

Research Article

Improving the Security of Reversible Data Hiding Using Multiple Histogram Modification

HaiShan Chen ¹, Jinye Wang,² Yiqing Zhou,² TingCheng Chang,¹ KunQuan Shi,¹
and JunYing Yuan ²

¹Guangzhou Panyu Polytechnic, Guangdong 511483, China

²Nanfang College, Guangzhou, Guangdong 510975, China

Correspondence should be addressed to JunYing Yuan; cihisa@outlook.com

Received 29 April 2021; Revised 12 November 2021; Accepted 27 December 2021; Published 18 January 2022

Academic Editor: Nouman Ali

Copyright © 2022 HaiShan Chen et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Reversible data hiding (RDH) allows carrying secret information in cover media without introducing permanent distortion. For a RDH method, the important performance measurements are embedding capacity and image quality. Since embedding capacity is an important requirement in the field of data hiding, it is necessary to consider the security of data embedding in RDH applications. In general, RDH algorithms usually prefer data embedding in simple image regions with low local complexity. As a result, image degradation is alleviated at the cost of poor embedding security. In this study, a novel RDH method is proposed to embed data into complex image regions, wherein the data hiding becomes more secure in defending against modern steganalysis. To measure regional local complexity, the harmonic mean of directional local variances is employed to combine directional pixel differences. To embed data into complex regions instead of smooth regions, multiple histogram modification is adopted and updated for optimized data embedding with higher complexity. Experiment results show that embedding security is significantly improved with a considerable amount of payload and well-preserved image quality.

1. Introduction

Reversible data hiding (RDH) refers to embedded messages into digital contents such as images, videos, and documents of the cover media which are completely recovered after data extraction [1]. As the image is the most broadly used media, image-based RDH has been extensively studied and applied in various applications including image authentication, cloud image processing, and medical image processing.

Several RDH techniques have been proposed for histogram modification (HM) [2, 3], difference expansion (DE) [4–6], prediction-error expansion (PEE) [7–10], deep neural networks (DNN) [11, 12], and multiple histogram modification (MHM) [3, 12, 13], and so forth. In recent years, MHM has attracted considerable attention owing to its potential in revealing spatial redundancy in natural images, while DNN suggests a new research direction since DNN serves as a higher performance predictor for prediction-based RDH methods.

DNN-based RDH usually segments a cover image into two layers in a chessboard pattern by employing a deep neural network to predict one layer using the other layer. Hu and Xiang [11] employed a lightweight and computation-efficient convolution neural network (CNN) with global optimization for DNN-based data embedding. This CNN predictor was trained using 1000 randomly selected images to prove its effectiveness in increasing the prediction accuracy and improving the embedding performance. Hou et al. [12] designed a DNN-based method for dynamical multiple histograms' generation and employed MHM for optimized data embedding. While CNN is used for layer-wise image prediction [11], DNN classifies a model to establish histograms with different sizes for better redundancy exploitation [12].

MHM was first proposed by Li et al. [3]. It is regarded as a framework for optimized and content-adaptive data embedding. In this framework, several techniques are employed including pixel sorting [7], PEE, and HM. Pixel

sorting uses the strong correlation between prediction error (PE) and local complexity (LC) and prefers data embedding in pixels with lower LC. In MHM, multiple prediction-error histograms (PEHs) are constructed by evenly distributing PEs according to the level of LC. PEE and HM are combined as PEE-HM to embed information by shifting and expanding selected bins from the constructed multiple PEHs. Owing to the significantly improved embedding performance, MHM has quickly attracted research interests and been developed in various applications [12–17].

In the aforementioned RDH methods, embedding performance is usually represented by embedding capacity and image quality since the methods embed data in pixels from the flat regions. Therefore, DNN- and MHM-based RDH approaches tend to hide data in smoother regions. However, embedding security (namely, the robustness to modern steganalysis) is rarely considered, as the most important performance measurement of data hiding. As pointed out by steganography [18, 19] and image forensics [20], a pixel in a smooth region is easier to model than in a complex image region owing to compact image representation. Therefore, the behavior of data embedding in a smooth region can be more easily modeled and detected than that in complex regions.

Considering embedding security, Hong et al. [21] proposed to selectively embed secret bits into pixels with high LC. This method is robust to the detection of modern steganalysis such as histogram analysis and subtractive pixel adjacency matrix (SPAM) [22], which is one of the compact image representations as a well-known feature set in detecting the operation of data embedding [21, 22]. With SPAM, the differences between adjacent pixels are modeled as Markov chains, and the transition probability matrices are exploited to derive the statistical features. Therefore, the transition probability in the complex regions shows a small occurrence rate and thus plays a less important role in the compact image representing.

DNN has shown a prosperous future in the field of RDH. However, it is usually employed to provide higher prediction accuracy or better pixel clustering result. As a result, existing DNN-based RDH works are effective in reducing image distortion and improving the embedding capacity. As a result, DNN is still not used in the area of RDH in improving embedding security. Even though RDH considers DNN in RDH to improve the embedding security, more efforts are required before a DNN-based solution becomes effective in that direction.

This study aims to improve the embedding security of RDH in resisting statistical analysis. Specifically, the technique of MHM is exploited to embed data into complex image regions with the inspiration of Li et al.'s [3] and Hong et al.'s work [21]. Firstly, the concept of embedding cost is introduced to measure the impact to compact image representations introduced by embedding one bit into a single pixel. For data embedding in a smooth region where LC is low, the impact to compact image representations is high, and vice versa. Therefore, embedding cost can be defined as the harmonic mean of directional LC. Secondly, the overall

embedding cost for hiding a given payload is derived as the sum of embedding cost from all modified pixels. Finally, the overall embedding cost is minimized via parameter optimization, and data bits are embedded into pixels in complex regions.

The rest of this paper is organized as follows. The related works are briefed in Section 2. The proposed work is detailed in Section 3. Section 4 illustrates the evaluation of the embedding performance, and Section 5 concludes this paper.

2. Preliminaries

In this section, PEE-related techniques are briefed, including optimal PEE [23], pixel sorting [7], and MHM [3].

Let x be the grey-scale value of a cover pixel, and $X = \{x_1, x_2, \dots, x_N\}$ be the values of the cover pixel sequence, where N is the number of cover pixels and \hat{x} is the predicted value of x . The prediction error e for estimating x is computed using the following equation:

$$e = x - \hat{x}. \quad (1)$$

With the obtained PE sequence $E = \{e_1, e_2, \dots, e_N\}$, a PEH is constructed as

$$h(e) = \#\{e_i \in E: e_i = e\}, \quad (2)$$

where $\#\{\cdot\}$ denotes taking the cardinality of the given set.

2.1. Optimal Prediction-Error Expansion. Let S be the bit sequence of payload. Given S , two bins (namely, a bin-pair) a and b , where $a < b$, are obtained from $h(e)$ for data embedding at the restriction of the following equation:

$$h(a) + h(b) \geq \#\{S\}. \quad (3)$$

With a and b , a data bit $m \in \{0, 1\}$ is embedded by using equation (4), as specified by

$$e' = \begin{cases} e - 1, & e < a, \\ e - m, & e = a, \\ e, & a < e < b, \\ e + m, & e = b, \\ e + 1, & e > b, \end{cases} \quad (4)$$

where e' is the marked PE. Then, the marked pixel y can be derived as $y = \hat{x} + e'$.

