

Research Article

Weighted Average-Based Complexity Calculation in Block Selection Oriented Reversible Data Hiding

Xin Tang , Yiteng Zhou , Yixuan Cheng , and Yan Shao 

School of Cyber Science and Engineering, University of International Relations, Beijing 100091, China

Correspondence should be addressed to Xin Tang; xtang@uir.edu.cn

Received 18 January 2022; Revised 20 May 2022; Accepted 7 June 2022; Published 26 September 2022

Academic Editor: Gökhan Kul

Copyright © 2022 Xin Tang 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.

Complexity of blocks is one of the key factors to influence the performance of reversible data hiding. By embedding data into blocks with smaller complexity, the invalid shifting in pixel value ordering (PVO) based reversible data hiding is largely reduced, thus increasing the imperceptibility. However, the existing techniques are not always accurate in complexity calculation, since context pixels are usually treated equally. As a result, a great number of invalid pixels are assigned too large a weight in complexity calculation, which inevitably affects the precision. In this paper, we propose a weighted average based novel strategy in complexity calculation. In this scheme, we calculate the weight of a certain context pixel with the help of its adjacent neighbors, which ensures the one in a smooth area is assigned a larger weight, and otherwise the opposite. Experimental results show that invalid shifting is largely reduced, thus our scheme is better in imperceptibility compared with the state of the art and has potential application in a lot of fields such as medical imaging.

1. Introduction

Reversible data hiding [1–4] is a novel technique to achieve convert communication, which enables the cover signal to be completely recovered to its original state once the embedded data are extracted. Because of its lossless nature, reversible data hiding has been widely employed in a lot of high fidelity required scenarios, such as medical images and military communications [5–9]. However, in order to achieve lossless data hiding, the maximum embedding capacity of a certain cover signal is significantly lower than that achieved by conventional data hiding algorithms, which usually leads to poor imperceptibility under the same amount of secret data embedded [10, 11]. As a result, how to achieve high fidelity reversible data hiding is becoming a big challenge.

Pixel value ordering (PVO) [12–15] is an emerging method to deal with this problem, which works by predicting both the maximum and minimum pixels with the help of the second largest and smallest ones before expanding the obtained appropriate prediction errors to achieve secret data embedding. Specifically, if the prediction

error is too large, the embedding condition is usually not satisfied. As a result, invalid shifting has to be performed, which in turn brings about a lot of distortions in imperceptibility. For texture blocks with complex features, the disadvantages are more obvious because of the inaccuracy in prediction. As one of the effective means to ensure the performance of reversible data hiding, block selection is widely employed [16–19]. A general principle is to select smooth blocks to embed data at first, thus ensuring the obtained prediction error at a low level. Nevertheless, the actual performance is seriously restrained by the smoothness of selected blocks, which is defined to be their complexity value. Even though a lot of works [20–24] focus on this problem, the actually obtained result is not always accurate in reflecting the real smoothness. The main reason is that those invalid context pixels are still assigned too large a weight to participate in the calculation, which inevitably introduces a lot of uncertainty.

In this paper, we propose a weighted average based complexity calculation method for block selection oriented reversible data hiding. To the best of our knowledge, the method proposed is the first work to well distinguish smooth

blocks from texture ones according to the defined complexity value. Specifically, for ordered pixels inside of a certain block, the valid context pixels are defined to be the second largest pixel itself and neighbors of the largest pixel with both the largest and smallest pixels eliminated, no matter which block they belong to. In order to eliminate the unnecessary error in predicting the largest pixel, those smaller than the second maximum of the current block are considered to be invalid ones and also eliminated. Then we assign a distinct weight to each one of the remaining pixels, according to their fluctuation in a small neighborhood. According to our design, context pixels in smooth areas are assigned relative larger weights, otherwise the opposite. Next the complexity value in the maximum side of the current block is defined to be the difference between the weighted average values of valid surrounding pixels and the second largest pixel itself. Experimental results demonstrate that the proposed scheme is better in imperceptibility than the state of the art, which is thus more suitable to high fidelity required fields such as medical imaging. Our main contributions are summarized as follows:

- (1) We put forward a novel block selection strategy-based framework to support the proposed complexity calculation method for blocks. With the help of the proposed framework, those blocks better in smoothness are selected to embed data at first, which largely reduces the invalid shifting and ensures the imperceptibility before full embedding.
- (2) We propose an improved method to calculate complexity value. Specifically, we assign a distinct weight to each one of the valid context pixels, according to their fluctuation in a small neighborhood, which largely improves the accuracy in complexity evaluation. Thus, blocks with large complexity could be skipped in data embedding, and high imperceptibility is ensured.

