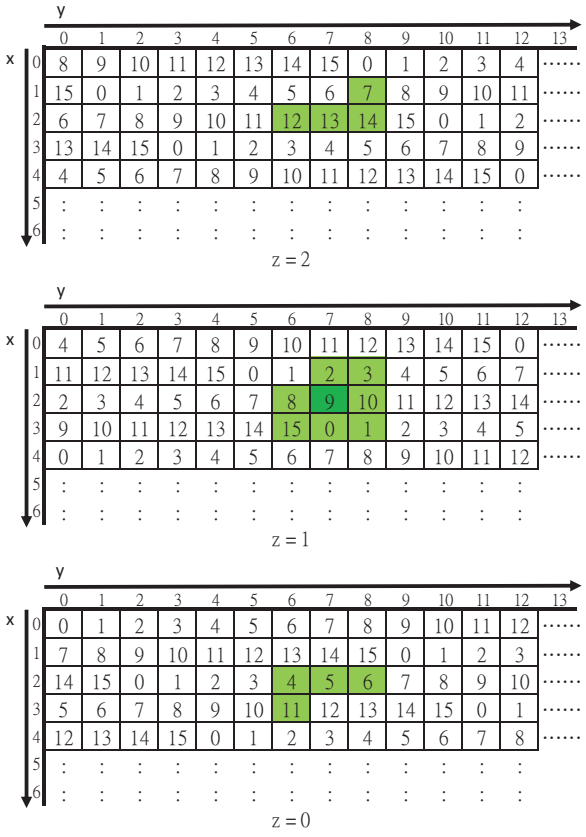


Figure 19: An example to illustrate the theoretic analysis of the *PSNR* for 2-layered turtle shell model

5.1.4 The 4-Layered Turtle Shell Model (Size 2^5)

Referring to Fig. 13, the MSE and PSNR of the 4-layered turtle shell model are given by

$$MSE = \frac{1}{3}\left(\frac{1}{32}\times 0+\frac{6}{32}\times 1+\frac{11}{32}\times 2+\frac{6}{32}\times 3+\frac{2}{32}\times 4+\frac{6}{32}\times 5\right)=\frac{84}{96}. \tag{15}$$

$$PSNR = 48.7107. \tag{16}$$

Comparing the PSNR value of the Kurup et al.'s model with the proposed 3D 2-layered and 4-layered turtle shell models, we can find that the PSNR value could be effectively improved by extending the RM to a 3D space.

5.2 Complexity

The series of models expanded from the turtle shell model discussed in the previous sub-section share the same feature that they are all based on the RM. Security of such models relies on the complexity of the RM. In this sub-section, we will analyze the periodicity of their RMs. Longer period means less repetition and, therefore, is harder to attack. In addition, the number of distinct contents depends on the radix number of that model, which also influences the complexity of the RMs. The periods of all axes and radix numbers are analyzed and summarized in Tab. 1.

Table 1: The periods and radices for different models

	T_x	T_y	T_z	Radix
Turtle	8 (p_i)	16 (p_{i+1})	—	8
Octagon	48 (x_1)	16 (x_2)	—	16
2-layered turtle	16	16	4	16
4-layered turtle	8	8	32	32

The octagon-shaped shell model has a long period on the x_1 direction of axis. According to its construction rule, the incremental values in the x_1 axis are periodic series of 4, 4 and 5. This makes a total increment of $4+4+5=13$, which is a prime number. To go back to zero, we need a total increment of 13×16, i.e., $3\times 16=48$ elements in length. As a result, we can conclude that, the RM of octagon-shaped shell model is tiled by 48×16 sub-matrices with 16 distinct contents. In the same way, the RM of the 2-layered turtle shell model is piled up by $16\times 16\times 4$ sub-blocks with 16 distinct contents. While the RM of the 4-layered turtle shell model is piled up by $8\times 8\times 32$ sub-blocks with 32 distinct contents.

6 Experimental Results

We apply the proposed 3D turtle shell models to the standard test images, including Lena, Boat, Airplane, Sailboat, Elaine, Goldhill, Peppers, and Baboon, as shown in Fig. 20. All experiments are implemented by MATLAB R2014b. Secret message S is produced by a random number generator. The resulting stego images corresponding to the 2-layered and 4-layered turtle shell models for data hiding are shown in Figs. 21 and 22. The PSNR values for full embedding are also given under the figures.

The experimental values precisely coincide with the theoretic values analyzed in Sub-section 5.1. Since the proposed hiding scheme is homogeneous throughout the RM except for the rare boundary cases and irrelevant to the cover image feature, the quality of stego image is fixed to a small random variation around the predicted theoretic value. The detailed data, including the PSNR, the embedding capacity (EC), and the structural similarity (SSIM) are listed in Tabs. 2 and 3 for 2-layered and 4-layered turtle shell model, respectively.

