An Effective Maximum Distortion Controlling Technology in the Dual-Image-Based Reversible Data Hiding Scheme

TZU-CHUEN LU1, TING-CHI CHANG2, AND JAU-JI SHEN2

1Department of Information Management, Chaoyang University of Technology, Taichung 41349, Taiwan
2Department of Information Management, National Chung Hsing University, Taichung 40227, Taiwan

Corresponding author: Jau-Ji Shen (jjshen@nchu.edu.tw)

This work was supported in part by the Ministry of Science and Technology (MOST) under Grant 108-2221-E-324-017.

ABSTRACT The dual-image information hiding method has received considerable research attention on account of its high hiding capacity and favorable stego-image quality. The dual-image hiding technology encodes secret data across two images, requiring both stego-images to recover the correct secret data. Many factors affect the embedding capacity and image quality of the dual-image hiding method, including the nature of the hidden data, the block size, and the method by which pixels are modified. This study analyzes and encodes secret data, while controlling the degree of pixel distortion using two parameters – NC and MXD to improve the image quality. The NC parameter controls how many codes are used to re-encode a secret symbol, which is used to control the number of code combinations. MXD specifies the maximum distortion of each code combination and is used to limit image distortion. To improve the effectiveness of the encoding, a pair of digital combinations was assigned according to the frequency of the secret numeric messages. A highly uniform secret message was encoded using a small distortion code. Subsequently, the code was concealed within the cover image, generating two stego-images. The experimental results indicate that the proposed method achieved not only a better embedding rate but also a greater image quality compared with previous methods. In particular, with regard to the smooth image, the image quality of the stego-image was higher than 64.27dB when the proposed scheme was applied. Furthermore, this study utilized many steganalytic technologies to validate the security of the proposed scheme.

INDEX TERMS Center folding strategy, dual-images based hiding technique, re-encoding, reversible information hiding.

I. INTRODUCTION

In this era of rapid growth of information technology, our preferred method of exchanging information has changed from handwritten letters to electronic means via the open Internet. However, any information sent via the Internet is at risk of being destroyed or intercepted by malicious third parties. Therefore, the safe transmission of information between users has become a crucial concern and is the focus of research into information hiding technology. In addition, the nature of the transmitted content has also changed from raw text to a diverse range of digital media, such as images, videos, and audio. Digital media is now in use everywhere, with images accounting for 35% of the digital media in use. Images are not only rapidly interpreted and rich in variety, anomalies within them are not easily recognized by humans. If certain modifications to an image do not result in excessive distortion to an image, they usually go unnoticed or undetected. Consequently, images are often used as a vector for transmitting hidden information.

The evaluation index of an information hiding method depends mainly on the volume of information that can be embedded and the quality of the resulting stego-image. The success of concealing information in this way depends on achieving a balance between the two. A favorable information hiding method must not only have a high information embedding capacity, but also be imperceptible. If the media containing the concealed data is intercepted, such
a method must also make it impossible for the malicious third party to extract the secret information embedded in the media.

Information hiding technology can be divided into irreversible and reversible technology, depending on whether or not the original image can be restored. In irreversible data hiding (IDH), a stego-image cannot be restored to the original image after extraction of the secret information, because hiding causes permanent distortion, which is not permitted in professions, such as medicine, the military, and law. Nevertheless, IDH technology can embed a substantial amount of information. Conversely, in reversible data hiding (RDH), the recipient can recover the stego-image to its original state, without distortion, after extraction of the secret information; however, the RDH technology is constrained by only allowing a limited amount of information to be embedded.

To date, many RDH methods have been proposed. For example, in 2003, Tian [19] proposed the difference expansion (DE) method, wherein the gap between two adjacent pixels is expanded to hide one secret bit. However, to overcome the possible overflow/underflow problem, this method requires the generation of a location map half the size of the original image in order to determine whether the expansion has occurred. In 2004, Alattar [3] improved Tian’s method by changing the number of cover pixels to four. They set one pixel as the reference point and calculated the gap between the remaining three pixels and the reference point to hide the secret bits. Compared with Tian’s method, the embedding capacity of the method suggested by Alattar et al., increased by 0.25 bpp. In 2003, Ni et al. proposed the histogram shifting (HS) technique [31] that counts the pixels into a histogram and the secret information is hidden by shifting between the most frequent (Peak Point) and least frequent (Zero Point) pixels. Because the embedding capacity of this method depends on the total number of peak points, it can achieve more favorable results when applied to smooth images.

In 2007, Thodi and Rodriguez [11] proposed for the first time the prediction error expansion (PEE) method, which combined the DE and HS methods. They used the similarity of adjacent pixels to reduce the distortion caused by DE and to increase the embedding capacity. Many scholars have extended the PEE method and gone on to propose improved techniques, including adaptive prediction error expansion (APEE) [29], optimal prediction error expansion bins selection (O-PEE) [10], [14], adaptive optimal prediction error expansion (AO-PEE) [12], and embedding methods based on multiple histogram modification (MHM) [30]. In 2009, Tsai et al. [23] used linear prediction to generate the residual histogram. They divided the image into \( 3 \times 5, 5 \times 5 \), and \( 7 \times 7 \) non-overlapping blocks. The central pixel of the block was used as the prediction datum point. The prediction errors between other pixels and the datum point were then calculated. This method was capable of generating two peaks for embedding secret information. Compared with the method of Ni et al., the capacity of this method increased by 0.08 bpp. In 2010, Luo et al. [21] applied interpolation errors in digital watermarking to achieve reversible results and obtained a higher embedding capacity and image quality.

Dual-images technology has since become one of the most attractive hiding methods of recent years. This method uses two identical cover images to hide information. It disperses the distortion into two images, which effectively improves the image quality and the embedding capacity, thereby effectively hiding information [15], [20], [22], [24]. In addition, dual-images technology uses the concept of secret sharing, such that only the legal person who owns two stego-images can completely extract the secret information. On account of this, the security of dual-images based technology is better than that of other methods, due to the fact that because two stego-images are very similar to the original one, the attacker won’t become suspicious.

In 2007, Chang et al. developed the first reversible dual-images method by combining the method of exploiting modification direction (EMD), as discussed below, and dual-images technology [5]. They generated a 256 × 256 matrix by using a modular function for converting two secret bits into a 5-ary secret symbol in a sequence. In this method, the two original pixels form a pair and hide the left and right diagonal line of the modulus matrix. In 2009, Chang et al. [6] changed the directions of the diagonal line to horizontal and vertical, thus improving the embedding capacity. In 2015, Qin et al. [9] combined EMD with asymmetric dual-images technology using the conventional EMD method to produce the first stego-image and modifying the second stego-image based on the first stego-image. In 2009, Lee et al. [8] categorized four secret bits into groups with four corresponding directions. However, in order to achieve reversibility, the last two secret bits could only be hidden if the secret bits had opposite or clockwise directions. In 2013, Lee and Huang [7] improved their method by converting secret information into 5-ary secret symbols and defined 25 embedding rules to increase the information payload. In the same year, Chang et al. [4] integrated the modular function to modify Lee and Huang’s embedding rules to achieve higher embedding capacity. The embedding rate of Lee and Huang’s scheme is higher than that of Chang and Lu’s scheme by 1.55 bpp. In 2016, Jafar et al. [17] proposed a prediction-based method. In the first stage of this method, a simple rule was used to embed one secret bit into two stego-images. Then, for secondary embedding, the first stego-image was used as the prediction value of the second stego-image to improve the embedding capacity.