When assuming S is independent and identically distributed (i.i.d), the resultant embedding distortion D can be written as

$$D \approx \frac{a+b}{2} + \sum_{e < a} h(e) + \sum_{e > b} h(e), \quad (5)$$

where D represents the number of modified pixels. Given the size of the payload $\#\{S\}$, D is minimized by optimally selecting the most proper bin-pair a^* and b^* , as specified by $(a^*, b^*) = \arg \min D(a, b)$ concerning equation (3).

2.2. Pixel Sorting. As pointed out by Sachnev et al. [7], LC is proportional to the magnitude of PE, and the lower the LC, the stronger the correlation. Therefore, the correlation among the neighboring pixels can be exploited to reduce image distortion. Specifically, pixel sorting orders cover pixels according to LC and embeds data bits into pixels with lower LC. The sorting technique is originally proposed in [24], then extended in [7], and later exploited in many PEE-based RDH methods [3, 25, 26]. The detailed sorting technique is detailed in [7].

2.3. Multiple Histogram Modification. Let $n \geq 0$ be the LC of the pixel x . Li et al.'s work [3] constructs multiple PEHs by increasing LC from low to high as specified by

$$h_n(e) = \#\{e_i \in e: e_i = e, n_i = n\}. \quad (6)$$

Let (a_n, b_n) be the to-be-expanded bin-pair selected from $h_n(e)$; the overall embedding distortion can be rewritten as

$$D = \sum_n D_n(a_n, b_n), \quad (7)$$

where

$$D_n(a_n, b_n) \approx \frac{a_n + b_n}{2} + \sum_{e < a_n} h_n(e) + \sum_{e > b_n} h_n(e). \quad (8)$$

Since n and e vary in a great range, the minimization of D under the constraint of payload size $\#\{S\}$ needs to be simplified. In Li et al.'s work [3], the computation is made simpler using two simplification techniques related to n and e , respectively. The simplifications are technically sound and the reason has been given in [3].

3. Proposed Method

In pixel sorting, LC reflects the smoothness of a neighboring image region [3, 9, 21]. For a pixel with local complexity n , a bigger n indicates a complex neighborhood, and a smaller n means a smooth neighboring region. Since pixels with smaller n take a larger portion in a compact image representation, conventional RDH methods tend to hide data into smooth regions for higher embedding capacity. Similarly, in compact image representations, the statistical features with high concurrency rates play more important roles [18, 19] than in simple image representations. As a result, data embedding in smooth regions influences compact image representation more than data embedding in complex regions. Therefore, conventional RDH methods are considered nonrobust to modern steganalysis attacks, e.g., histogram analysis and SPAM [22].

In the contrast, for a pixel with a complex neighborhood, the local statistical features impact less in compact image representations; in other words, the pixel is hard to model. Therefore, data embedding in complex regions influences less to compact image representations. For simplicity, the data embedding impact to a compact image representation is termed as embedding cost. Generally speaking, the larger n is, the smaller the embedding cost is, and vice versa. The

embedding cost shows a reciprocal relationship to n . Thus, to improve embedding security in RDH, data bits should be embedded into complex image regions, where the embedding cost is smaller [18, 19].

In this section, a novel RDH with improved robustness to modern steganalysis is presented. Firstly, embedding cost is quantified as the reciprocal of local complexity. Then, multiple PEHs are constructed by varying LC from low to high, and overall embedding cost is derived as the sum of embedding costs for modified pixels. Finally, the overall embedding cost is minimized through parameter optimization.

3.1. Complexity Measurement and Embedding Cost. Given LC measurement n , the related embedding cost, denoted by ρ , can be simply defined as the reciprocal of n , as specified by

$$\rho = n^{-1}. \quad (9)$$

Note that the larger is n , the smaller is ρ , and vice versa. For simplicity, pixels with smaller ρ (or bigger n) are named as complex pixels and those with $\rho = \infty$ (or $n = 0$) are named as flat pixels. A flat pixel has an infinite embedding cost and must not be modified during data embedding.

In general, n is constructed from various independent components $n(d) \geq 0$, where $d \in \{1, \dots, K\}$, and K is the number of independent components. For example, in Li et al.'s work [3], $K = 2$ and LC are derived from both the horizontal and vertical directions. When $K > 1$, the way of constructing n from multiple $n(d)$ may impact the accuracy in measuring local complexity.

In this paper, the Hölder norm with $\rho < 0$ is exploited to derive n from $n(d)$ [19], as specified by

$$n = \left(\sum_{d=1}^K |n^{(d)}|^p \right)^{p^{-1}}, \quad (10)$$

where ρ can be simply set to constant -1. Now, embedding cost ρ can be derived using equations (9) and (10).

3.2. Construction of Multiple PEH. Since full-enclosed predictors have shown improved performance than half-enclosed predictors [3, 7], the double-layered image partition scheme and the rhombus prediction technique [7] are employed for prediction in this paper. In a double-layered image partition, a cover image is segmented into two layers, namely, the shadow layer and the blank layer, as shown in Figure 1(a). With a double-layered image partition, a pixel in one layer can be estimated by using its neighboring pixels in the other layer.

With a double-layered image partition, the rhombus predictor can be employed to estimate a pixel x from its four close neighbors v_1, v_2, v_3, v_4 (see Figure 1(b)), as specified by

$$\bar{x} = \left\lfloor \frac{v_1 + v_2 + v_3 + v_4}{4} \right\rfloor, \quad (11)$$

where $\lfloor \cdot \rfloor$ is the flooring operation. With \bar{x} , the PE can be derived as $e = x - \bar{x}$.

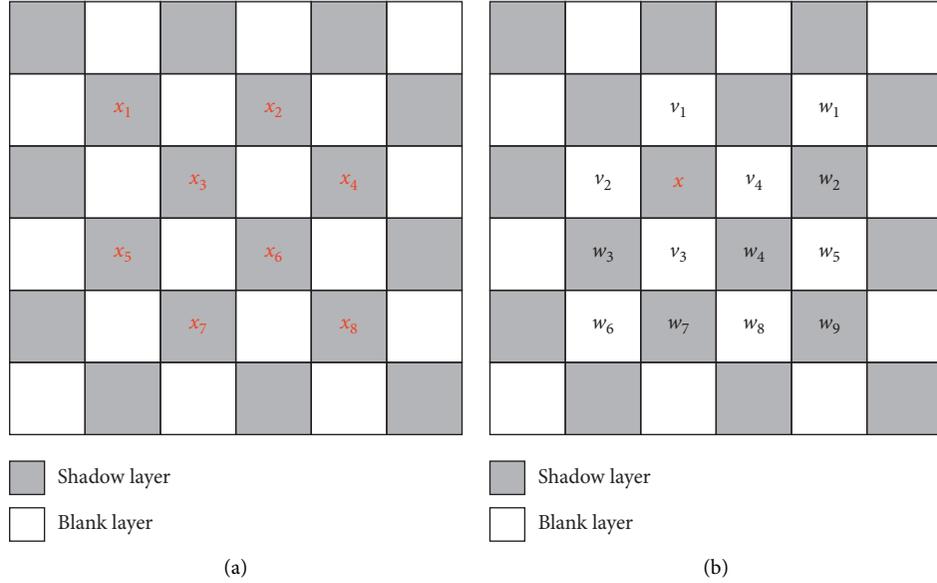


FIGURE 1: Double-layered image partition for layer-wise data embedding: (a) shadow and blank pixels' partition and (b) the local context for prediction and local complexity.