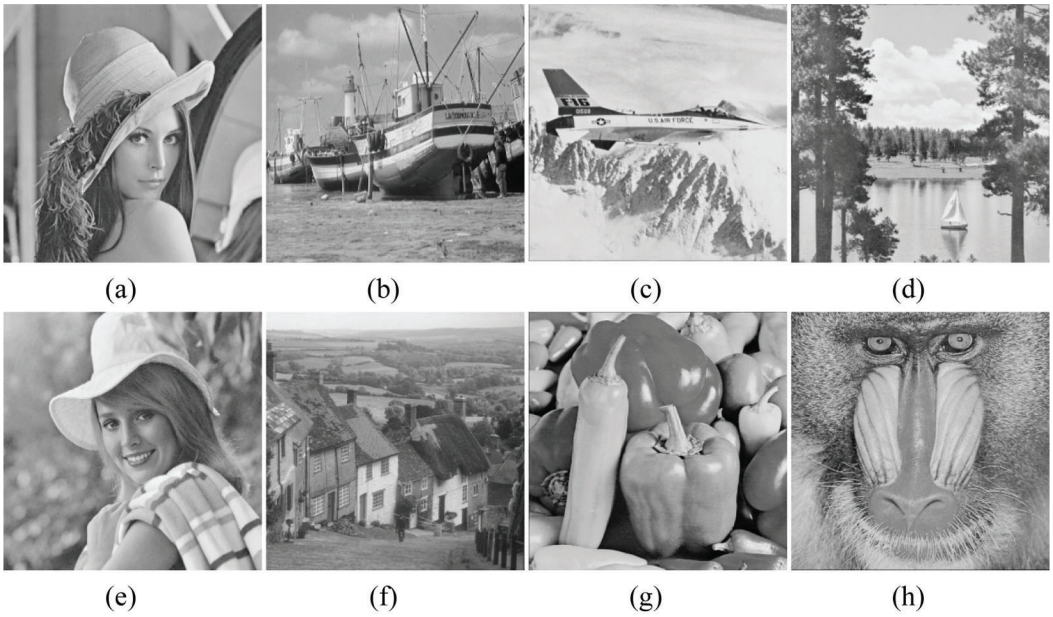


Figure 20: Eight test images with the size of 512 × 512 (a) Lena (b) Boat (c) Airplane (d) Sailboat (e) Elaine (f) Goldhill (g) Peppers (h) Baboon

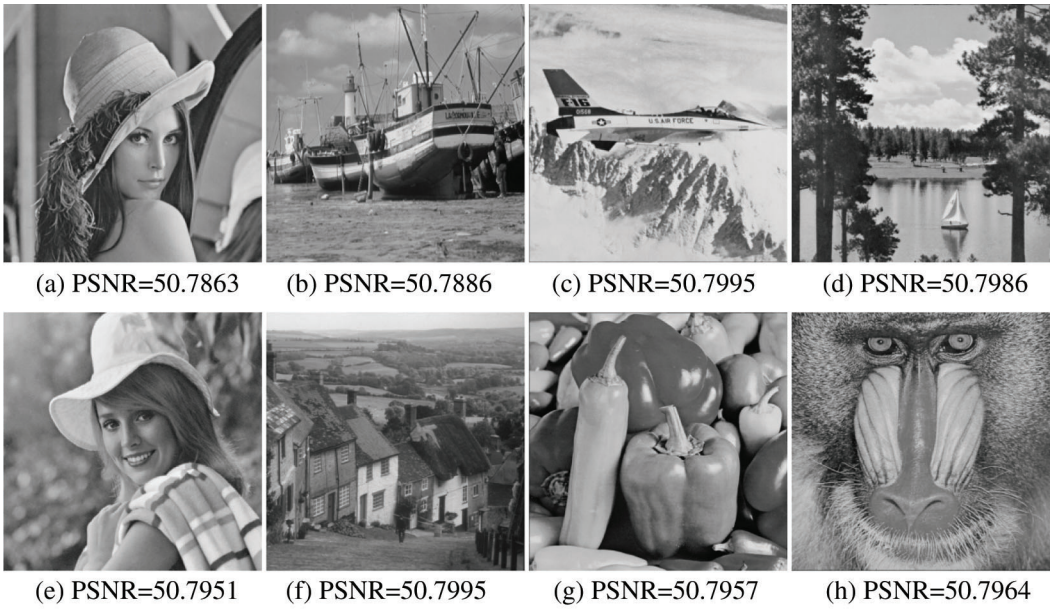
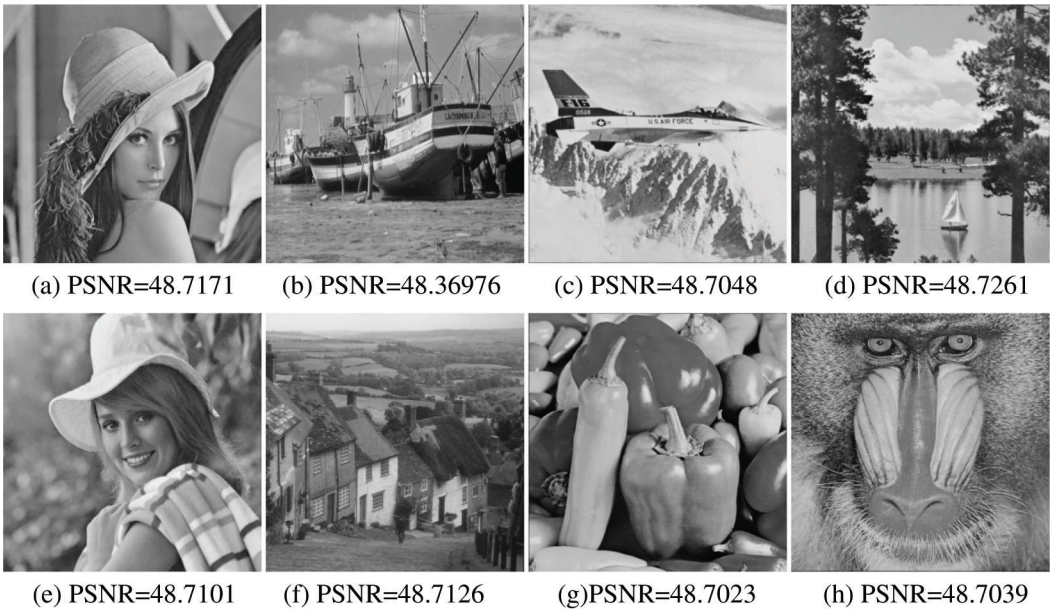