In 2015, Lu et al. [26] improved on the least significant bit matching (LSBM) method to make it reversible. They moved irreversible pixels using a regular table, while the reversible pixels remained unchanged. In recent years, many scholars improved on the LSBM method to increase its hiding capacity. For example, [1] and [2]. In [2], the authors proposed a dual-layer-based LSBM. In the first layer, each pixel hides two secret bits using an improved LSBM to generate an intermediate pixel pair (IPP). In the next layer, the IPP was...
used to hide four secret bits which could effectively improve its embedding rate.

In the 2015, Lu et al. proposed a center folding strategy [27] which suggested that folding reduces distortion of the image thus reducing the amount of secret information. In 2017, Lu et al. found that encoding decimal secret symbols according to their frequency of occurrence can improve image quality more effectively [28]. In 2017, Yao et al. [15] modified the center folding strategy, proposed by Lu et al., and expanded the parameter $K$ range of embedding rate control to $K \geq 1$. They found that at $K = 1$, the image quality can be improved. In the same year, Lu and Leng [25] also modified their method and proposed a dual-images technique based on block folding, to improve the image quality. This technique divides the original pixel into many blocks and encodes secret symbols according to their frequency of occurrence. In 2018, Yao et al. [16] proposed an adaptive dual-image method based on mobile location. They found that the maximum allowable embedding number is more crucial than the parameter $K$ of the length of secret bits. Unlike the methods in [15] and [27], this method entails setting the parameters manually. This method which involves choosing the optimal parameters according to the required embedding rate, has shown to improve work efficiency.

Dual-images technology has been found to have a favorable effect on the ability to embed information and an improvement in image quality. In this current study, we extended the concept of encoding, proposed by Lu et al., to create a new encoding method. Lu et al. fixed the encoding method of decimal secret symbols to two-digit numbers. In order to make the encoding method more flexible, in the current study, the number of encoding values was adjusted and the image quality was controlled using the overall error to effectively reduce the distortion of stego-images and achieve more obvious effects according to the frequency of the decimal secret symbols.

The rest of this paper is organized as follows: Chapter II presents dual-images-related hiding techniques, Chapter III introduces our proposed method, Chapter IV presents the experimental results, and Chapter V presents the conclusions.

II. RELATED WORK
A. DIRECTIONAL-REVERSIBLE DUAL-IMAGES TECHNIQUE [8]

In 2009, Lee et al. proposed a directional-reversible dual-images technique. They designed a cross pattern with four directions; the two original pixels formed a set of pixel pairs, $(P_1, P_2)$, which were set as the center of the cross pattern. The other direction positions were modified according to different secret messages, as shown in Fig. 1. When the secret message is $\{0, 0\}$, the camouflage pixel becomes $(P_1 + 1, P_2)$; when the message is $\{1, 0\}$, the camouflage pixel becomes $(P_1, P_2 + 1)$.

Each of the four secret bits is a set of $m = \{m_1, m_2, m_3, m_4\}$. The first stego-image hides the two secret bits, $m_1, m_2$, according to the corresponding coordinate position, as shown in Equation (1). To achieve reversibility, the corresponding positions of the last two secret bits, $\{m_3, m_4\}$, must be opposite or clockwise to the direction of $m_1, m_2$. Fig. 2 illustrates the case where hiding can be achieved, i.e., if the rules on the right-hand side are met, then $\{m_3, m_4\}$ is hidden in the second stego-image in the same manner, as shown in Equation (2). Fig. 2 (a) - (d) illustrates the opposite-direction case and Fig. 2 (e) - (h) illustrates the clockwise-direction case.

\[
(P'_1, P'_2) = \begin{cases} 
(P_1 + 1, P_2), & \text{if } m_1m_2 = 00, \\
(P_1, P_2 - 1), & \text{if } m_1m_2 = 01, \\
(P_1, P_2 + 1), & \text{if } m_1m_2 = 10, \\
(P_1 - 1, P_2), & \text{if } m_1m_2 = 11. 
\end{cases} 
\]  \tag{1}

\[
(P''_1, P''_2) = \begin{cases} 
(P_1 + 1, P_2), & \text{if } m_3m_4 = 00, \\
(P_1, P_2 - 1), & \text{if } m_3m_4 = 01, \\
(P_1, P_2 + 1), & \text{if } m_3m_4 = 10, \\
(P_1 - 1, P_2), & \text{if } m_3m_4 = 11. 
\end{cases} 
\]  \tag{2}

In the above, $(P'_1, P'_2)$ is the first camouflage pixel and $(P''_1, P''_2)$ is the second camouflage pixel. Assuming the original pixel pair, $(P_1, P_2)$, is $(156, 134)$, the set of secret bits $m$ is $\{1101\}$, and the first two secret bits, $\{m_1, m_2\}$, which correspond to the modification rule, $(P_1 - 1, P_2)$, are $\{11\}$, then the first camouflage pixel is changed to $(155, 134)$. The last two secret bits, $\{m_3, m_4\}$, are $\{01\}$, which is neither in an opposite nor clockwise direction, so the second camouflage pixel remains unchanged.
TABLE 1. Embedding rules proposed by Lee and Huang.

| $d_1$ | $P'_1$ | $P'_2$ | $d_2$ | $P''_1$ | $P''_2$ |
|-------|--------|--------|-------|---------|---------|
| 0     | $P_1$  | $P_2$  | 1     | $P_1$   | $P_2$   |
|       | $P_1$  | $P_2$  | 2     | $P_1$   | $P_2$ +1|
|       | $P_1$  | $P_2$  | 3     | $P_1$ +1| $P_2$   |
|       | $P_1$  | $P_2$  | 4     | $P_1$ +1| $P_2$ -1|
| 1     | $P_1$ -1| $P_2$  | 1     | $P_1$   | $P_2$   |
|       | $P_1$  | $P_2$  | 2     | $P_1$ +1| $P_2$ +1|
|       | $P_1$  | $P_2$  | 3     | $P_1$ +1| $P_2$   |
|       | $P_1$  | $P_2$  | 4     | $P_1$ +1| $P_2$ -1|
| 2     | $P_1$ -1| $P_2$ +1| 1     | $P_1$   | $P_2$ +1|
|       | $P_1$  | $P_2$  | 2     | $P_1$ +1| $P_2$ +1|
|       | $P_1$  | $P_2$  | 3     | $P_1$ +1| $P_2$   |
|       | $P_1$  | $P_2$  | 4     | $P_1$ +1| $P_2$ -1|
| 3     | $P_1$ +1| $P_2$ +1| 1     | $P_1$   | $P_2$   |
|       | $P_1$  | $P_2$  | 2     | $P_1$ +1| $P_2$ +1|
|       | $P_1$  | $P_2$  | 3     | $P_1$ +1| $P_2$   |
|       | $P_1$  | $P_2$  | 4     | $P_1$ +1| $P_2$ -1|
| 4     | $P_1$ +1| $P_2$ -1| 1     | $P_1$   | $P_2$   |
|       | $P_1$  | $P_2$  | 2     | $P_1$ +1| $P_2$ +1|
|       | $P_1$  | $P_2$  | 3     | $P_1$ +1| $P_2$   |
|       | $P_1$  | $P_2$  | 4     | $P_1$ +1| $P_2$ -1|