Accordingly, the horizontal directional information $n^{(1)}$ and the vertical directional information $n^{(2)}$ are obtained as the sum of absolute adjacent pixel difference values in the horizontal and vertical direction, respectively, as specified by

$$n^{(1)} = |v_4 - w_2| + |w_3 - v_3| + |v_3 - w_4| + |w_4 - w_5| + |w_6 - w_7| + |w_7 - w_8| + |w_8 - w_9| \quad (12)$$

and

$$n^{(2)} = |v_2 - w_3| + |w_3 - w_6| + |v_3 - w_7| + |v_4 - w_4| + |w_4 - w_8| + |w_1 - w_2| + |w_2 - w_5| + |w_5 - w_9|, \quad (13)$$

where v_2, v_3, v_4 and w_1, w_2, \dots, w_9 are the surrounding pixels of the to-be-predicted pixel x (Figure 1(b)). The directional information $n^{(1)}$ and $n^{(2)}$ represent the local complexities in the horizontal and vertical directions, respectively. To obtain LC, $n^{(1)}$ and $n^{(2)}$ are combined using equation (10), as specified by

$$n = \left(\frac{1}{n^{(1)}} + \frac{1}{n^{(2)}} \right)^{-1}. \quad (14)$$

With n , the embedding cost ρ is then derived using equation (9).

Now, multiple PEHs $h_n(e)$ can be constructed using equation (6). Let a_n and b_n ($a_n < b_n$) be the selected bins from PEH $h_n(e)$. Data bit m is embedded by modifying $h_n(e)$, as specified by

$$e' = \begin{cases} e - 1, & e < a_n, \\ e - m, & e = a_n, \\ e, & a_n < e < b_n, \\ e + m, & e = b_n, \\ e + 1, & e > b_n. \end{cases} \quad (15)$$

3.3. Efficient Parameter Optimization. Let Q be the corresponding embedding cost of the cover pixel sequence and X and Y be the marked pixel sequence. $X^{(n)}$ denote the subset of X with local complexity n , $Y^{(n)}$ denote the marked version of $X^{(n)}$, and $Q^{(n)}$ is the corresponding embedding cost for $X^{(n)}$. The overall embedding cost, denoted by C , can be calculated using the following equation:

$$\begin{aligned} C &= E(\|Y - X\|_2^2 \cdot Q) \\ &= \sum_n E(\|Y^{(n)} - X^{(n)}\|_2^2 \cdot Q^{(n)}) \\ &= \sum_{i=0}^N (y_i - x_i)^2 \rho_i, \end{aligned} \quad (16)$$

where y_i is the marked pixel and ρ_i is the embedding cost of x_i . Note that, compared to the image distortion D in equation (7), the overall embedding cost C in equation (16) weights a pixel modification by ρ , and thus, C reflects the overall embedding cost on local image statistics due to data embedding operations.

To increase the embedding security in RDH, data bits should be hidden into complex regions, where the embedding cost is smaller. Given a data bit sequence S , the problem of performance optimization regarding C can be written as $\min\{C\}$ subject to $\#\{S\} \leq \sum_n (a_n + b_n)$. This problem can be resolved numerically by selecting the optimal set of bin-pairs (a_n^*, b_n^*) , where $n \in [1, +\infty)$, but maybe with unacceptable computation cost. Therefore, three simplification strategies are employed to make the optimization process computationally efficient. Even though the three strategies are explained sequentially, they are logically independent and used separately. In this paper, all of the three simplification techniques are employed.

3.3.1. Reducing the Value Range of LC. LC usually varies from zero to a big value, which makes the solution of the overall embedding cost pretty complicated (equation (16)). Therefore, it is necessary to reduce the range of n to a smaller range $[0, M - 1]$ to simplify the problem of $\min\{C\}$. In other words, LC is split into M segments with $M - 1$ thresholds (denoted by L_t , where $t \in [1, M - 1]$), as specified by

$$L_t = \arg \min_n \left\{ \frac{\#\{i \in [1, N]: n_i \leq n\}}{N} \geq \frac{t}{M} \right\}. \quad (17)$$

As a result, the complexity measurement is mapped to M segments including $V_0 = [0, L_1), V_1 = [L_1, L_2), \dots, V_{M-1} = [L_{M-1}, +\infty)$. Thereby, the value of the new complexity measurement is set to n if the original LC falls in $V_n, 0 \leq n \leq M - 1$. For simplicity, n is reused to denote the new LC measurement, and thus, $n \in [0, M - 1]$.

3.3.2. Simplification on Calculating Embedding Cost. The calculation of C in equation (16) involves vector multiplexing operations and is time-consuming. By replacing $Q^{(n)}$ with the averaged embedding cost, denoted by $\rho^{(n)}$, the calculation can be made much efficient and C is approximated as

$$C \approx \sum_{n \geq 1} \rho^{(n)} E \left(\|Y^{(n)} - X^{(n)}\|_2^2 \right). \quad (18)$$

Since the data embedding process ends when S is embedded completely, only the former part X is visited. Let $N_{\text{end}}^{(n>0)} \leq N^{(n>0)}$ denote the number of visited pixels with $n > 0$. The embedding cost of modifying $X^{(n)}$ for a bin pair of (a_n, b_n) is approximated as

$$C^{(n)} \approx \frac{N_{\text{end}}^{(n>0)}}{N^{(n>0)}} \rho^{(n)} D_n(a_n, b_n), \quad (19)$$

where $D_n(a_n, b_n)$ represents the number of modified pixels in $X^{(n)}$ (equation (8)).

Let $\text{PS} = \#\{S\}$ be the size of the payload; PS is approximated as

$$\text{PS} \approx \frac{N_{\text{end}}^{(n>0)}}{N^{(n>0)}} \sum_{n \geq 1} (h_n(a_n) + h_n(b_n)). \quad (20)$$

By far, C can be approximated by combining equations (18)–(20).

3.3.3. Simplified Process of Bin-Pair Selection. The value of PE varies within a large range, e.g., for an eight-bit grey-scale image, $e \in [-255, 255]$. Since the PEH generally follows Gaussian distribution with two-sided fast decay [3, 7], the range of a_n and b_n can be restricted as follows:

- (i) For each $n \in [0, M - 1]$, $a_n = -b_n - 1$
- (ii) For each $n \in [0, M - 1]$, $b_n \in \{0, 1, 2, 3, 4, 5, 6, 7, +\infty\}$
- (iii) $b_0 \geq b_1 \geq \dots \geq b_{M-1} \geq 0$

Note that, these restrictions are made in a similar way as that of Li et al.'s work [3], the exception is that PEH $h_0(e)$ is not employed for data embedding. Since PEH generally follows a Gaussian-like distribution with fast two-sided decay, this simplification is technically reasonable for application.