Figure 21: Eight stego images using the 2-layered turtle shell model (a) Lena (b) Boat (c) Airplane (d) Sailboat (e) Elaine (f) Goldhill (g) Peppers (h) Baboon

The definition of PSNR is the same as that in Eq. (10), while the MSE for real image applications is obtained by averaging the square error of all pixels as given in Eq. (17), where H and W are the height and width of the cover image; $p(i,j)$ and $p'(i,j)$ are the pixel values of the cover image and the corresponding stego image, respectively.



(a) PSNR=48.7171 (b) PSNR=48.36976 (c) PSNR=48.7048 (d) PSNR=48.7261

(e) PSNR=48.7101 (f) PSNR=48.7126 (g)PSNR=48.7023 (h) PSNR=48.7039

Figure 22: Eight stego images using the 4-layered turtle shell model (a) Lena (b) Boat (c) Airplane (d) Sailboat (e) Elaine (f) Goldhill (g) Peppers (h) Baboon

Table 2: Experimental results of the 2-layered turtle shell model

Image	Lena	Boat	Airplane	Sailboat	Elaine	Goldhill	Peppers	Baboon
PSNR	50.7891	50.7913	50.7931	50.7911	50.7871	50.8004	50.7998	50.7971
EC	4/3	4/3	4/3	4/3	4/3	4/3	4/3	4/3
SSIM	0.9804	0.9908	0.9519	0.9868	0.9900	0.9892	0.9869	0.9977

Table 3: Experimental results of the 4-layered turtle shell model

Image	Lena	Boat	Airplane	Sailboat	Elaine	Goldhill	Peppers	Baboon
PSNR	48.7171	48.6976	48.7048	48.7261	48.7101	48.7126	48.7023	48.7039
EC	5/3	5/3	5/3	5/3	5/3	5/3	5/3	5/3
SSIM	0.9686	0.9852	0.9287	0.9793	0.9846	0.9830	0.9788	0.9962

$$MSE = \frac{1}{H \times W} \sum_{i=1}^{H} \sum_{j=1}^{W} (p(i,j) - p'(i,j))^2. \tag{17}$$

The EC is measured in bits per pixel. The 2-layered turtle shell scheme can hide a 16-ary secret digit in 3 cover pixels, which is equivalent to 4 bits. While the 4-layered scheme can hide a 32-ary secret digit, equivalent to 5 bits, in 3 cover pixels. Therefore, their embedding capacities are 4/3 and 5/3, respectively.

In addition, we use SSIM to measure the quality of stego images. SSIM is a perception-based model that takes luminance, contrast, and structural information to evaluate similarity between two images. Let x be the cover image and y be the stego image. The individual mean values μ_x, μ_y, standard deviations σ_x, σ_y, and the covariance σ_{xy} between them are calculated firstly. Then, apply Eqs. (18)–(20) to compute the SSIM between cover image and stego image, where L is the dynamic range of the pixel values, i.e., $L = 255$, $k_1 = 0.01$, and $k_2 = 0.03$.

$$SSIM(x,y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{\left(\mu_x^2 + \mu_y^2 + c_1\right)\left(\sigma_x^2 + \sigma_y^2 + c_2\right)}, \tag{18}$$

$$c_1 = (k_1 L)^2, \tag{19}$$

$$c_2 = (k_2 L)^2. \tag{20}$$

The two variables c_1 and c_2 are applied to stabilize the division with weak denominator. The resultant SSIM index is a decimal value between −1 and 1, and value 1 is only reachable in the case of two identical images and therefore indicates perfect structural similarity. A value of 0 indicates no structural similarity.