B. RDH SCHEME BASED ON DUAL STEGANO-IMAGES BY USING ORIENTATION COMBINATIONS [7]

In 2013, Lee and Huang improved Lee’s method by proposing a set of embedding rules, as shown in Table 1. They used five secret bits in a set of $m = \{m_1, m_2, m_3, m_4, m_5\}$, and converted the set to a decimal secret symbol $d_i$ if $d \leq 24$, $d$ would be converted into two 5-ary secret symbols, $d_1$ and $d_2$. Assuming the original pixels, $P_1 = 34$ and $P_2 = 91$, and the secret bit, $m = \{10011\}$, converts $d$ to 5-ary number $d = (34)_5$, we can obtain $d_1 = (3)_5$ and $d_2 = (4)_5$. According to Table 1, the corresponding rule of $d_1$ is $(P_1 + 1, P_2 + 1)$; therefore, the first camouflage pixel is modified to $P'_1 = 34 + 1 = 35$ and $P'_2 = 91 + 1 = 92$. The corresponding rule of $d_2$ is $(P_1 - 1, P_2 + 1)$; hence, the second camouflage pixel is changed to $P''_1 = 34 - 1 = 33$ and $P''_2 = 91 + 1 = 92$.

C. DUAL-IMAGES REVERSIBLE TECHNOLOGY BASED ON FREQUENCY RECODING [28]

Lu et al. upgraded the center-folding strategy [24], proposed in 2015, and proposed a dual-image reversible technology based on frequency recoding. The center-folding strategy involves reducing the decimal secret symbols to achieve superior image quality. Lu et al. found that the secret symbols can be sorted according to occurrence frequency and that the secret symbols with larger occurrence frequencies can be set with less distortion coding to improve their image quality. Similar to [27], the secret bits are a set of $m = \{m_1, m_2, m_3, \ldots, m_K\}$, which is converted to the decimal secret symbol $d$. The secret symbols are folded in the center number, in the range, $2^{K-1}$, as shown in Equation (3). Assuming $K = 3$, the center of the value range is $2^{K-1} = 2^{3-1} = 4$, and the result is shown in Fig. 3.

$$d = d - 2^{K-1}. \ (3)$$

In Equation (3), $d$ is the original decimal secret symbol and $\hat{d}$ is the folded secret symbol. The folded secret symbol $\hat{d}$ is sorted according to the occurrence frequency. Then, the scheme encodes the ordered folded symbols. Assuming $K = 3$, for example, the occurrence frequency of secret symbols is shown in Table 2.

In Table 2, $F(\hat{d})$ is the number of occurrences of secret symbols, $O(\hat{d})$ is the result of sorting according to the occurrence frequency, and $\hat{d}'$ is the folded encoding which has been reassigned based on the occurrence frequency. The number of $\hat{d}'$ can be obtained using Equation (4):

$$\hat{d}' = \begin{cases} \left\lfloor \frac{O(\hat{d})}{2^K} \right\rfloor, & \text{if } O(\hat{d}) \text{ is an odd number}, \\ \left\lceil \frac{O(\hat{d})}{2^K} \right\rceil, & \text{otherwise}. \end{cases} \ (4)$$

Finally, $\hat{d}'$ is split into two parts and embedded into two identical cover images to generate the stego-images.

$$\begin{align*}
Sd_1 &= \hat{d}' \mod 2^1, \\
Sd_2 &= \hat{d}' \div 2^1,
\end{align*} \ (5)$$

$$\begin{align*}
P'_1 &= P_1 + Sd_1, \\
P'_2 &= P_2 - Sd_2. \ (6)$$

In Equation (6), $P_i$ is the original pixel and $P'_1$ and $P'_2$ are the camouflage pixels with embedded secret symbols. The secret symbol is divided into two digital symbols, $sd_1$ and $sd_2$. In the embedding process, if $P_i > 256 - 2^{K-1}$ or $P_i < 2^{K-1}$, overflow or underflow problems may arise, and the pixels become non-embeddable. The camouflage pixels are set as $P'_i = P_i$. 

FIGURE 3. Center folding strategy.

TABLE 2. Recoding results according to occurrence frequency.

| $d$ | $\hat{d}$ | $F(\hat{d})$ | $O(\hat{d})$ | $\hat{d}'$ |
|-----|--------|-------------|-------------|---------|
| 7   | 3      | 879,232     | 0           | 0       |
| 0   | -4     | 787,558     | 1           | 1       |
| 6   | 2      | 90,853      | 2           | -1      |
| 3   | -1     | 90,853      | 3           | 2       |
| 4   | 0      | 90,362      | 4           | -2      |
| 1   | -3     | 90,362      | 5           | 3       |
| 5   | 1      | 34,203      | 6           | -3      |
| 2   | -2     | 33,713      | 7           | 4       |
D. IMPROVED DUAL-IMAGES RDH METHOD BY USING THE SELECTION STRATEGY OF SHIFTABLE PIXELS [15]

In 2017, Yao et al. found that Lu’s method presented a modified case that was not to be used; as a result of this, they proposed an improved method. Assuming that in Lu’s method, $K = 2$ and the folded secret symbol $\hat{d} = \{-2, -1, 0, 1\}$, and taking $d = -2$ as an example, $d$ is split into $-1$ and $-1$. The method modifies the pixel $P_i$ with $(P_i - 1, P_i - 1)$. The modified cases of pixel $P_i$ are $(P_i - 1, P_i - 1), (P_i - 1, P_i), (P_i, P_i), (P_i, P_i + 1)$. The case $(P_i + 1, P_i - 1)$ is unused. The modified cases are illustrated in Fig. 4.

Fig. 5 presents the flowchart of the scheme proposed by Yao et al. Similar to the method of Lu et al., the scheme of Yao et al. converts the secret bits $m$ into decimal secret symbol $d$ in a group of $K$ bits. If the secret symbol is $d = 2^K - 1$, which is the maximum value in the range, then the scheme can hide one more secret bit in the image.

For example, let us suppose that the original pixel is $P_i = 45$ and the secret message is $m = \{111\}_2$. When $K = 2$, the decimal secret symbol is $d = 3\{1\}_2 = 3\{3\}_{10}$. Because the secret message is 3, which is equal to the maximum value, $2^K - 1$, the scheme can hide one extra bit $3\{1\}_2 = 3\{1\}_{10}$, and $d$ is updated to $3 + 1 = 4$. According to Equation (7), $d$ is an even number, so the camouflage pixels are $P_i' = 45 + \lfloor d/4 \rfloor = 45 + 1 = 46$ and $P_i'' = 46 - \lfloor d/2 \rfloor = 46 - 2 = 44$.