The structure of the rest of the paper is organized as follows. We introduce the related work in Section 2 and present our weighted average-based complexity calculation scheme in Section 3. Next, the experimental evaluation is performed in Section 4, followed by the security analysis in Section 5. Finally, we conclude this paper in Section 6.

2. Related Works

In order to evaluate the complexity of the blocks to support the block selection-oriented reversible data hiding, Li et al. [12] first proposed to utilize internal pixels to participate in calculation. According to their method, the complexity value of a certain block is defined to be absolute difference of the second maximum pixel and second minimum one inside, each of which is ensured to be consistent even though the secret data is embedded. By selecting blocks smaller in complexity to embed data at first, the invalid shifting is largely reduced thus the imperceptibility is ensured. However, the rather limited number of pixels participating in calculation largely restrains the performance to predict the prediction error. Moreover, according to their design, a

single fluctuation value is utilized to predict both the maximum and minimum sides simultaneously, which also inevitably introduces uncertainty. Based on their work, Ou et al. [14, 15] introduced outside pixels to participate in complexity calculation. Specifically, they select two columns of pixels on the right side of a block and two rows of pixels on the bottom as context ones to calculate complexity, which is able to achieve a better performance. Similar to their design, Weng et al. [20] employed a column and a row. In particular, the complexity of a block is defined to be the summation of adjacent pixels' absolute difference in both vertical and horizontal directions. However, take the maximum side for example, once the largest and second largest pixels are positioned at the top left corner of the block, their difference actually cannot be inferred by context pixels which are faraway. To deal with the problem, Xiang et al. [21] proposed a full-closed strategy to select context pixels. Specifically, nonoverlapped blocks are alternatively classified to two groups in a rhombus pattern. In each one of the groups, the complexity value of a block is determined by its surrounding blocks that belong to the other group. However, the internal pixels are actually neglected in the calculation. Moreover, a single complexity value is also employed to evaluate both sides, which is similar to Li et al.'s work [12].

As a follow-up work, Pan et al. [22] take internal pixels into account on the basis of Xiang's work. Take the maximum side for example, the valid neighbors of both the largest and second largest pixels are defined to be context pixels. Then the second largest and second smallest pixels are selected in calculation. In addition, Tang et al. [23] introduced an average strategy, which evaluates the difference of both the largest and smallest pixels by employing the difference between the average value of their valid neighbors. As a result, a single complexity value is not utilized for both sides anymore. Furthermore, Tang et al. [24] provided an improved scheme to further eliminate those invalid pixels. However, the average strategy is still utilized, which means each one of the neighbors is treated equally. For texture images, this approach is obviously inappropriate.

On the other hand, Jana et al. [25] proposed a dual image-based steganography scheme by combining the adjacent two pixels together and calculating their difference before expansion. As a result, more pixels are employed in data embedding, which means a larger potential capacity could be achieved. Similar to their work, Mukherjee et al. [26] further achieved pixels filtering by calculating their correlations. However, both methods are not good enough in visual quality. Based on their works, Jana et al. [27] employed (7, 4) Hamming code to obtain a better performance in PSNR. Specifically, they created a redundant position to embed data at first and recovered the original cover image by Hamming code-based error correction. In order to further improve the performance, Jana et al. [28, 29] proposed the first weighted average-based work. In particular, they defined a weighted matrix to perform multiple rounds of modular summation with original pixel blocks and achieved data embedding based on the results of calculation. As a follow-up work, Chowdhuri et al. [30] extended the design to DCT coefficient and achieved a better

imperceptibility. However, in these works, the weighted values are not utilized in complexity calculation.

3. The Proposed Scheme

In this section, we introduce a general framework to support the proposed method before presenting the whole process in detail. Specifically, in the part of scheme description, we begin with weight definition for context pixels, followed by the weighted average-based complexity value calculation. At last, according to the value obtained, we describe the procedures of data embedding and extracting, respectively.

3.1. Framework. As shown in Figure 1, the cover image is divided into nonoverlapping blocks of equal size at first, and then the complexity values for both sides are calculated according to the procedure introduced in Section 3.2. We compare each one with the predefined threshold value to determine whether a secret bit should be embedded in the corresponding position. Once the embedding process is completed, the receiver enables to extract the embedded data and restore the cover image according to the method introduced in Section 3.4.