Figure 2 illustrates an example of data embedding with this simplification. PEH $h_0(e)$ is not modified, and the selection of $b_0 \geq b_1 \geq \dots \geq b_{M-1} \geq 0$ forces data embedding into pixels with high LC. As a result, more bits can be embedded into the complex regions, and thus, the embedding cost becomes smaller.

3.4. Implementation Details. In this section, the implementation of the proposed method is presented. Figure 3 illustrates the procedures of implementation, where I_c, I , and I' denote the cover image, the preprocessed image, and the marked counterpart, respectively. In the embedding procedure, I_c is processed to avoid overflow and underflow. The processed pixels are registered in a location map LM, which is firstly compressed using lossless compression JBIG2 [27] and then embedded along with secret bits. The preprocessed image is partitioned into a shadow layer and a blank layer (Figure 1), and data bits are embedded into each layer individually.

3.4.1. Data Embedding Procedure. After the image pre-processing, data embedding is performed in a layer-wise manner. The processes of data embedding in the shadow layer and the blank layer are similar as follows:

- Step 1: obtain the cover pixel sequence X by scanning pixels in a top to bottom and left to right order
- Step 2: pixel estimation using equation (11) and then obtain PE using equation (1)
- Step 3: calculate local complexity n using equations (12)–(14) and the embedding cost ρ using equation (9)
- Step 4: reduce the value range of LC by projecting n to M levels ranging from 0 to $M - 1$ (see Section 3.3)
- Step 5: simplify the calculating of embedding cost by dividing Q into M subsets $Q^{(n)}$, and then, calculate the average embedding cost $\rho^{(n)}$ for each $Q^{(n)}$
- Step 6: construct M PEHs $h_n(e)$ using equation (6)
- Step 7: select M pairs of PEH bins (a_n^*, b_n^*) by minimizing C (equation (16)) with the simplified process of bin-pair selection (see Section 3.3)
- Step 8: perform data embedding via PEE (equation (15)) with the selected bin-pairs (a_n^*, b_n^*) and record the index of the last visited element $N_{\text{end}} \leq N$

3.4.2. Side Information. Side information is those parameters required by data extraction and image recovery. It is usually embedded into the preprocessed image I together with the data bit sequence S . The required side information includes

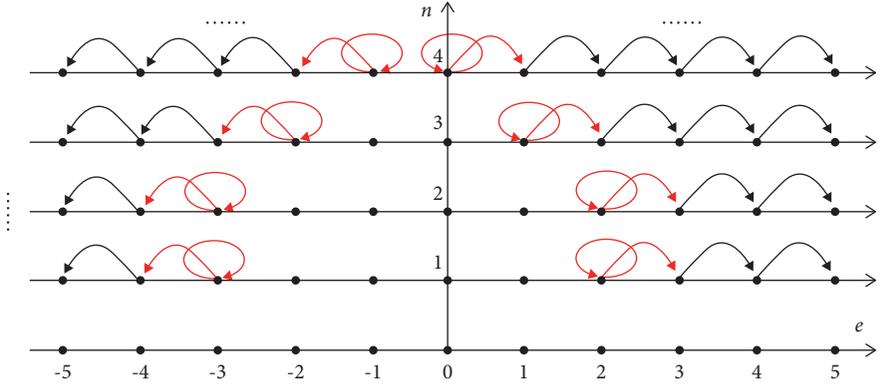


FIGURE 2: Example of the proposed MHM-based RDH embedding with selected bins of $b_1 = 2, b_2 = 2, b_3 = 1$, and $b_n = 0$, for $n \geq 4$.

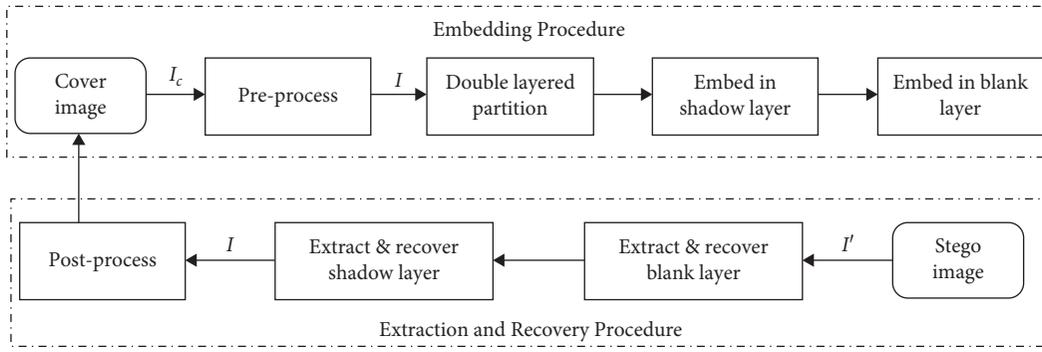


FIGURE 3: The implementation procedures of the proposed RDH method.

- (i) Segmentation threshold L_t , where $t \in \{1, \dots, M - 1\}$ ($10(M - 1)$ bits)
- (ii) The optimized bins b_n^* , where $n \in [0, M - 1]$ ($4M$ bits)
- (iii) The length of the pure payload ($\lceil \log_2 PS \rceil$ bits)
- (iv) The location of the last modified pixel N_{end} ($\lceil \log_2 N \rceil$ bits)
- (v) The compressed LM (N_{CLM} bits)
- (vi) The length of compressed LM ($\lceil \log_2 N_{\text{CLM}} \rceil$ bits),- where $\lceil \cdot \rceil$ is the ceiling operation, $M = 16$, and $N = 2^{17}$ denotes the number of cover pixels in each layer

Side information of the first four categories is generated individually during data embedding for each layer. The length is $14M + \lceil \log_2 PS \rceil + \lceil \log_2 N \rceil - 10 \leq 231 + \lceil \log_2 PS \rceil \leq 248$ bits. Since side information is required before data extraction and image recovery, it is stored in a preserved region of the cover image. The last two kinds of side information, including the compressed LM and its length ($N_{\text{CLM}} + \lceil \log_2 N_{\text{CLM}} \rceil \leq 18 + N_{\text{CLM}}$ bits), are shared by the whole data embedding procedure. The compressed location map and the length are only required in recovering the preprocessed images I to cover the image I_c ; therefore, they are embedded together along with the payload.

3.4.3. Data Extraction and Image Recovery. The reverse procedure is oppositely performed layer by layer compared to the embedding procedure. For each layer, the pixel sequence is scanned from bottom to top and right to left. Then, data extraction and image recovery procedures are performed with side information from the blank layer to the shadow layer. The compressed location map and its length can be obtained from the extracted bit sequence and used to recover the preprocessed image. By far, the cover image I_c is completely restored.

4. Experimental Analysis

Performances are evaluated from four perspectives including embedding security, embedding capacity, image quality, and time complexity. Since we aim at improving embedding security while maintaining embedding capacity and image quality, the evaluation of embedding security is put to high priority.