Yao et al. expanded the range of the embedding rate to $K \geq 1$, so that superior image quality can be obtained when $K = 1$.

$$\begin{align*}
P_i' &= P_i + \lfloor d/4 \rfloor, & \text{if } d \text{ is an even number} \\
P_i'' &= P_i - \lfloor d/2 \rfloor, & \text{if } d \text{ is an odd number}
\end{align*}$$

E. REVERSIBLE DUAL-IMAGE-BASED DATA HIDING SCHEME BY USING BLOCK FOLDING [25]

The center folding method divides the secret symbol range into two sections, as shown in Fig. 3. In 2017, Lu et al. modified the center folding method and proposed a reversible multi-block folding technique for dual images. This method divides the original image into several blocks. Each block has $B$ pixels such that $BK = \{BK_1, BK_2, \ldots, BK_B\}$. The encoding process of the secret bits is shown in Fig. 6, according to the occurrence frequency of the secret symbol. Assume that $K = 3$, the bit string with 3 bits is converted into a decimal secret symbol $d$, and $d$ is encoded into a pair of code numbers according to the encoding table. In the code pair, $(I, d)$, $I$ denotes the segment number and $d$ denotes the folded secret symbol. Assume that the secret symbol is divided into two segments, $I \in \{0, 1\}$, and each segment has 5 numbers, $d \in \{-2, -1, 0, 1, 2\}$, then the pairs of $(I, d)$ are $(0, -2), (0, -1), (0, 0), (0, 1), (0, 2), (1, -2), (1, -1), (1, 0), (1, 1), (1, 2)$. According to the degree of image distortion, the order of the encoding, from small to large, is $(0, 0), (0, 1), (0, -1), (1, 0), (1, 1), (1, -1), (0, 2), (0, -2), (1, 2), (1, -2)$. Therefore, the most frequently occurring secret symbol is recoded with the least distortion coding pair $(0, 0)$, and the second most frequent symbol is coded with $(0, 1)$. Taking Fig. 6 as an example, $d = 6$ is the third most frequent value, and the encoded pair is $(0, -1)$. The encoding table can be generated using Equations (9) and (10), where $F(d)$ is the number of occurrences of $d$, and $O(d)$ is sorted in decreasing order according to $F(d)$.

$$\hat{d} = \text{sign}(O(d)) \times \left\lceil \frac{O(d)}{4} \right\rceil,$$  (9)
Each block $BK$ can hide $(B−1)$ code pairs, $(I_1, \hat{d}_1), (I_2, \hat{d}_2), \ldots, (I_{B−1}, \hat{d}_{B−1})$; the scheme collects the first digit $I$ from each code pair to form a decimal digit $(ID)_{10} = (I_1, I_2, \ldots, I_{B−1})$. $ID$ is split into two parts and hidden in the last pixel, $BK_B$, to obtain $BK'_B$ and $BK''_B$ by using Equations (12) and (13) and then the other numbers are embedded in other pixels, $BK_1, BK_2, \ldots, BK_{B−1}$, by using Equations (14) and (15). The example given in Fig. 6 assumes that two decimal numbers, 6 and 3, are hidden in the pixel block $BK = [162, 150, 130]$. The two code pairs are $(0, −1)$ and $(1, 1)$. The scheme collects the first digits, $I_1$ and $I_2$, to form $ID = (01)_2 = (1)_{10}$ and splits it into two parts to hide it in the last pixel $BK_B$. $BK_3$ is modified to generate $BK'_{B} = 130 + \lfloor 1/2 \rfloor = 130$ and $BK''_{B} = 130 − \lfloor 1/2 \rfloor = 129$. The folded secret symbols, $\hat{d}_1 = −1$ and $\hat{d}_2 = 1$, are hidden in $BK_1$ and $BK_2$ to obtain two stego-images. The first camouflage pixel block is modified to $BK'_1 = 162 + \lfloor -1/2 \rfloor = 161$ and $BK''_1 = 150 + \lfloor 1/2 \rfloor = 150$, and the second camouflage pixel block is modified to $BK'_2 = 162 − \lfloor -1/2 \rfloor = 162$ and $BK''_2 = 150 − \lfloor 1/2 \rfloor = 149$. The embedding process is illustrated in Fig. 7.

### III. PROPOSED METHOD

#### A. EMBEDDING PROCESS

Lu et al. re-encoded the secret symbol into code pairs, $(I, \hat{d})$, where $\hat{d}$ is a folded value. However, when the secret symbol was large, the $\hat{d}$ also was large, which seriously affected the image quality. In the present study, we variably adjusted the code and divided the secret symbols into multiple number codes to form a code combination $S_j = (Sd_1, Sd_2, \ldots, Sd_{NC})$. Two variables, number of codes ($NC$) and maximum distortion ($MXD$), were used to adjust the degree of image distortion. $NC$ represents the number of codes in the code combination and $MXD$ is the maximum distortion caused by the code combination.

The diagram of the proposed scheme is shown in Fig. 8. Two parameters, $NC$ and $MXD$, are used to generate the code combination. The scheme analyses the frequency of the secret message and re-encodes the secret data with a code combination, $S_i = (Sd_1, Sd_2, \ldots, Sd_{NC})$. The codes are split into two parts to generate two stego-images.

The process of converting a secret symbol into the code array is as follows:

1. Set the parameters, $NC$ and $MXD$.
2. Generate all code combinations, $AC$, where each code combination has $NC$ codes, and the absolute sum of all codes in each combination is smaller than $MXD$.
3. Generate $PNsigns$ which are all positive and negative sign permutations in $NC$ numbers.
4. Multiply $AC$ and $PNsigns$ to obtain all code combinations with $NC$ digits.
5. Calculate the distortion of each code combination by using Equation (16).

$$E_j = \sum_{i=1}^{NC} (Sd_i^2)$$

6. Sort all code combinations, in increasing order, according to the distortion $E_j$.

Taking the number of codes, $NC = 3$, and maximum errors, $MXD = 2$, as an example all possible combinations are shown in Table 1. Each combination satisfies the rule that the absolute sum, $E_j$, is smaller than $MXD$, where $j$ represents the number of code combinations.

7. Calculate the number of bits that can be hidden in the cover pixel, using Equation (17), where $ND$ is the number of all code combinations which are generated from step 4. If $ND$ is equal to $2^K − 1$, which is the maximum digit in number of $K$ bits, then one more secret bit can be concealed within the cover pixel, such that the hidden bit is set to be $K = \lceil \log_2 ND \rceil + 1$.

8. Use Equation (18) to obtain the maximum embeddable secret symbols, $dR$.

$$dR = 2^K − 1$$

9. Take $K$ secret bits as a bit string, $m = m_1m_2 \ldots m_K$, and then convert the bit string to a decimal secret symbol $d$. 