The resulting values listed in Tabs. 2 and 3 show that the stego images are very close to their corresponding cover images. In Tab. 4, we compare the proposed models with the related works. To hide 16-ary secret digits, the 2-layered turtle shell model is better than the octagon-shaped shell model in PSNR with a trade-off in embedding capacity. Fig. 23 shows the evolution of PSNR with respect to the

Table 4: Comparisons of EC and PSNR

Factor	Turtle shell model	Octagon-shaped shell model	2-layered turtle shell model	4-layered turtle shell model
Radix	8	16	16	32
EC (bpp)	1.5	2	4/3	5/3
PSNR (dB)	50.1741	46.8476	50.7964	48.7096

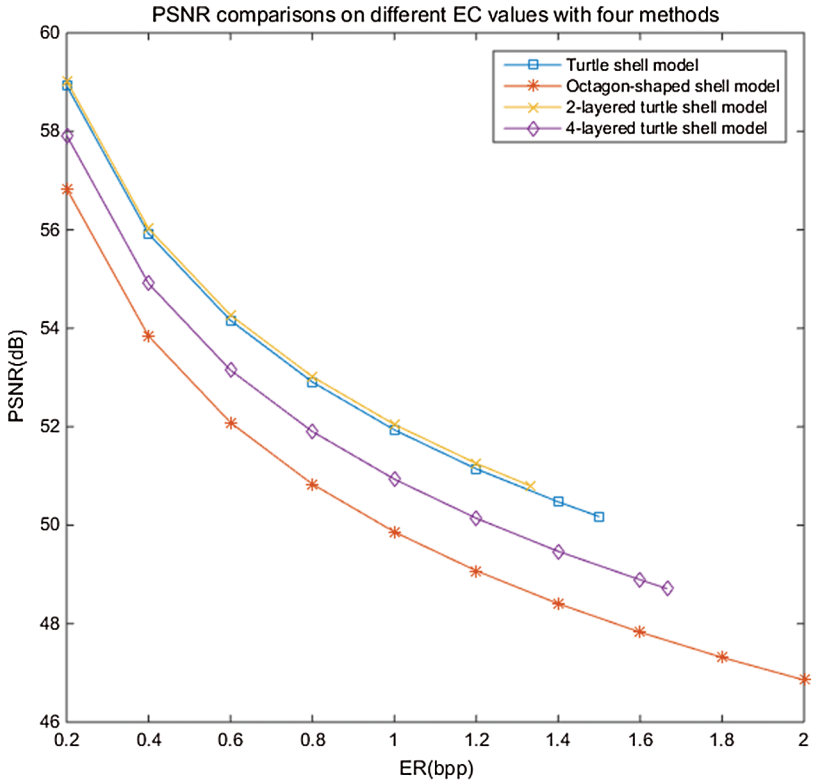


Figure 23: PSNR comparisons on different ER values

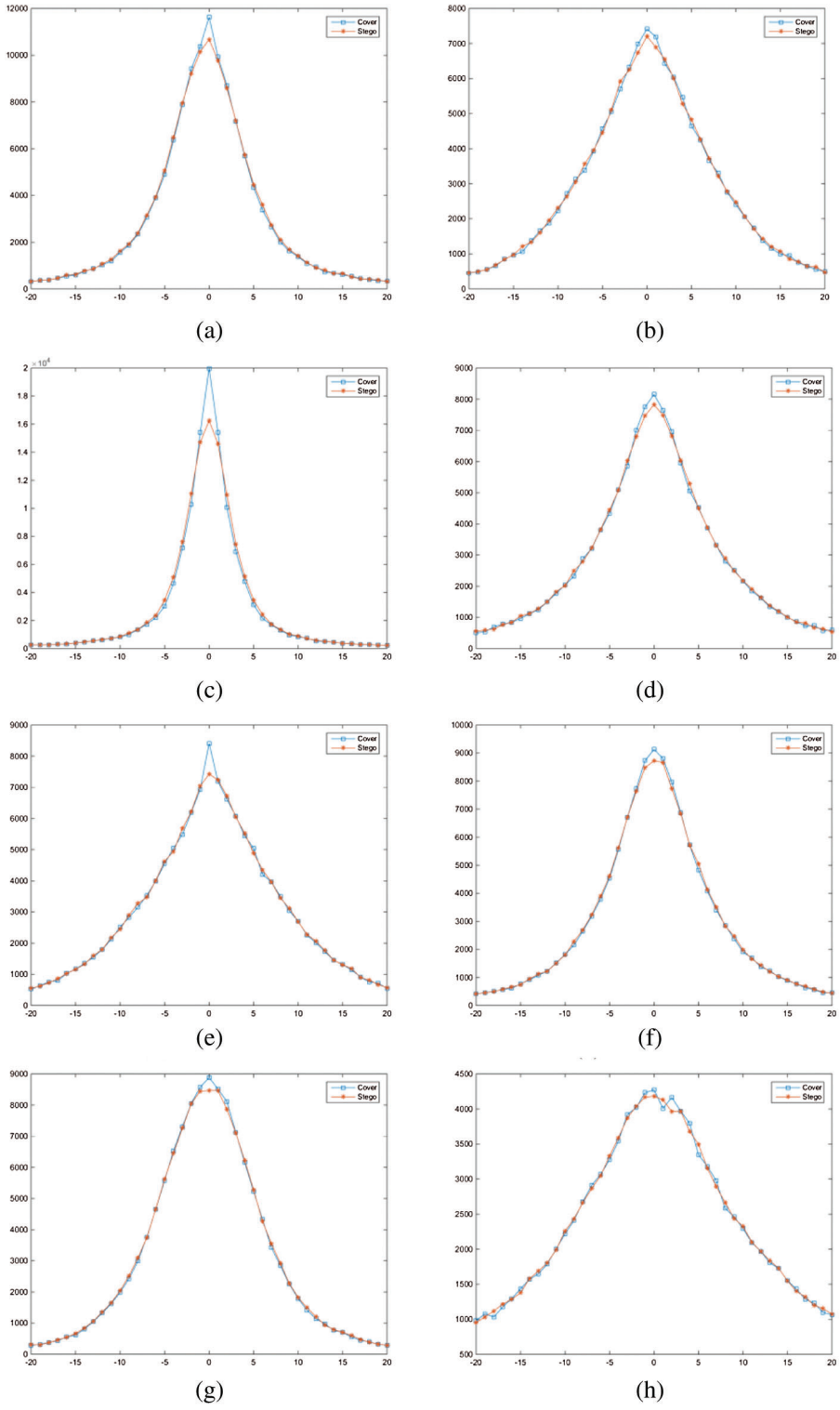


Figure 24: PDH of the 2-layered turtle shell model. (a) Lena (b) Boat (c) Airplane (d) Sailboat (e) Elaine (f) Goldhill (g) Peppers and (h) Baboon

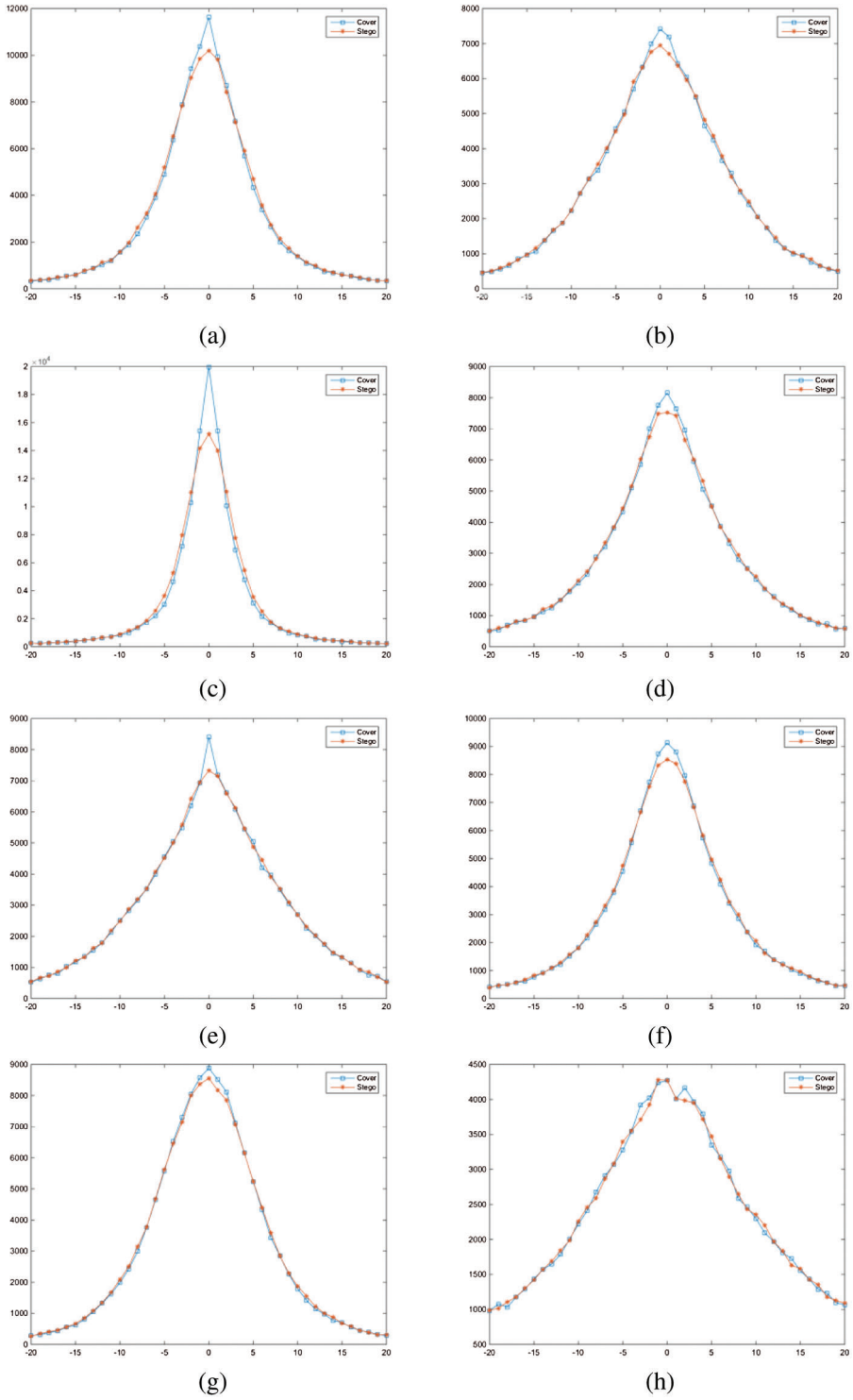


Figure 25: PDH of the 4-layered turtle shell model. (a) Lena (b) Boat (c) Airplane (d) Sailboat (e) Elaine (f) Goldhill (g) Peppers and (h) Baboon