State-of-the-art RDH methods are employed for comparison, including CNN-based RDH [11], two-dimensional PEE-based method [8], pixel value ordering (PVO) based PEE method [9], RDH using Delaunay triangulation [21], and RDH method using MHM [3]. For simplicity, they are denoted by CNN, 2D-PEE, PVO, Delaunay, and MHM,

respectively. In Delaunay [21], the normalized startup position is set to 1, such that data embedding is restricted to pixels with high LC.

4.1. Embedding Security. The embedding security is evaluated by using two steganalyses including histogram analysis [28] and SPAM [22]. Histogram analysis is used to evaluate the abnormality of the difference histogram, and the SPAM steganalysis is employed to detect the changes in statistical image features.

4.1.1. Histogram Analysis. Histogram analysis is used as the first steganalysis to evaluate the difference of the histogram on eight typical grey-scale images of size 512×512 obtained from image database USC-SIPI [29], including Lena, Baboon, Peppers, Lake, Elaine, Man, House, and Tiffany. Figures 4(a) and 4(b) illustrate the different images of Lena when payload size is 10 and 30 kbits, respectively. The white dots represent data embedding locations. Embedding locations are found in complex regions instead of smooth regions. Therefore, the proposed method tends to preserve the smooth image regions from modification, and thus, the characteristics of smooth regions are better preserved.

Figure 5 presents the difference histograms for Lena, Baboon, Peppers, and Lake, respectively. In Figure 5, the scope of pixel differences is set to $[-10, 10]$ to better illustrate the deviation. Compared to the cover distributions, the difference histogram of the proposed method presents the least deviation. Specifically, the deviation is slightly better than that of 2D-PEE and much better than that of CNN, PVO, MHM, and Delaunay. Therefore, the proposed method better preserves the difference histogram, and thus, its robustness in resisting histogram analysis is proved. The difference histograms of the other typical images are not shown since they present similar results.

4.1.2. SPAM Analysis. The SPAM feature [22], which is a 686-dimensional second-order feature set, is employed to measure image statistical distortion. The ensemble classifier [30], for which Fisher linear discriminant is used as a base learner, is employed to classify cover images from their counterparts. The training-to-testing ratio of the ensemble classifier is set to 0.5, which means that half of the cover images and their marked counterparts are used for training and the remaining for testing. The performance is measured by the ensemble's "out-of-bag" (OOB) error [30]. The value of OOB ranges from 0 to 0.5. When OOB equals 0.5, the classifier cannot tell a cover image from its marked counterpart. The smaller the OOB, the less secure the data embedding.

The SPAM steganalysis is performed on 2000 images that are randomly selected from the widely used BOWS-2 [31] image database, where the cover signals are uncompressed eight-bit natural grey-scale images of size 512×512 . Since the payload of the PEH-based method varies from image to

image, there may exist cases when the maximum embedding capacity of a cover image is not enough for a given payload. For simplicity, a cover image, which is "capable" of carrying the given payload PS, is named as a "capable" image concerning PS. Figure 6(a) illustrates the number of "capable" images as the increment of embedding capacity. The number of "capable" images drops as the increment of embedding capacity. Therefore, the SPAM tests are performed using the same set of "capable" images to make reasonable security evaluations.

Figure 6(b) presents the curves of payload vs. OOB for which the corresponding payload is smaller than those of the embedding capacity among the 2000 BOWS-2 images. The detailed OOB data are listed in Table 1, where the best OOB is marked in bold black. OOB decreases as the increment of embedding capacity from 5 to 40 kbits. Compared to the other five methods, the proposed RDH presents the best OOB results when embedding capacity is larger than 5 kbits. For example, for embedding capacity of 20 kbits, the OOB value of the proposed method is 0.2232 which overpasses those of CNN, PVO, 2D-PEE, Delaunay, and MHM (0.2116, 0.2232, 0.1973, 0.1929, and 0.0946), respectively. When embedding capacity is 5 kbits, Delaunay produces a better OOB result, but when embedding capacity becomes larger, the OOB of Delaunay drops significantly. The reason is that Delaunay embeds data bits into the most complex pixels, and thus, less embedding cost is introduced at a low embedding capacity. In general, the proposed method is more secure in resisting SPAM attacks than other methods.

4.2. Embedding Capacity. The proposed method tends to embed data bits into high complexity pixels. Since the number of complex pixels is much less than that of low complexity pixels, the embedding capacity of the proposed method should be lower than that of Li et al.'s work [3]. As the increment of n , $h_n(e)$ becomes flattened and the peak becomes smaller, when considering $b_n \geq b_{n+1}$ (Figure 2), the proposed RDH embeds data into complex regions, and thus, it produces a lower embedding capacity than that of MHM.

As shown in Figure 6(a), the number of "capable" images of the proposed method is bigger than that of PVO but smaller than that of CNN, 2D-PEE, Delaunay, and MHM. As a result, the proposed scheme produces a larger embedding capacity compared to that of PVO, but a smaller embedding capacity than that of CNN, 2D-PEE, Delaunay, and MHM. The reason is that $h_n(e)$ is preserved from data embedding in the proposed method. Since the proposed work aims at improving embedding security, the evaluation of embedding capacity is not further discussed.

4.3. Image Quality. Research studies in image quality evaluation have shown that image fidelity, which is measured by the peak signal-to-noise ratio (PSNR), may not be the best image quality measurement [32]. Therefore, the structural similarity index (SSIM) [32] is employed to evaluate the image quality in addition to

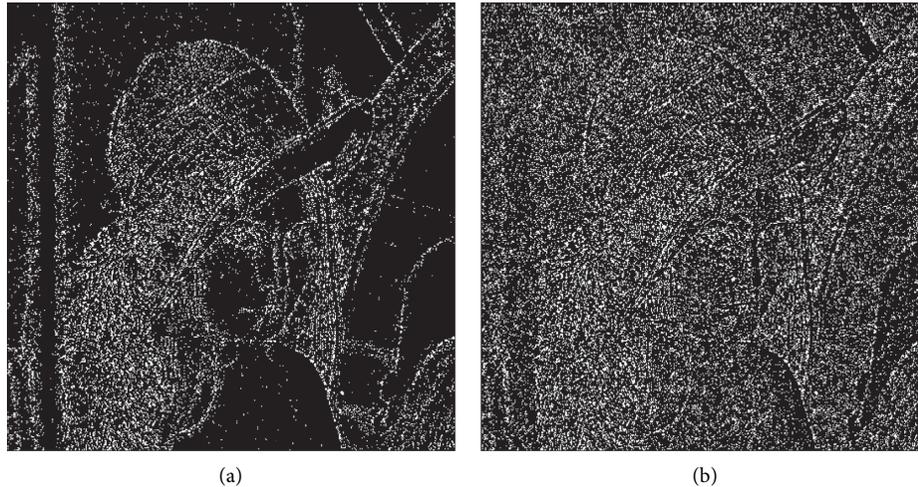


FIGURE 4: Illustration of the difference images of Lena with payload sizes of 10 kbits (a) and 30 kbits (b) for the proposed method. (a) PS: 10 kbits. (b) PS: 30 kbits.