For the i th ($1 \leq i \leq K$) block, we sort its n pixels to obtain an ascending sequence $(x_{\sigma(1)}^i, x_{\sigma(2)}^i, \dots, x_{\sigma(n)}^i)$, in which $\sigma: (1, 2, \dots, n) \rightarrow (1, 2, \dots, n)$ is a one-to-one mapping ensuring that: $x_{\sigma(1)}^i \leq x_{\sigma(2)}^i \leq \dots \leq x_{\sigma(n)}^i$. For any $u, v \in [1, n]$, if $x_{\sigma(u)}^i \leq x_{\sigma(v)}^i$ and $u < v$, we make $\sigma(u) < \sigma(v)$. Take the maximum side for example, we first eliminate those largest and smallest pixels from eight neighbors of $x_{\sigma(n)}^i$, no matter which block they belong to. In addition, pixels smaller than the second largest one are removed as well since they are actually invalid to predict the largest one. As a result, the remaining pixels are defined to be valid context ones of the maximum side. Then we assign a certain weight to each one of the context pixels, according to their fluctuation in a small neighborhood, and employ a weighted average-based strategy to calculate the complexity value in this case. If the result is lower than the predefined threshold value, secret data are able to be embedded in the corresponding position, and then the auxiliary information accordingly.

3.2. Calculation of Weight Value. Consider the same block mentioned in Section 3.1, suppose the context pixels for $x_{\sigma(n)}^i$ are obtained as shown in Figure 2(a). As is shown in the figure, the pixel marked by the black dot is supposed to be the largest or the smallest one of this or other blocks, and the pixel marked by the gray dot is the one whose value is lower than the second largest one. In order to ensure reversibility, both of them are removed. Then the remaining pixels $X_1^i, X_2^i, \dots, X_6^i$ are defined to be valid context pixels of $x_{\sigma(n)}^i$. Take X_j^i ($0 \leq j \leq 6$) for example, its adjacent pixels are defined to be a, b, c , and d , which is shown in Figure 2(b). If a, b, c , and d are valid neighbors, each one of them is neither the largest nor the smallest pixel of other blocks. Similar as in [31], the fluctuation value F_j^i of X_j^i is defined in (1).

$$F_j^i = |a - b| + |c - d| + |a + d - b - c| + |a + c - b - d|. \quad (1)$$

If any one of a, b, c , and d is an invalid one, the corresponding absolute value relating to this value in (1) is removed from the definition to ensure reversibility. On the other hand, if more than a half of a, b, c , and d are invalid, (1) is no longer applicable. In this case, add X_j^i to set B, otherwise to set A. Define p to be the number of elements in B and q the total number in both A and B.

It is clear that context pixels with smaller fluctuation value usually appears in the smooth regions of the image and thus should be assigned a larger weight. In contrast, those with larger fluctuation value should be assigned a smaller weight. This is because the values of pixels in textural regions can vary widely, which increases the uncertainty when predicting the maximum pixel value. In order to ensure the prediction accuracy, their weights in complexity computation should be reduced as a result.

As a means to achieve this feature, for a context pixel X_j^i ($1 \leq i \leq K, 1 \leq j \leq q$) in set A, we define the normalized initial weight w_j^i to be $(1/F_j^i + 1)/(\sum 1/F_j^i + 1)$. According to the definition, a larger F_j^i is ensured to associate to a smaller result and vice versa. On the other hand, if X_j^i is in Set B, the weight w_j^i is initiated to be 1. To ensure the sum of weights for context pixels of $x_{\sigma(n)}^i$ is 1, we further normalize them as

$$W_j^i = \begin{cases} w_j^i \cdot \frac{q-p}{q}, & \text{if } X_j^i \in A, \\ w_j^i \cdot \frac{1}{q}, & \text{if } X_j^i \in B. \end{cases} \quad (2)$$

In addition, if all pixels in a certain block are identical in value, the weights of context pixels can still be distinct according to (1) and (2). Since they could be determined by both inside and outside pixels simultaneously. Specifically, if the valid context pixel is an external one, their weight is mainly affected by outside pixels. On the other hand, even though a valid context pixel is an internal one, its valid neighbors may still contain outside pixels.

3.3. Complexity Value Calculation and Block Selection. For the i th block of the certain cover image, according to the weights defined in Section 3.2, the value of the largest pixel $x_{\sigma(n)}^i$ is estimated to be $P_{\sigma(n)}^i$ as follows, in which m denotes the number of valid context pixels:

$$P_{\sigma(n)}^i = \sum_{j=1}^m W_j^i \cdot X_j^i. \quad (3)$$

Correspondingly, the complexity value C_{\max}^i for the i th block is defined as