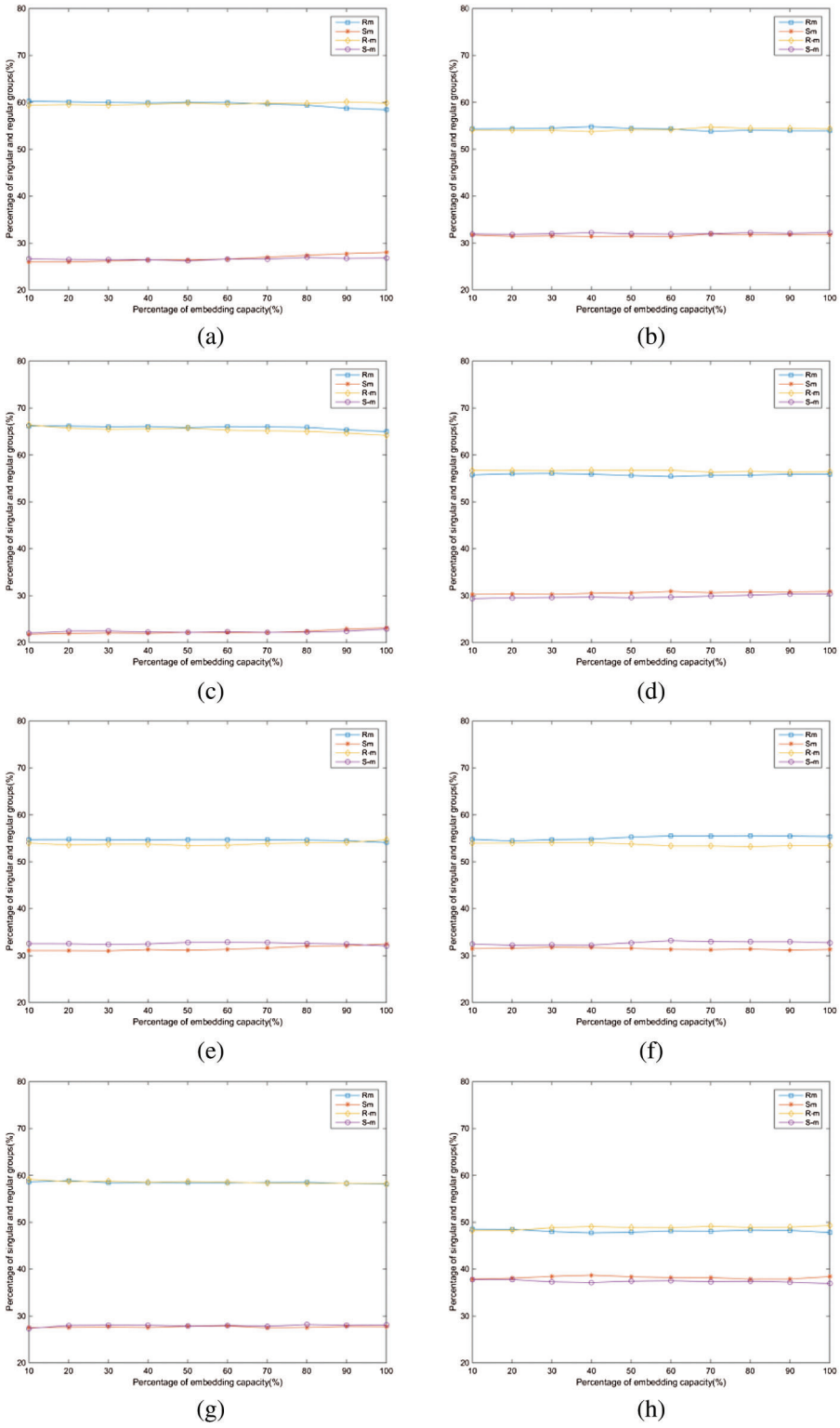


Figure 26: RS steganalysis of the 2-layered turtle shell model. (a) Lena (b) Boat (c) Airplane (d) Sailboat (e) Elaine (f) Goldhill (g) Peppers and (h) Baboon