PSNR. Compared to the distortion measurement PSNR, SSIM is a modern perception-based image quality indicator to reflect the change of structure information and the perceptual phenomena. The value of SSIM ranges from 0 to 1, and the bigger SSIM is, the better is the image structure similarity. When two images are identical, SSIM equals 1.

Figures 7(a) and 7(b) present averaged performances of payload against SSIM and averaged payload against PSNR, respectively. Since the maximum embedding capacity differs from image to image, the performance is averaged for only part of the 2000 BOWS-2 images, where the corresponding payload is smaller than the smallest embedding capacity. In Figure 7(a), the performance is averaged for the “capable” images (Figure 6(a)) only. The proposed method presents improved SSIM results than that of CNN, PVO, 2D-PEE, and Delaunay, but slightly reduced SSIM results when compared to that of MHM. In Figure 7(b), the proposed method offers improved PSNR results than CNN, PVO, and Delaunay, but reduced PSNR performance than that of 2D-PEE and MHM. As the proposed method is designed to improve the embedding security, such a result is significant.

4.4. Time Complexity. We evaluate the time complexity of the relevant algorithms by executing them on the same computer, which is configured with an Intel(R) Xeon(R) Gold 5120 CPU running at 2.20 GHz and 32 GB DRAM memory. The time complexity is obtained by varying the size of the payload from 10 to 60 kbits. The averaged time consumption for the 2000 BOWS-2 images is used for comparison, as shown in Table 2.

CNN takes more than 15 seconds on average and is the most time-consuming. 2D-PEE and PVO cost less than one second on average and are the least time-consuming. Delaunay, MHM, and the proposed method require two to three seconds on average. Furthermore, for CNN, PVO, and

Delaunay, the averaged time complexity increases as the increment of the size of the payload, while for 2D-PEE, MHM, and the proposed method, the average consumed time does not vary much. Compared with that of the previous MHM, the proposed work requires longer time since more calculation is required in the procedure of embedding cost minimization.

4.5. Discussion. The experiment results in the previous sections present the embedding performance of the proposed work from four aspects including embedding security (histogram analysis and SPAM analysis), embedding capacity, image quality (SSIM and PSNR), and time complexity. Compared to the previous studies, the proposed method is more secure in defending cyberattacks by using histogram analysis and the modern steganalysis ensemble with SPAM features (Figure 6(b)). In addition to the improved security, the proposed work presents well-preserved structural image similarity (Figure 7(a)), considerable embedding capacity (Figure 6(a)), well-preserved image fidelity (Figure 7(b)), and low time complexity (Table 1).

There are two reasons for the improved embedding security and well-preserved structural similarity. The first reason is that data embedding is restricted into complex image regions that are hard to model when not using pixel difference histogram or SPAM features. The second reason is that embedding cost is minimized by using the framework of MHM. When compared to the original work of MHM, the proposed method presents lower embedding capacity and image fidelity. The reason is that the proposed work selects larger PEH bins for smaller LC and smaller PEH bins for larger LC. Thus, less embedding capacity is obtained and more distortion is introduced. Compared to the framework of MHM, the proposed work is more time-consuming, as more time is required in the optimized overall embedding cost.

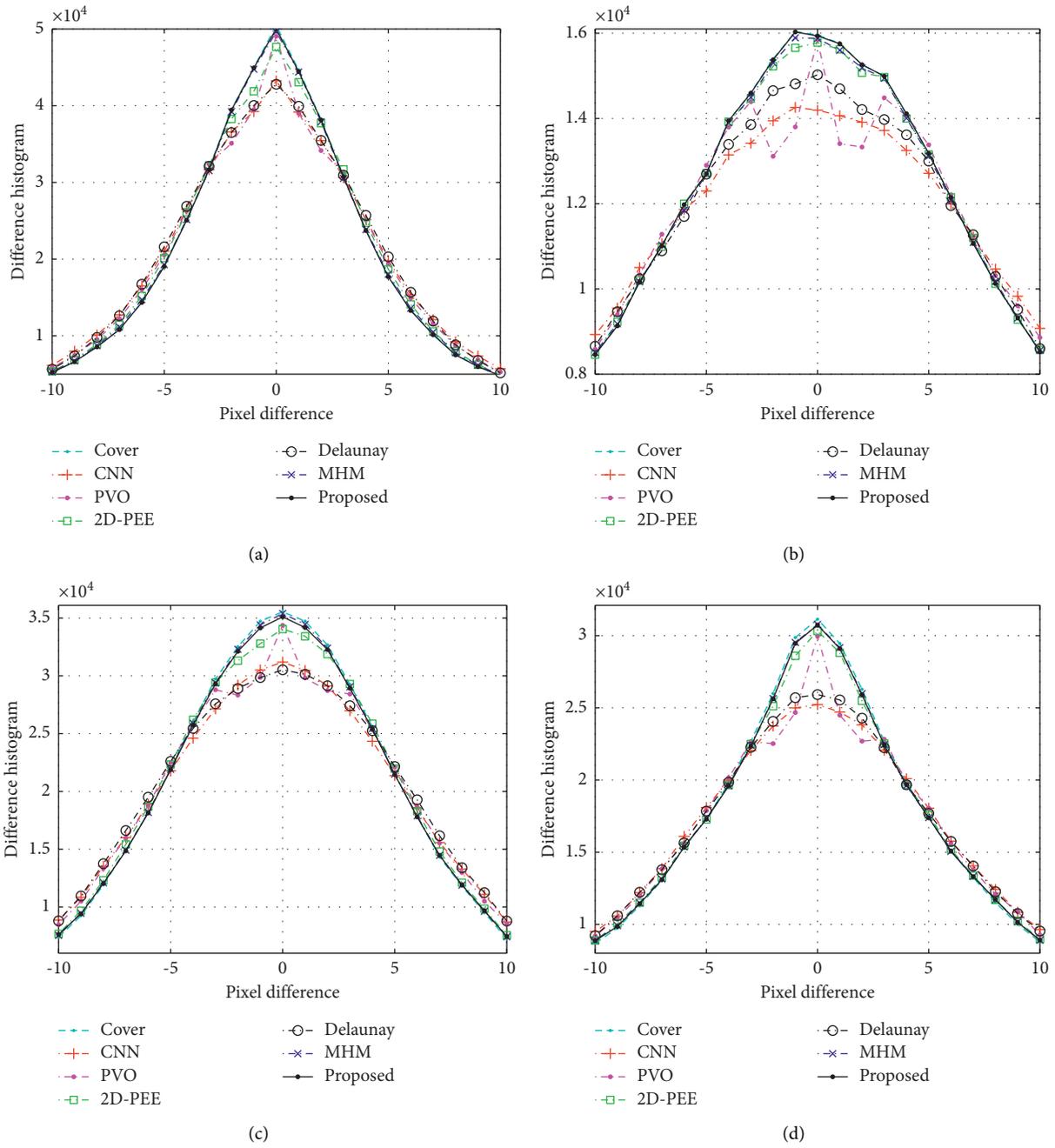


FIGURE 5: Histogram analysis by comparing the difference histograms. (a) Lena, PS: 20 kbits. (b) Baboon, PS: 10 kbits. (c) Peppers, PS: 20 kbits. (d) Lake, PS: 20 kbits.