$$C_{\max}^i = \left| P_{\sigma(n)}^i - X_{\sigma(n-1)}^i \right|. \quad (4)$$

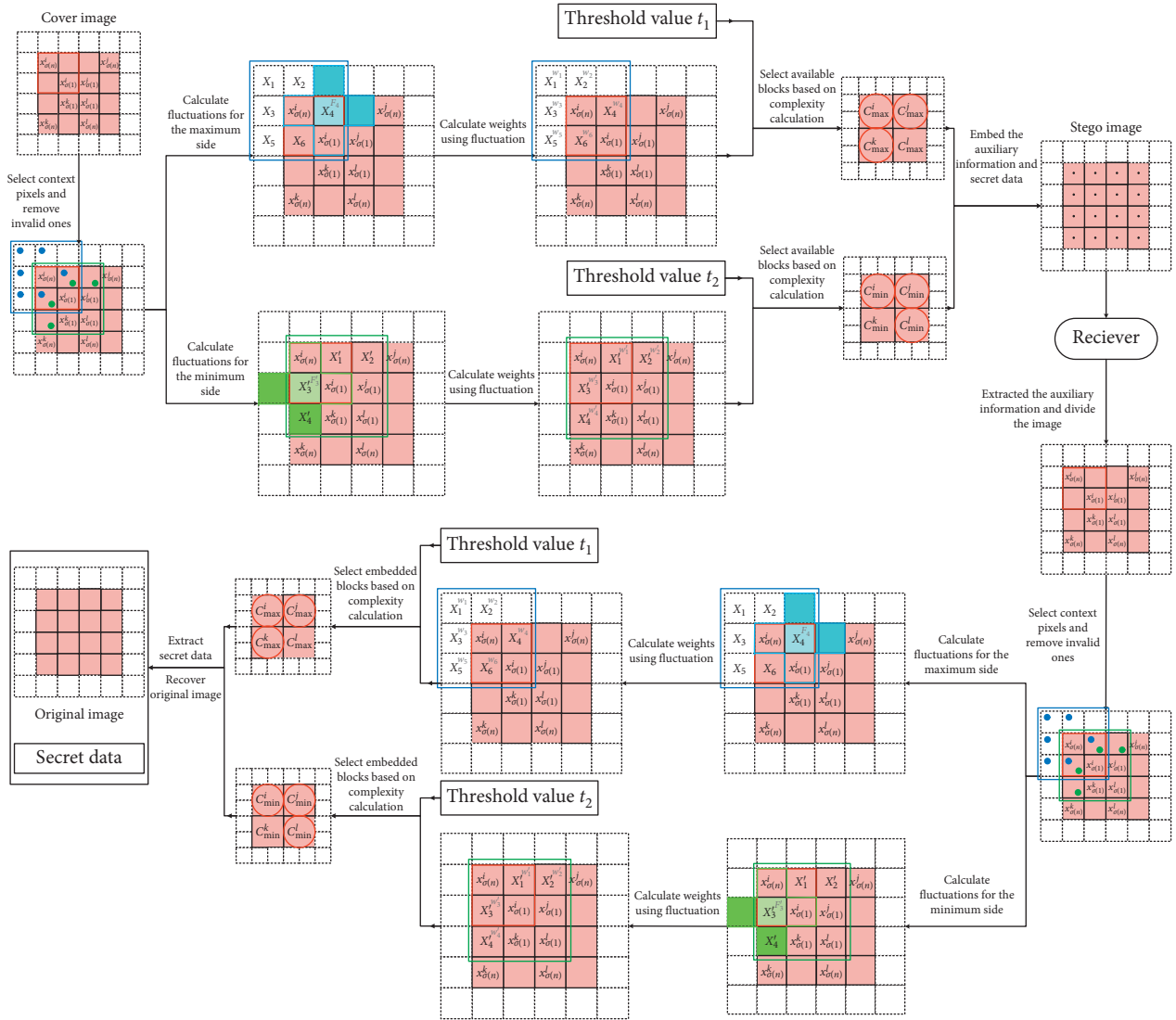


FIGURE 1: The framework of the proposed block-level complexity calculation-based reversible data hiding.

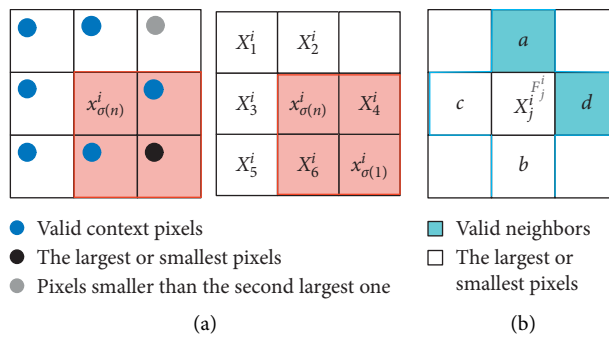


FIGURE 2: The context pixels of the maximum one in the block and fluctuation calculation about context pixels in the small neighborhood. (a) Context pixels of $x_{\sigma(n)}^i$. (b) Adjacent pixels of X_j^i .