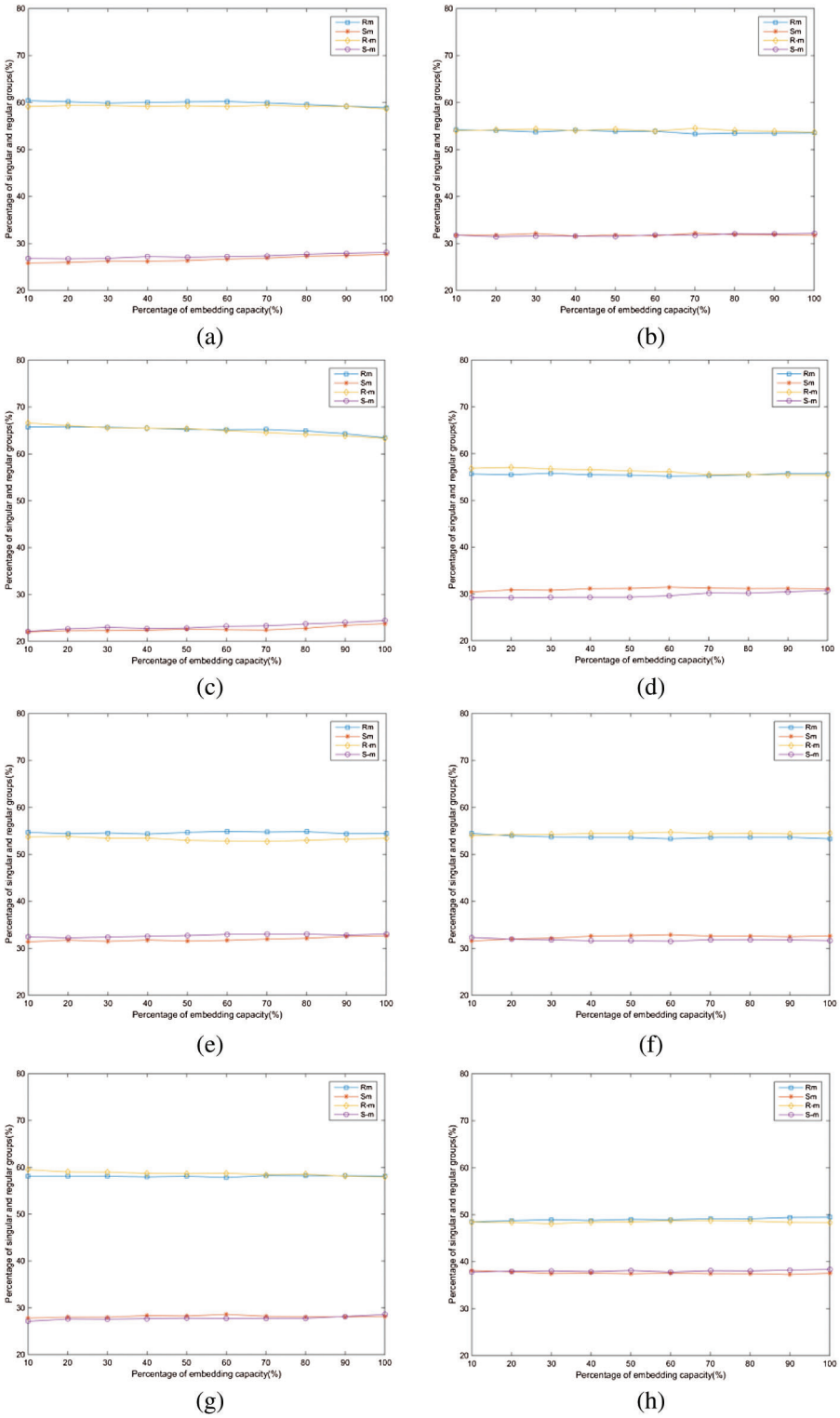


Figure 27: RS steganalysis of the 4-layered turtle shell model. (a) Lena (b) Boat (c) Airplane (d) Sailboat (e) Elaine (f) Goldhill (g) Peppers and (h) Baboon

embedding rate (ER). To make a fare comparison, the ER is defined as the ratio of currently embedded bits to the total cover pixels instead of the percentage of each model's EC.

$$ER = \frac{||S||}{H \times W}, \tag{21}$$

where H and W are the height and width of the cover image; $||S||$ is the number of embedded bits.

The PSNR decreases with increasing ER. The terminal points of each PSNR curve represents the maximum ER, i.e., the embedding capacity (EC), of the corresponding model. Under the same technical frame structure, the four curves show that there is a trade-off between PSNR and EC.

To estimate the performance of the proposed models under steganalysis, we apply the pixel-value difference histogram (PDH) analysis [21] and the RS steganalysis [22] to our stego images. The PDH for the 2-layered and 4-layered turtle shell models are given in Figs. 24 and 25. As shown in the figures, the histograms are preserved well after embedding for all cases. Figs. 26 and 27 display the RS steganalysis results for the two models. The curve R_M is very close to R_{-M} and S_M is very close to S_{-M} in all cases. It represents that the proposed models are robust to steganalysis of different techniques.

7 Conclusions

In this paper, we proposed two novel 3D turtle shell models for image steganography. By extending the reference matrix into a 3D space, more elements can be arranged within a compact space and therefore the PSNR can be greatly improved. The generalized data embedding rule provides a more accurate and efficient way to hide secret data. In addition, we proposed a theoretic analysis technique to estimate the PSNR value before experiments. The real image applications validate that our theoretic estimation can precisely predict the experimental PSNR values for all models under the same technical frame structure. The complexity analysis of RMs for different models are also given. The SSIM shows that our stego images are very close to the cover images. Finally, the PDH and RS analysis are applied to confirm the robustness of the proposed models under steganalysis.