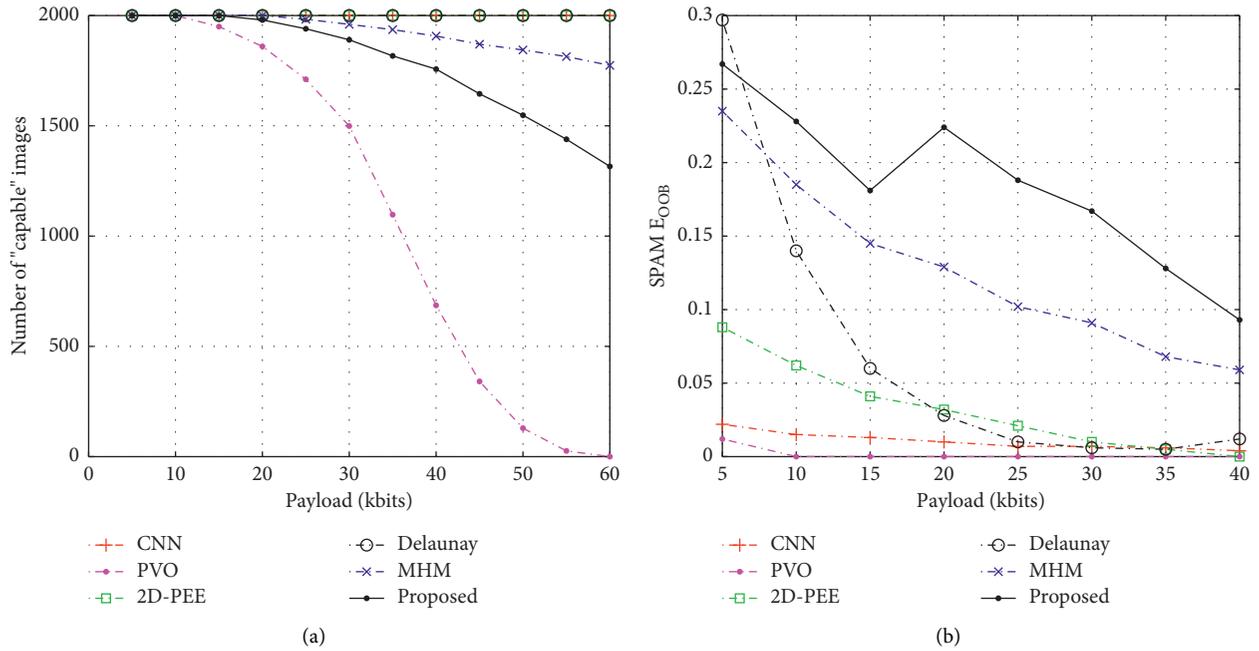


FIGURE 6: Comparison of the performance of embedding security against SPAM attacks. (a) Number of "capable" images. (b) Performance of OOB vs. payload.

TABLE 1: Performance of embedding security (OOB) against SPAM attacks.

Methods	Performance of embedding security (OOB)						
	5	10	15	20	25	30	40
CNN	0.022	0.015	0.013	0.010	0.007	0.007	0.004
PVO	0.012	0.000	0.000	0.000	0.000	0.000	0.000
2D-PEE	0.088	0.062	0.041	0.032	0.021	0.010	0.000
Delaunay	0.297	0.140	0.060	0.028	0.010	0.006	0.012
MHM	0.235	0.185	0.145	0.129	0.102	0.091	0.059
Proposed	0.267	0.228	0.181	0.224	0.188	0.167	0.093

The best security performance is marked in bold for each size of payload.

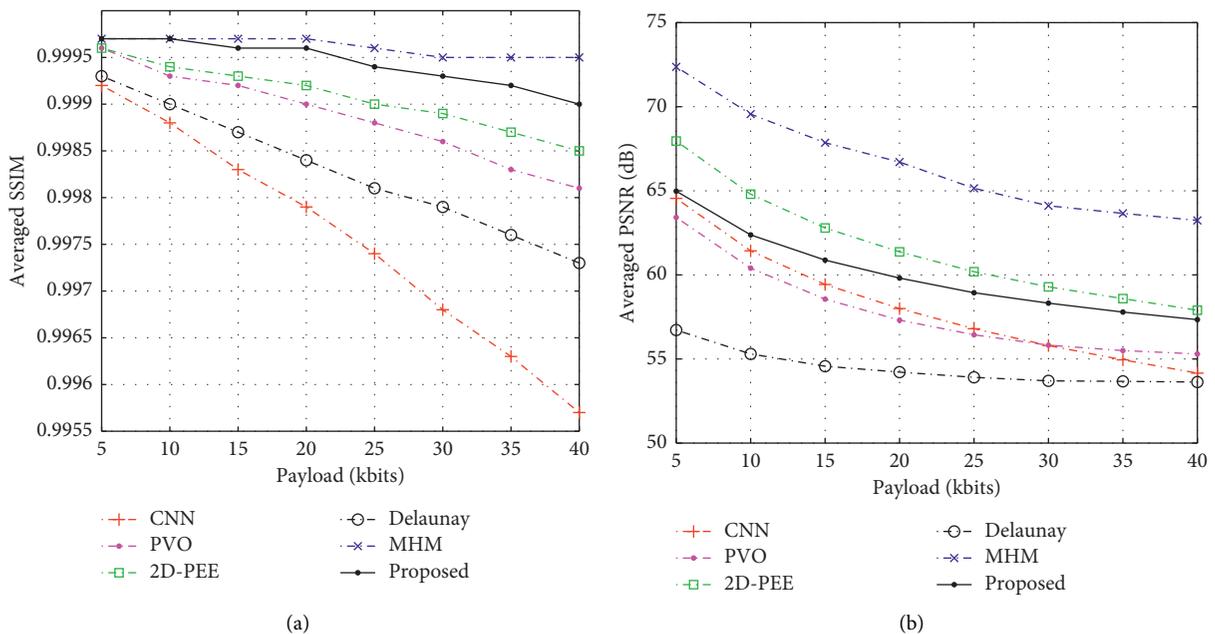


FIGURE 7: Comparison of image quality using averaged SSIM and PSNR. (a) Performance of SSIM vs. payload. (b) Performance of PSNR vs. payload.

TABLE 2: Averaged time consumption (in second).

Methods	Size of pure payload (kbits)					
	10	20	30	40	50	60
CNN	15.1	28.0	24.4	24.9	24.8	33.5
PVO	0.5	0.7	0.8	0.8	0.9	0.9
2D-PEE	0.4	0.4	0.4	0.5	0.5	0.5
Delaunay	1.9	2.0	2.1	2.4	2.9	3.1
MHM	2.0	2.0	2.0	2.0	2.0	2.0
Proposed	3.3	3.2	3.1	3.0	2.9	2.9

5. Conclusion

A novel RDH method is proposed based on multiple histograms modification to improve the embedding security. The MHM framework is modified to restrict data hiding into high complexity regions, and the harmonic mean of directional pixel differences is employed as the complexity measurement. As different from previous RDH developments, this study highlights embedding security in addition to image quality. Experimental results demonstrate that the proposed method yields improved embedding security, well-preserved structural image similarity, considerable embedding capacity, and image fidelity, compared to other methods. The proposed method provides the basis for the future development of secure and reversible data hiding methods.