![Example of embedding secret symbols.](image-url)
In this example, the number of all code combinations, $ND$, is 25, which means we can hide $K = \lceil \log_2 25 \rceil = 4$ bits within the cover pixel and the maximum embeddable secret symbol is $dR = 15$. The scheme collects 4 secret bits and converts the bit string into a decimal secret symbol, $d$. For example, a secret message, $m = 1111$, is converted into the decimal secret symbol, $d = (1111)_2 = (15)_{10}$.

10) Adjust the coding table size according to the total number of the secret symbols.

11) Count the occurrences $F(d)$ of each secret symbol $d$.

12) Sort the secret symbols according to $F(d)$ in decreasing order. Let $O(d)$ be the ranking of the sorting results.

13) Encode all secret symbols according to the occurrence frequency. The lowest distortion value, $E_1$, is allocated to the code combination with the highest frequency to obtain the coding table.

Let’s assume that the frequency of occurrence of $d$ is as shown in Table 4. Each secret symbol is re-encoded to one code combination. For example, the secret symbol, $d = 9$, is encoded by $(Sd_1, Sd_2, Sd_3) = (-1, 0, 0)$ . The distortion of the code is $E = \sum_{i=1}^{3} (Sd_i)^2 = 1$.

Finally, the code combinations $(Sd_1, Sd_2, \ldots, Sd_{NC})$ are split into $Sd'_i$ and $Sd''_i$, as shown in Equation (19), where $1 \leq i \leq NC$. $Sd'_i$ and $Sd''_i$ are embedded into two identical cover images, as shown in Equation (19).

In the embedding process, overflow and underflow may occur when the pixel is out of the range $[\lceil \frac{MXD}{2} \rceil, 255 − \lceil \frac{MXD}{2} \rceil]$. Hence, if the original pixel, $P_i < \lceil \frac{MXD}{2} \rceil$, or $P_i > 255 − \lceil \frac{MXD}{2} \rceil$, then the pixel is non-embeddable and no information will be embedded. The stego-pixel is equal to
TABLE 4. Coding table of NC = 3 and MXD = 2.

| Secret symbols d | P(d) | O(d) | Sd₁ | Sd₂ | Sd₃ | E' |
|------------------|------|------|-----|-----|-----|----|
| 7                | 57131| 1    | 0   | 0   | 0   | 0  |
| 8                | 50006| 2    | 0   | 0   | 1   | 1  |
| 11               | 48572| 3    | 0   | 1   | 0   | 1  |
| 10               | 47234| 4    | 1   | 0   | 0   | 1  |
| 9                | 45067| 5    | -1  | 0   | 0   | 1  |
| 6                | 42197| 6    | 0   | -1  | 0   | 1  |
| 5                | 37769| 7    | 0   | 0   | -1  | 1  |
| 4                | 33182| 8    | 0   | 1   | 1   | 2  |
| 3                | 28717| 9    | 1   | 0   | 1   | 2  |
| 12               | 27591| 10   | 1   | 1   | 0   | 2  |
| 2                | 22396| 11   | -1  | 0   | 1   | 2  |
| 1                | 18143| 12   | -1  | 1   | 0   | 2  |
| 13               | 17074| 13   | 0   | -1  | 1   | 2  |
| 0                | 16652| 14   | 1   | -1  | 0   | 2  |
| 15               | 16312| 15   | 0   | 1   | -1  | 2  |
| 14               | 16244| 16   | 1   | 0   | -1  | 2  |

The original pixel, \( P'_i = P''_i = P_i \),

\[
\begin{align*}
Sd'_i &= \left\lfloor \frac{Sd_i}{2} \right\rfloor, \\
Sd''_i &= \left\lfloor \frac{Sd_i}{2} \right\rfloor, \\
P'_i &= P_i + Sd'_i, \\
P''_i &= P_i - Sd''_i.
\end{align*}
\]

Following the same example, the secret symbol, \( d = 11 \), is re-encoded by \( (Sd_1, Sd_2, Sd_3) = (0, 1, 0) \). Let’s assume that the cover pixels are \( \{162, 150, 130\} \) and the first code, \( Sd_1 \), is split into \( Sd'_1 = \left\lfloor \frac{Sd_1}{2} \right\rfloor = 0/2 = 0 \) and \( Sd''_1 = \left\lfloor \frac{Sd_1}{2} \right\rfloor = 1 \), then, the codes are concealed within the first pixel to generate two camouflage pixels, \( P'_1 = 162 + 0 = 162 \) and \( P''_1 = 162 - 0 = 162 \). The second code, \( Sd_2 = 1 \), is split into \( Sd'_2 = \left\lfloor \frac{Sd_2}{2} \right\rfloor = 1/2 = 0 \) and \( Sd''_2 = \left\lfloor \frac{Sd_2}{2} \right\rfloor = 1/2 = 1 \), and the codes are hidden in the second pixel to obtain \( P'_2 = 150 + 0 = 150 \), \( P''_2 = 150 - 1 = 149 \), and so on. The embedded stego-images are \( P' = \{162, 150, 130\} \) and \( P'' = \{162, 149, 130\} \).

The variables, NC, MXD, and \( O(d) \) are encrypted with encryption keys as additional messages. Finally, the two stego-images and additional messages are sent to the receiver for data extraction and recovery.

**B. EXTRACTION AND RECOVERY**

The receiver will receive two stego-images, the encrypted additional message, and the encryption key. The encryption key is used to decrypt the encrypted additional message. The parameters that can be obtained include the two parameters, NC and MXD, as well as the order of each secret symbol, \( O(d) \), all of which are used to create the code table. The extraction and the recovery procedure are used to extract the concealed information and to restore the original image.

Fig. 9 is the diagram of the extraction process. The receiver computes the distance between two stego-pixels in the first and the second images to obtain the secret codes, \( (Sd_1, Sd_2, \ldots, Sd_{NC}) \). The scheme maps the secret codes to the code table in order to obtain the secret symbols. The secret symbols are then concatenated and transformed to form the original secret message. Then, the scheme calculates the average value of the two stego-images in order to recover the original pixel.

The extraction and recovery procedure are given below.

1) Generate the coding table by using NC, MXD, and \( O(d) \).

2) Divide the two stego-images into several blocks with NC pixels. The blocks of the first and the second stego-image are \( P' = \{P'_1, P'_2, \ldots, P'_{NC}\} \) and \( P'' = \{P''_1, P''_2, \ldots, P''_{NC}\} \), respectively.

3) Check whether the pixel is embeddable or not. If \( P'_i = P''_i \) and both \( P'_i \) and \( P''_i \) are not in the range, \( \left[ \frac{\text{MXD}}{2}, 255 - \frac{\text{MXD}}{2} \right] \), then \( P'_i \) and \( P''_i \) are non-embeddable. If the pixel is non-embeddable, then there is no message concealed in the pixels. The original pixel is equal to the stego-pixel, and the scheme skips the pixel. Go to Step 2 to find the next pixel until the block has NC pixels.