Funding Statement: The author(s) received no specific funding for this study.

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Qin, C., Hu, Y. C. (2016). Reversible data hiding in VQ index table with lossless coding and adaptive switching mechanism. *Signal Processing, 129,* 48–55. DOI 10.1016/j.sigpro.2016.05.032.
2. Hong, W., Ma, Y. B., Wu, H. C., Chen, T. S. (2017). An efficient reversible data hiding method for AMBTC compressed images. *Multimedia Tools and Applications, 76(4),* 5441–5460. DOI 10.1007/s11042-016-4032-8.
3. Huang, F., Huang, J., Shi, Y. Q. (2012). New channel selection rule for JPEG steganography. *IEEE Transactions on Information Forensics and Security, 7(4),* 1181–1191. DOI 10.1109/TIFS.2012.2198213.
4. Langelaar, G. C., Lagendijk, R. L. (2001). Optimal differential energy watermarking of DCT encoded images and video. *IEEE Transactions on image Processing, 10(1),* 148–158. DOI 10.1109/83.892451.
5. Liu, H., Liu, J., Huang, J., Huang, D., Shi, Y. Q. (2002). A robust DWT-based blind data hiding algorithm. *IEEE International Symposium on Circuits and Systems, 2,* 672–675.
6. Lee, S., Yoo, C. D., Kalker, T. (2007). Reversible image watermarking based on integer-to-integer wavelet transform. *IEEE Transactions on Information Forensics and Security, 2(3),* 321–330. DOI 10.1109/TIFS.2007.905146.
7. Bender, W., Gruhl, D., Morimoto, N., Lu, A. (1996). Techniques for data hiding. *IBM Systems Journal, 35(3–4),* 313–336. DOI 10.1147/sj.353.0313.

8. Mielikainen, J. (2006). LSB matching revisited. *IEEE Signal Processing Letters, 13(5),* 285–287. DOI 10.1109/LSP.2006.870357.
9. Sahu, A. K., Swain, G., Babu, E. S. (2018). Digital image steganography using bit flipping. *Cybernetics and Information Technologies, 8(1),* 59–80.
10. Yang, C. H., Weng, C. Y., Wang, S. J., Sun, H. M. (2010). Varied PVD + LSB evading detection programs to spatial domain in data embedding systems. *Journal of Systems and Software, 83(10),* 1635–1643. DOI 10.1016/j.jss.2010.03.081.
11. Wu, D. C., Tsai, W. H. (2003). A steganographic method for images by pixel-value differencing. *Pattern Recognition Letters, 24(9–10),* 1613–1626. DOI 10.1016/S0167-8655(02)00402-6.
12. Sahu, A. K., Swain, G. (2019). An optimal information hiding approach based on pixel value differencing and modulus function. *Wireless Personal Communications, 108(1),* 159–174. DOI 10.1007/s11277-019-06393-z.
13. Zhang, X., Wang, S. (2006). Efficient steganographic embedding by exploiting modification direction. *IEEE Communications Letters, 10(11),* 781–783. DOI 10.1109/LCOMM.2006.060863.
14. Kim, H. J., Kim, C., Choi, Y., Wang, S., Zhang, X. (2010). Improved modification direction methods. *Computers & Mathematics with Applications, 60(2),* 319–325. DOI 10.1016/j.camwa.2010.01.006.
15. Chang, C. C., Chou, Y. C., Kieu, T. D. (2008). An information hiding scheme using Sudoku. *Third International Conference on Innovative Computing, Information and Control, Dalian, China,* 17–22.
16. Chang, C. C., Liu, Y., Nguyen, T. S. (2014). A novel turtle shell based scheme for data hiding. *Tenth International Conference on Intelligent Information and Multimedia Signal, Kitakyushu, Japan,* 89–93.
17. Kurup, S., Rodrigues, A., Bhise, A. (2015). Data hiding scheme based on octagon shaped shell. *International Conference on Advances in Computing, Communication and Informatics, Kerala, India,* 1982–1986.
18. Niu, X., Ma, M., Tang, R., Yin, Z. (2015). Image steganography via fully exploiting modification direction. *International Journal of Security and Its Applications, 9(5),* 243–254. DOI 10.14257/ijsia.2015.9.5.24.
19. Xia, B. B., Wang, A. H., Chang, C. C., Liu, L. (2016). An image steganography scheme using 3D-Sudoku. *Journal of Information Hiding and Multimedia Signal Processing, 7(4),* 836–845.
20. Leng, H. S. (2019). Generalized scheme based on octagon-shaped shell for data hiding in steganographic applications. *Symmetry, 11(6),* 760. DOI 10.3390/sym11060760.
21. Zhang, X., Wang, S. (2004). Vulnerability of pixel-value differencing steganography to histogram analysis and modification for enhanced security. *Pattern Recognition Letters, 25(3),* 331–339. DOI 10.1016/j.patrec.2003.10.014.
22. Fridrich, J., Goljan, M. (2002). Practical steganalysis of digital images. state of the art. *Proceedings of SPIE, 4675,* 1–13.