Similarly, the definitions for the minimum side are presented in (5) and (6) accordingly, in which m' denotes the number of valid context pixels in this case:

$$P_{\sigma(1)}^i = \sum_{j=1}^{m'} W_j^i \cdot X_j^i, \quad (5)$$

$$C_{\min}^i = \left| P_{\sigma(1)}^i - X_{\sigma(2)}^i \right|. \quad (6)$$

According to the definitions above, the complexity of a block is equal to the difference between the predicted largest (smallest) pixel value and the second largest (smallest) one, which reflects its smoothness. The smaller the complexity value, the smoother the block. Actually, the smoothness of a certain block is defined to be the similarity degree of pixels inside. In PVO-based reversible data hiding, the difference between the value of the largest (smallest) pixel and that of the second largest (smallest) one is usually employed to evaluate the smoothness of a block. In smooth blocks, the difference is small enough to satisfy the embedding condition, thus most of the largest (smallest) pixels are used to embed data. On the other hand, textural ones are skipped to eliminate invalid shifting. Considering the variability in data embedding process, in the definition of complexity, the largest (smallest) pixel is replaced by its predicted value. Clearly, the more accurate the prediction, the more closer the evaluated difference to the true result. In order to well predict the largest (smallest) pixels, the weights of their context pixels should be considered, since if they are considered equally, a high level of uncertainty would be introduced inevitably. According to our design, a smaller weight is assigned to context pixels in textural regions, and a larger weight is to those in smooth regions. As a result, the complexity value obtained by weighted average strategy is able to better reflect the smoothness of the corresponding block.

In the process of block selection, take the maximum side for example, blocks of the cover image are scanned in the order from top to bottom, and left to right. Only those with a complexity value lower than the predefined threshold value t_1 are selected to embed data. Otherwise, the blocks are skipped. The same process is also applied to the minimum side as well with assistance of the threshold value t_2 . In order to support such a design, the i th ($1 \leq i \leq K$) block corresponds to two complexity values c_{\max}^i and c_{\min}^i , respectively. It is worth mentioning that in the choice of t_1 and t_2 , a trade-off between embedding capacity and imperceptibility is considered. On one hand, we desire a large threshold value to obtain more available blocks for embedding. On the other hand, a larger threshold value means that more textural blocks have to be selected, which inevitably introduces a lot of invalid shifting as a result.

3.4. Secret Data Embedding. In this section, we introduce how to achieve secret data embedding by utilizing the block

selection strategy demonstrated in Section 3.3. The process of our design is described in detail as follows:

Step 1: prediction error calculation and overflowing determination. Take the i th block selected for example, suppose its complexities for both sides are smaller than t_1 and t_2 , respectively, the prediction errors can be defined as

$$\begin{aligned} e_{\max}^i &= x_{\sigma(n)}^i - x_{\sigma(n-1)}^i, \\ e_{\min}^i &= x_{\sigma(1)}^i - x_{\sigma(2)}^i. \end{aligned} \quad (7)$$

Suppose the number of valid cases on both sides are K_1 and K_2 , the prediction errors are denoted as $E_{\max} = (e_{\max}^1, e_{\max}^2, \dots, e_{\max}^{K_1})$ and $E_{\min} = (e_{\min}^1, e_{\min}^2, \dots, e_{\min}^{K_2})$. Considering the possible overflowing problem, we employ the same disposal as in [32] to generate and record the location maps as a part of auxiliary information.

Step 2: secret data and auxiliary information embedding. Take the i th block for example, if the complexity values for both sides are appropriate, and the positions are not in the location map, the process of secret data embedding on the maximum side can be denoted as

$$x_{\sigma(n)}^i = \begin{cases} x_{\sigma(n)}^i + b, & \text{if } e_{\max}^i = 1, \\ x_{\sigma(n)}^i + 1, & \text{if } e_{\max}^i > 1. \end{cases} \quad (8)$$

The similar procedure is also applied for the other side as follows:

$$x_{\sigma(1)}^i = \begin{cases} x_{\sigma(1)}^i - b, & \text{if } e_{\min}^i = -1, \\ x_{\sigma(1)}^i - 1, & \text{if } e_{\min}^i < -1. \end{cases} \quad (9)$$

On finishing secret data embedding, we record end positions and define them to be another part of auxiliary information, which is further embedded into least significant bits from the beginning of the cover image. In order to ensure reversibility, the original LSBs whose length is known to the receiver are embedded together with secret data sequence.

3.5. Secret Data Extraction. On receiving a stego-image, we present how to extract the secret data and recover the cover image in this section.