4) If \( P'_i \) and \( P''_i \) are embeddable, the hidden secret codes \( Sd_i \) can be extracted using Equation (21). Assume that the camouflage pixels are \( P'_1 = 150 \) and \( P''_1 = 149 \), then the extracted message is \( Sd_1 = 150 - 149 = 1 \).

\[
Sd_i = P'_i - P''_i
\]

5) Repeat Step 3 until all the codes in the block \( (Sd_1, Sd_2, \ldots, Sd_{NC}) \) are extracted.

6) Accede to the coding table and the extracted code combination \( (Sd_1, Sd_2, \ldots, Sd_{NC}) \) to obtain the secret symbol \( d \).

7) Convert \( d \) into a binary string and concatenate all extracted binary strings to obtain the original secret message.

8) Restore the original pixel by averaging the values of \( P'_i \) and \( P''_i \), as shown in Equation (22).

\[
P_i = \left\lfloor \frac{P'_i + P''_i}{2} \right\rfloor
\]

9) Repeat Step 2 to Step 7 until all pixels are extracted and recovered.

For example, in the camouflage blocks, \( P' = \{162, 150, 130\} \) and \( P'' = \{162, 149, 130\} \), the first pixels of two images are \( P'_1 = 162 \) and \( P''_1 = 162 \). Both of the pixels are in the value range \( \left[ \frac{\text{MXD}}{2}, 255 - \frac{\text{MXD}}{2} \right] \) = \( \left[ \frac{2}{2}, 255 - \frac{2}{2} \right] \) = \( \{1, 254\} \). Hence, the pixels are embeddable. Equation (21) is used to extract the secret code, \( Sd_1 = P'_1 - P''_1 = 162 - 162 = 0 \). The other two pixels are embeddable, hence, the secret codes are \( Sd_2 = P'_2 - P''_2 = \).
150 − 149 = 1 and $Sd_3 = P_3' - P_3'' = 130 - 130 = 0$. The extracted code array is $(Sd_1, Sd_2, Sd_3) = (0, 1, 0)$. Assuming the coding table is as represented in Table 2, the corresponding secret symbol of $(Sd_1, Sd_2, Sd_3) = (0, 1, 0)$ is $d = 11$. The binary string of $d = 11$ is $(1011)_2$.

In the recovery procedure, the mean value of the camouflage pixels is the original pixel. Therefore, the original pixels are $P_1 = \lceil (162 + 162)/2 \rceil = 162$, $P_2 = \lceil (150 + 149)/2 \rceil = 150$, and $P_3 = \lceil (130 + 130)/2 \rceil = 130$.

**IV. EXPERIMENTAL RESULTS**

In this study, we used eight $512 \times 512$ grayscale images, namely, Airplane, Baboon, Boat, Logo, Lena, Male, Peppers, and Tiffany, as shown in Fig. 10, to test the performance of the proposed scheme. In the first experiment, various $NC$ and $MXD$ values were tested to determine the appropriate setting of the proposed scheme. In the second experiment, the proposed scheme was compared with other methods. In the third experiment, the feasibility and security of the proposed scheme were evaluated through regular and singular group (RS) steganalysis.

Bits per pixel (bpp) and peak signal-to-noise ratio (PSNR) were used to evaluate the embedding capacity and image quality, respectively. The following was used to calculate bpp:

$$\text{bpp} = \frac{\text{capacity}}{2 \times h \times w},$$

where capacity is the number of embedded bits and $h \times w$ is the size of the cover image. Because we used two cover images, the embedding capacity was divided by 2 to calculate the average embedding capacity. A higher number of bpp indicates more hiding payload. PSNR is inferred using the mean square error (MSE) of the cover image and the stego-images. PSNR is computed as follows:

$$\text{MSE} = \frac{\sum_{i=1}^{h \times w} (P_i - P_i')^2}{h \times w},$$

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

In the equation, $P_i$ and $P_i'$ are the original pixel value and stego-pixel value, respectively. A high PSNR value corresponds to high stego-image quality. If the PSNR value is
TABLE 5. Experimental results of different combinations of NC and MXD.

| NC | MXD | PSNR1 | PSNR2 | PSNR3 | Capacity | bpp |
|----|-----|-------|-------|-------|----------|-----|
| 4  | 15  | 47.34 | 47.41 | 47.37 | 917,504  | 1.75|
| 5  | 15  | 50.57 | 50.77 | 50.67 | 838,848  | 1.60|
| 6  | 15  | 50.44 | 50.57 | 50.51 | 830,110  | 1.58|
| 4  | 10  | 49.36 | 49.51 | 49.43 | 786,432  | 1.50|
| 4  | 8   | 50.71 | 50.96 | 50.84 | 655,360  | 1.25|
| 7  | 10  | 52.31 | 52.77 | 52.54 | 636,633  | 1.21|
| 6  | 6   | 53.17 | 53.9  | 53.54 | 524,280  | 1.00|
| 6  | 4   | 53.9  | 54.82 | 54.36 | 393,210  | 0.75|
| 7  | 4   | 54.46 | 55.88 | 55.17 | 374,490  | 0.71|
| 4  | 2   | 54.59 | 60.03 | 57.31 | 262,144  | 0.50|

greater than 35 dB, then the modified error is difficult for the naked eye to detect.

A. COMPARISON BETWEEN DIFFERENT COMBINATIONS OF NC AND MXD

This experiment tested the performance with different NC and MXD. NC is the total number of codes in each code combination, which is used to re-encode each secret symbol. The absolute sum of each code is less than the MXD. Table 5 represents the experimental result of hiding the secret image, “Peppers,” into the cover image, “Lena.” PSNR* is the average value of two stego-images.

In this experiment, the values of NC were defined from 2 to 7, and the values of MXD were set from 2 to 15. Among them, we selected 10 sets of parameters with high performance to demonstrate the experimental results. Under the same settings of NC, if MXD is set to be high, then we can obtain more code combinations, such that the embedding rate is better. The experimental results show that the hiding capacity of (NC = 4, MXD = 10) is 1.5 bpp, and that of (NC = 4, MXD = 15) is 1.75 bpp. The parameter (NC, MXD) = (4, 15) can achieve higher performance.

NC is used to control the image quality. The experimental results show that the average PSNR values with the parameters (NC, MXD) = (4, 10) and (NC, MXD) = (7, 10) are 49.43 dB and 52.54 dB, respectively. The parameter (NC, MXD) = (7, 10) can achieve a higher image quality.

The experimental results in Table 5 were plotted as a graph, as shown in Fig. 11. We selected 6 settings with higher performance to compare with other methods, these being (4, 15), (5, 15), (7, 10), (6, 6), (7, 4), and (4, 2). Among them, the setting (NC, MXD) = (4, 15) has the highest embedding capacity, and (NC, MXD) = (4, 2) has the highest image quality.

B. COMPARISON WITH DIFFERENT METHODS

This study selected 6 parameter settings to compare with five related studies, and these have been introduced in Chapter II. In this experiment, the secret messages are converted from “Peppers” and the cover image is “Lena.”