Data Availability

There are two types of data in this manuscript, the source code and employed cover images. The source code in this study is available from the corresponding author upon request. The employed cover images are from third parties and can be accessed publicly from Internet as specified in [29, 31].

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the Guangzhou Science and Technology Plan Project of China, under Grant 201904010276, Guangdong Innovative Projects with Characteristics in Colleges and Universities, under Grant 2019KTSCX228, scientific research projects of Guangzhou Panyu Polytechnic, under Grants 2021KJ04 and 2021KJ13, Scientific research platform and projects in Guangdong Universities of China, under Grant 2019GKTSCX069, and Key Platforms and Scientific Research Projects in Universities in Guangdong Province under Grant 2020KCXTD053.

References

- [1] Y. Q. Shi, X. Li, X. Zhang, H.-T. Wu, and B. Ma, "Reversible data hiding: advances in the past two decades," *IEEE Access*, vol. 4, pp. 3210–3237, 2016.
- [2] D. Li, Y. Zhang, X. Li, K. Niu, X. Yang, and Y. Sun, "Two-dimensional histogram modification based reversible data hiding using motion vector for H.264," *Multimedia Tools and Applications*, vol. 78, no. 7, pp. 8167–8181, 2019.
- [3] X. Li, W. Zhang, X. Gui, and B. Yang, "Efficient reversible data hiding based on multiple histograms modification," *IEEE Transactions on Information Forensics and Security*, vol. 10, no. 9, pp. 2016–2027, 2015.
- [4] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890–896, 2003.
- [5] D. M. Thodi and J. J. Rodriguez, "Expansion embedding techniques for reversible watermarking," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 721–730, 2007.
- [6] X. Wang, C. Chang, C. Lin, and C. Chang, "Privacy-preserving reversible data hiding based on quad-tree block encoding and integer wavelet transform," *Journal of Visual Communication and Image Representation*, vol. 79, no. 2, Article ID 103203, 2021.
- [7] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y. Q. Shi, "Reversible watermarking algorithm using sorting and prediction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 7, pp. 989–999, 2009.
- [8] X. Li, W. Zhang, X. Gui, and B. Yang, "A novel reversible data hiding scheme based on two-dimensional difference-histogram modification," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 7, pp. 1091–1100, 2013.
- [9] X. Li, J. Li, B. Li, and B. Yang, "High-fidelity reversible data hiding scheme based on pixel-value-ordering and prediction-error expansion," *Signal Processing*, vol. 93, no. 1, pp. 198–205, 2013.
- [10] M. Xiao, X. Li, Y. Wang, Y. Zhao, and R. Ni, "Reversible data hiding based on pairwise embedding and optimal expansion path," *Signal Processing*, vol. 158, pp. 210–218, 2019.
- [11] R. Hu and S. Xiang, "CNN prediction based reversible data hiding," *IEEE Signal Processing Letters*, vol. 28, pp. 464–468, 2021.
- [12] J. Hou, B. Ou, H. Tian, and Z. Qin, "Reversible data hiding based on multiple histograms modification and deep neural networks," *Signal Processing: Image Communication*, vol. 92, Article ID 116118, 2021.
- [13] W. Wang, C. Wang, J. Wang, S. Bian, and Q. Huang, "Improving multi-histogram-based reversible watermarking using optimized features and adaptive clustering number," *IEEE Access*, vol. 8, Article ID 134334, 2020.
- [14] B. Ou, X. Li, and J. Wang, "Improved PVO-based reversible data hiding: a new implementation based on multiple histograms modification," *Journal of Visual Communication and Image Representation*, vol. 38, pp. 328–339, 2016.
- [15] J. Qin and F. Huang, "Reversible data hiding based on multiple two-dimensional histograms modification," *IEEE Signal Processing Letters*, vol. 26, no. 6, pp. 843–847, 2019.
- [16] J. Wang, N. Mao, X. Chen, J. Ni, C. Wang, and Y. Shi, "Multiple histograms based reversible data hiding by using FCM clustering," *Signal Processing*, vol. 159, pp. 193–203, 2019.
- [17] S. Weng, W. Tan, B. Ou, and J. S. Pan, "Reversible data hiding method for multi-histogram point selection based on improved crisscross optimization algorithm," *Information Sciences*, vol. 549, pp. 13–33, 2021.
- [18] B. Li, S. Tan, M. Wang, and J. Huang, "Investigation on cost assignment in spatial image steganography," *IEEE Transactions on Information Forensics and Security*, vol. 9, no. 8, pp. 1264–1277, 2014.

- [19] V. Holub and J. J. Fridrich, "Designing steganographic distortion using directional filters," in *Proceedings of the IEEE Workshop Inf. Forensics & Secur*, Costa Adeje, Spain, December 2012.
- [20] X. Liu, J. Liang, Z. Wang, Y. Tsai, C. Lin, and C. Chen, "Content-based image copy detection using convolutional neural network," *Electronics*, vol. 9, no. 12, 2020.
- [21] W. Hong, T. S. Chen, and J. Chen, "Reversible data hiding using Delaunay triangulation and selective embedment," *Information Sciences*, vol. 308, pp. 140–154, 2015.
- [22] T. Pevny, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 2, pp. 215–224, 2010.
- [23] C. Wang, X. Li, and B. Yang, "Efficient reversible image watermarking by using dynamical prediction-error expansion," in *Proceedings of the IEEE International Conference on Image Processing*, Hong Kong, China, September, 2010.
- [24] L. Kamstra and H. J. A. M. Heijmans, "Reversible data embedding into images using wavelet techniques and sorting," *IEEE Transactions on Image Processing*, vol. 14, no. 12, pp. 2082–2090, 2005.
- [25] W. He, Z. Cai, and Y. Wang, "High-fidelity reversible image watermarking based on effective prediction error-pairs modification," *IEEE Transactions on Multimedia*, vol. 23, pp. 52–63, 2021.
- [26] A. Mohammadi and M. Nakhkash, "Sorting methods and adaptive thresholding for histogram based reversible data hiding," *Multimedia Tools and Applications*, vol. 80, no. 3, pp. 3307–3325, 2021.
- [27] P. G. Howard, F. Kossentini, B. Martins, S. Forchhammer, and W. J. Rucklidge, "The emerging JBIG2 standard," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 8, no. 7, pp. 838–848, 1998.
- [28] X. Zhang and S. Wang, "Vulnerability of pixel-value differencing steganography to histogram analysis and modification for enhanced security," *Pattern Recognition Letters*, vol. 25, no. 3, pp. 331–339, 2004.
- [29] Usc-Sipi, "The USC-SIPI image database," 1977, <http://sipi.usc.edu/database>.
- [30] J. Kodovsky, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 2, pp. 432–444, 2012.
- [31] P. Bas and T. Furon, "BOWS-2," 2007, <http://bows2.ec-lille.fr/>.
- [32] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE Transactions on Image Processing*, vol. 13, no. 4, pp. 600–612, 2004.