Step 1: auxiliary information extraction and block selection. Extract the auxiliary information according to the predetermined length from the beginning of the cover image. And divide the stego-image into blocks of the same size as in secret data embedding. Then calculate the complexity value for each one of the block and obtain the complexity sequences C'_{\max} and C'_{\min} as

well as prediction error sequences $E'_{\max} = (e_{\max}^{1'}, e_{\max}^{2'}, \dots, e_{\max}^{K_1'})$ and $E'_{\min} = (e_{\min}^{1'}, e_{\min}^{2'}, \dots, e_{\min}^{K_2'})$.

Step 2: overflowing pixels locating and data extraction. According to the location map, locate the corresponding positions that are at risk of overflowing. And then extract the secret data to recover the original cover image. Take the i th block for example, suppose both sides are appropriate, the process of data extraction and pixel value recovery on the maximum side are shown follows:

$$b = \begin{cases} 0, & \text{if } e_{\max}^{i'} = 1, \\ 1, & \text{if } e_{\max}^{i'} = 2, \end{cases} \quad (10)$$

$$x_{\sigma(n)}^i = \begin{cases} x_{\sigma(n)}^i - 1, & \text{if } e_{\max}^{i'} \geq 2, \\ x_{\sigma(n)}^i, & \text{if } e_{\max}^{i'} = 1. \end{cases}$$

Similarly, the process on the minimum side is as

$$b = \begin{cases} 0, & \text{if } e_{\min}^{i'} = -1, \\ 1, & \text{if } e_{\min}^{i'} = -2, \end{cases} \quad (11)$$

$$x_{\sigma(1)}^i = \begin{cases} x_{\sigma(1)}^i + 1, & \text{if } e_{\min}^{i'} \leq -2, \\ x_{\sigma(1)}^i, & \text{if } e_{\min}^{i'} = -1. \end{cases}$$

3.6. Discussion. According to the design of the proposed scheme, in order to ensure visual quality, only those blocks good in smoothness are selected to embed data, which inevitably affects the maximum embedding capacity. Actually, how to achieve the trade-off between embedding capacity and imperceptibility is an important issue, which is necessary to explore in future research. Based on this work, it is suggested to investigate more accurate complexity calculation methods and seek solutions to increase the embedding capacity with the imperceptibility guaranteed at the same time.

4. Experimental Evaluation

In this section, we take experiments to evaluate the effectiveness and performance of our proposed scheme. Specifically, we employ a workstation with Intel Core i5-6500 CPU @ 3.20 GHz, 4 GB RAM, and a 1 TB hard drive to implement all algorithms using Python version 3.0. We select three images Boat, Man, and Airport from the USC-SIPI image data set [33]; three high definition images Vanessa, Philipp, and Mohammad from the UNSPLASH image data set [34]; and three medical images 8FDE, Aweek, and RXtorace from the real world ChestX-ray14 image data set [35] separately as test images to perform our experiments, each of which is shown in Figures 3(a)–3(h), and 3(i), respectively. Our scheme is compared with PVO scheme [12], Pan et al.'s scheme [22], and Tang et al.'s two schemes [23] in performance. All results are on average of 20 tries.

4.1. Comparison of the Prediction Error Histogram. First, we calculate the maximum capacities of nine test images under the block size of 2×2 . For a certain appropriate block, whose complexity is less than the predefined threshold value, take the maximum side for example, only when the prediction error obtained by equation (7) is 1, the largest pixel is able to embed 1 bit according to (8). Otherwise, pixel value shifting is performed instead. Check both sides for each one of the blocks and count the total number of embeddable cases. The results are shown in Table 1.

According to the table, the maximum capacities of the nine test images are 7,226, 16,406, 51,399, 2,251,398, 2,130,241, 1,940,066, 21,413, 25,397, and 24,081 bits, respectively. In order to intuitively exhibit the advantage of proposed scheme, we first select 5,000 blocks from the 9 test images, respectively, through block selection strategy for data embedding or pixel value shifting and present the corresponding prediction error histograms in Figures 4(a)–4(h), and 4(i) for comparison among our proposed scheme and the other four schemes.

As shown in Figure 4, five prediction error histograms are plotted in a single coordinate system, each of which is centered at zero. Compared with other methods, the PEH generated by our proposed method is obviously higher in height, which means a smaller prediction error is obtained. To a certain extent, it indicates that the complexity value calculated by our proposed method is much closer to the true situation in reality. According to our design, more smooth blocks are selected based on the obtained complexity value, thus a larger percentage of smaller prediction errors could be obtained as a result.

4.2. Comparison of the Imperceptibility

4.2.1. Subjective Evaluation. In order to compare the imperceptibility of test images, we first employ subjective evaluation to determine the quality of the stego image based on the intuitive feeling of human observers. Specifically, we embed the maximum amount of secret bits into the nine test images and show the results in Figures 5(a)–5(h), and 5(i), respectively. It is clear that there is no obvious difference in visual quality compared with the original ones in Figures 3(a)–3(h), and 3(i).