The experimental results are shown Table 6. The proposed method has better results in terms of embedding rate and image quality. Compared with Lee (2009) [8], the image quality of (NC = 6, MXD = 6) has been improved by 0.66 dB, and the embedding rate has been improved by 0.28 bpp. Compared with Lee (2013) [7], the image quality of (NC = 7, MXD = 10) has been improved by 1.81 dB, and the embedding rate has been improved by 0.21 bpp. Compared with Lu (2017; B = 4, K = 3) [25], the image quality of (NC = 5, MXD = 15) has been improved by 2.97 dB, and the embedding rate has been improved by 0.47 bpp. Compared with Yao (2017; K = 3) [15] and Lu (2017; K = 3) [28], the average image quality of (NC = 4, MXD = 15) has been improved by 0.33 dB, and the embedding capacity has been improved by 0.23 bpp.

The detailed experimental results of different cover image experiments are shown in Table 7. The proposed scheme with (NC, MXD) = (4, 15) showed the highest embedding capacity, especially for “Airplane,” “Baboon,” “Boat,” “Lena,” “Sailboat,” and “Tiffany,” which achieved the highest embedding capacity of 1.75 bpp, and the proposed scheme with (NC, MXD) = (4, 2) yielded a higher image quality. Regardless of whether the cover image was smooth or complex, the proposed scheme achieved better hiding results than other schemes.
Table 6: Experimental results of secret image “peppers” hidden in “Lena”.

| Method | PSNR¹ | PSNR² | PSNR³ | Capacity  | bpp |
|--------|--------|--------|--------|-----------|-----|
| Propose (NC=4, MXD=2) | 54.59 | 60.03 | 57.31 | 262,144 | 0.50 |
| Propose (NC=7, MXD=4) | 54.46 | 55.88 | 55.17 | 374,490 | 0.71 |
| Propose (NC=6, MXD=6) | 53.17 | 53.9  | 53.54 | 524,280 | 1.00 |
| Lee 2009 | 51.14 | 54.61 | 52.88 | 380,108 | 0.72 |
| Propose (NC=7, MXD=10) | 52.31 | 52.77 | 52.54 | 636,633 | 1.21 |
| Lee 2013 | 51.62 | 49.83 | 50.73 | 524,288 | 1.00 |
| Propose (NC=5, MXD=15) | 50.57 | 50.77 | 50.67 | 838,848 | 1.60 |
| Lu 2017 (B=4, K=3) | 48.17 | 47.22 | 47.70 | 589,824 | 1.13 |
| Propose (NC=4, MXD=15) | 47.34 | 47.41 | 47.37 | 917,504 | 1.75 |
| Yao 2017 (K=3) | 47.03 | 47.07 | 47.05 | 807,913 | 1.54 |
| Lu 2017 (K=3) | 47.06 | 47.02 | 47.04 | 783,872 | 1.50 |

In the subsequent experiment, different secret images were hidden within the cover image, “Lena.” In this experiment, the secret image was the key factor affecting the hiding effect. We found that if the secret image was smooth, the occurrence frequency increased so that the secret symbol could effectively be encoded with small distortion codes.

Table 8 shows the detailed results, which indicates that the proposed scheme has better results than other schemes in its capacity to embed payload and its image quality, especially when the secret image is smooth, such as for “Logo,” where the results are more significant, and the image quality is 64.27 dB. Because the proposed scheme encoded the high occurrence symbols with low distortion codes, this effectively increased the image quality. The proposed scheme with \((NC, MXD) = (4, 2)\) showed the best image quality, given that the average image quality was 58.36 dB, which is higher than that of Lee (2009), Lee (2013), Lu (2017; \(K = 3\)), Yao (2017; \(K = 3\)), and Lu (2017; \(B = 4, K = 3\)) with 5.8, 8.0, 10.25, 12.4, and 9.6 dB, respectively. The proposed scheme with \((NC, MXD) = (4, 15)\) achieved the best hiding capacity of 1.75 bpp.

C. RS STEG ANALYSIS

RS steganalysis is an attack steganalysis technique proposed by Fridrich et al. [18] in 2001, wherein the pixels are divided into three categories: regular (R), singular (S), and unusable (U), and the change in pixel class is detected on the basis of the LSB. Each category represents the percentage of hidden content with two change curves, which are regular (R): \(R_MG\) and \(R_FM_G\); singular (S): \(S_MG\) and \(S_FM_G\); and unusable (U): \(U_MG\) and \(U_FM_G\). If the two curves almost overlap (i.e., \(R_MG \approx R_FM_G\) or \(S_MG \approx S_FM_G\) or \(_MG \approx U_FM_G\)), then the pixels detected contain hidden messages. Conversely, if the two change curves are separated (i.e., \(R_MG \not\approx R_FM_G\) or \(S_MG \not\approx S_FM_G\)

FIGURE 12. Experimental results of RS steganalysis.

In the experiment, we hid the secret image, “Peppers,” into “Lena” with different \(NC\) and \(MXD\) settings. We used the best image quality, \((NC, MXD) = (4, 2)\), and the best embedding rate, \((NC, MXD) = (4, 15)\), for the RS experiments. The experimental results are shown in Fig. 12. This figure shows that the change curves were almost overlapping, indicating that this method is safe under RS steganalysis.
V. CONCLUSIONS

In this study, we used the frequency of secret symbols to re-code the secret symbols. Most occurrence symbols were encoded with low distortion codes by using dual-images technology. The proposed scheme is especially useful for smooth secret images. The total number of code combinations, \( ND \), was computed according to the number of codes, \( NC \), and the maximum difference, \( MXD \). A large \( ND \) indicates that large decimal secret symbols are able to be embedded.

According to the experimental results, using the same settings of \( NC \), the proposed scheme with higher \( MXD \) performs better in embedding rate. In terms of \( NC \), the proposed scheme with high \( NC \) can achieve a better image quality.
because under the same distortion we can obtain more code combinations. However, it also indicates that the number of pixels required to hide the secret symbol is greater, which, in turn, affects the embedding capacity.

The proposed scheme has higher hiding capacity and image quality than other related schemes. Furthermore, the proposed scheme can work against RS steganalysis, and the experimental results reveal that the change curves which were obtained were almost overlapping, thus indicating that the method is safe.

ACKNOWLEDGMENT
(Tzu-Chuen Lu and Ting-Chi Chang contributed equally to this work.)
REFERENCES

[1] A. K. Sahu and G. Swain, “Dual stego-imaging based reversible data hiding using improved LSB matching,” Int. J. Intel. Eng. Syst., vol. 12, no. 5, pp. 63–73, 2019.

[2] A. K. Sahu and G. Swain, “Reversible image steganography using dual-layer LSB matching.” Sens. Imag., vol. 21, no. 1, pp. 1–21, Dec. 2020.

[3] A. M. Alattar, “Reversible watermark using the difference expansion of a generalized transform,” IEEE Trans. Image Process., vol. 13, no. 8, pp. 1147–1156, Aug. 2004.

[4] C.-C. Chang, T.-C. Lu, G. Horng, Y.-H. Huang, and Y.-M. Hsu, “A high payload data embedding scheme using dual stego-images with reversibility,” in Proc. 9th Int. Conf. Inf., Commun. Signal Process., Dec. 2013, pp. 1–5.