4.2.2. Objective Evaluation. In this part, we employ the peak signal-to-noise ratio (PSNR) to measure the objective distortion introduced by data embedding. The higher the PSNR value, the better the imperceptibility is achieved. Specifically, we embed different amount of secret bits into the nine test images and show the results in Figures 6(a)–6(h), and 6(i), respectively. It is clear from these figures that our scheme exhibits a better performance in imperceptibility than any other schemes for most of the time. For instance, under the embedding capacity of 2,705, 8,115, 13,526, 18,936, 24,346, 29,757, 32,462, 37,872, 43,283, and 51,399 bits, the PSNRs of airport obtained by our scheme are 66.82, 61.82, 59.76, 58.29, 57.31, 56.50, 55.84, 55.31, 54.79, and 54.37 dB, each of which is obviously higher than 65.76, 60.90, 58.53, 56.94, 55.82,

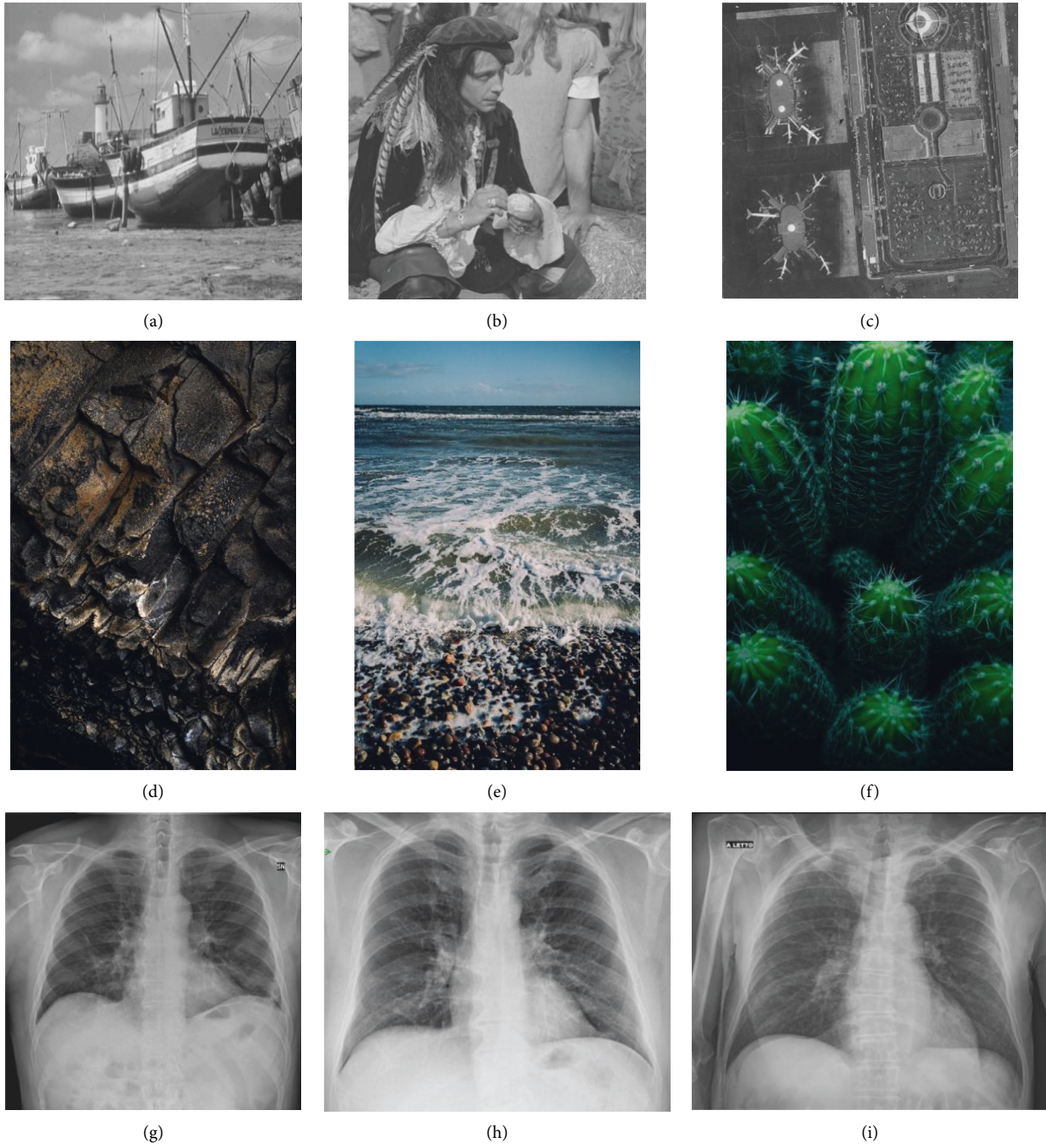


FIGURE 3: Test images: (a) Boat, (b) Man, (c) Airport, (d) Vanessa, (e) Philipp, (f) Mohammad, (g) 8FDE, (h) Aweek, and (i) RXtorace.