[5] C.-C. Chang, T. Kieu, and Y.-C. Chou, “Reversible data hiding scheme using two steganographic images,” in Proc. IEEE Region 10 Conf. (TENCON), Oct. 2007, pp. 1–4.

[6] C.-C. Chang, Y.-C. Chou, and T. D. Kieu, “Information hiding in dual images with reversibility,” in Proc. 3rd Int. Conf. Multimedia Ubiquitous Eng., Jun. 2009, pp. 145–152.

[7] C.-F. Lee and Y.-L. Huang, “Reversible data hiding scheme based on dual stego-images using orientation combinations,” Telecommun. Syst., vol. 52, no. 4, pp. 2237–2247, Apr. 2013.

[8] C.-F. Lee, K.-H. Wang, C.-C. Chang, and Y.-L. Huang, “A reversible data hiding scheme based on dual steganographic images,” in Proc. 3rd Int. Conf. Ubiquitous Inf. Manage. Commun. (ICUIMC), vol. 9, 2009, p. 228.

[9] C. Qin, C.-C. Chang, and T.-J. Hsu, “Reversible data hiding scheme based on exploiting modification direction with two steganographic images,” Multimedia Tools Appl., vol. 74, no. 15, pp. 5861–5872, Aug. 2015.

[10] C. Wang, X. Li, and B. Yang, “Efficient reversible image watermarking by using dynamical prediction-error expansion,” in Proc. IEEE Conf. Image Processing., Sep. 2010, pp. 3673–3676.

[11] D. M. Thodi and J. J. Rodriguez, “Expansion embedding techniques for reversible watermarking,” IEEE Trans. Image Process., vol. 16, no. 3, pp. 721–730, Mar. 2007.

[12] G. Xuan, X. Tong, J. Teng, X. Zhang, and Y. Q. Shi, “Optimal histogram-pair and prediction-error based image reversible data hiding,” presented at the Int. Workshop Digit. Forensics Watermarking, 2013, pp. 368–383.

[13] H. J. Hwang, “Reversible watermarking method using optimal histogram pair shifting based on prediction and sorting,” KSII Trans. Internet Inf. Syst., vol. 4, no. 4, pp. 655–670, 2010.

[14] H.-T. Wu and J. Huang, “Reversible image watermarking on prediction errors by efficient histogram modification,” Signal Process., vol. 92, no. 12, pp. 3000–3009, Dec. 2012.

[15] H. Yao, C. Qin, Z. Tang, and Y. Tian, “Improved dual-image reversible data hiding method using the selection strategy of shiftable pixels’ coordinates with minimum distortion,” Signal Process., vol. 135, pp. 26–35, Jun. 2017.

[16] H. Yao, C. Qin, Z. Tang, and X. Zhang, “A general framework for shiftable position-based dual-image reversible data hiding,” EURASIP J. Image Video Process., vol. 2018, no. 1, pp. 41, Dec. 2018.

[17] I. F. Jafar, K. A. Darabkh, R. T. Al-Zubi, and R. R. Saifan, “An efficient reversible data hiding algorithm using two steganographic images,” Signal Process., vol. 128, pp. 98–109, Nov. 2016.

[18] J. Fridrich, M. Goljan, and R. Du, “Detecting LSB steganography in color, and gray-scale images,” IEEE Multimedia Mag., vol. 8, no. 4, pp. 22–28, 2001.

[19] J. Tian, “Reversible data embedding using a difference expansion,” IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 8, pp. 890–896, Aug. 2003.

[20] K.-H. Jung, “Authenticable reversible data hiding scheme with less distortion in dual stego-images,” Multimedia Tools Appl., vol. 77, no. 5, pp. 6225–6241, Mar. 2018.

[21] L. Luo, Z. Chen, M. Chen, X. Zeng, and Z. Xiong, “Reversible image watermarking using interpolation technique,” IEEE Trans. Inf. Forensics Security, vol. 5, no. 1, pp. 187–193, Mar. 2010.

[22] L.-P. Chi, C.-H. Wu, and H.-P. Chang, “Reversible data hiding in dual stego-image using an improved center folding strategy,” Multimedia Tools Appl., vol. 77, no. 7, pp. 8785–8803, Apr. 2018.

[23] P. Tsai, Y.-C. Hu, and H.-L. Yeh, “Reversible image hiding scheme using predictive coding and histogram shifting,” Signal Process., vol. 89, no. 6, pp. 1129–1143, Jun. 2009.

[24] S. Shastrti and V. Thanikaiselvan, “Dual image reversible data hiding using trinary assignment and centre folding strategy with low distortion,” J. Vis. Commun. Image Represent., vol. 61, pp. 130–140, May 2019.

[25] T.-C. Lu and H.-S. Leng, “Reversible dual-image-based hiding scheme using block folding technique,” Symmetry, vol. 9, no. 10, p. 223, 2017.

[26] T.-C. Lu, C.-Y. Tseng, and J.-H. Wu, “Dual imaging-based reversible hiding technique using LSB matching,” Signal Process., vol. 108, pp. 77–89, Mar. 2015.

[27] T.-C. Lu, J.-H. Wu, and C.-C. Huang, “Dual-image-based reversible data hiding method using center folding strategy,” Signal Process., vol. 115, pp. 195–213, Oct. 2015.

[28] T.-C. Lu, L. P. Chi, C.-H. Wu, and H. P. Chang, “Reversible data hiding in dual stego-images using frequency-based encoding strategy,” Multimedia Tools Appl., vol. 76, no. 22, pp. 23903–23929, 2017.

[29] X. Li, B. Yang, and T. Zeng, “Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection,” IEEE Trans. Image Process., vol. 20, no. 12, pp. 3524–3533, Dec. 2011.

[30] X. Li, W. Zhang, X. Gui, and B. Yang, “Efficient reversible data hiding based on multiple histograms modification,” IEEE Trans. Inf. Forensics Security, vol. 10, no. 9, pp. 2016–2027, Sep. 2015.

[31] X. Zhang and S. Wang, “Efficient steganographic embedding by exploiting modification direction,” IEEE Commun. Lett., vol. 10, no. 11, pp. 781–783, Nov. 2006.

[32] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, “Reversible data hiding,” in Proc. Int. Conf. Image Process. (ICASSP), vol. 2, 2003, pp. II-912–II-915.

Tzu-Chuenn Lu received the B.M. and M.S.I.M. degrees in information management from the Chaoyang University of Technology, Taiwan, 1999 and 2001, respectively, and the Ph.D. degree in computer engineering from National Chung Cheng University, in 2006. She is currently a Professor at the Department of Information Management, Chaoyang University of Technology.

Ting-Chi Chang received the B.S. degree in information management from the Chaoyang University of Technology, Taichung, Taiwan, in 2018. She is currently pursuing the master’s degree in information management with Chung Hsing University, Taiwan. Her research interests include image techniques and steganography.

Jau-Ji Shen received the Ph.D. degree in computer science and information engineering from National Taiwan University, Taiwan, in 1988. He is currently a Professor of information management at Chung Hsing University, Taiwan. His research interests include image techniques, data techniques, and software engineering.