TABLE 1: The maximum capacities (MC) of nine test images from USC-SIPI data set [33], UNSPLASH data set [34], and ChestX-ray14 data set [35].

Image	Boat	Man	Airport	Vanessa	Philipp	Mohammad	8FDE	Aweek	RXtorace
MC (bits)	7,226	16,406	51,399	2,251,398	2,130,241	1,940,066	21,413	25,397	24,081

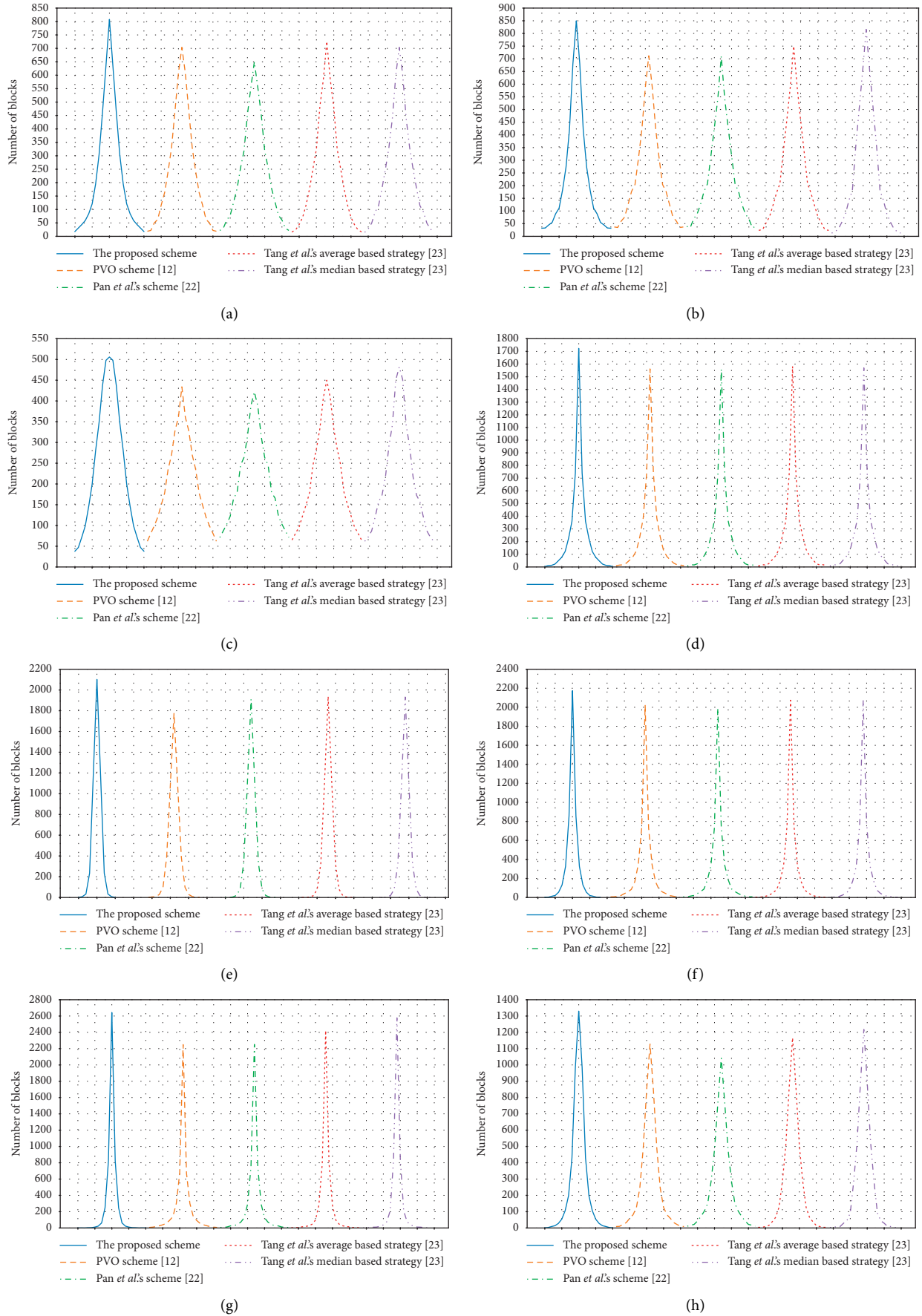


FIGURE 4: Continued.

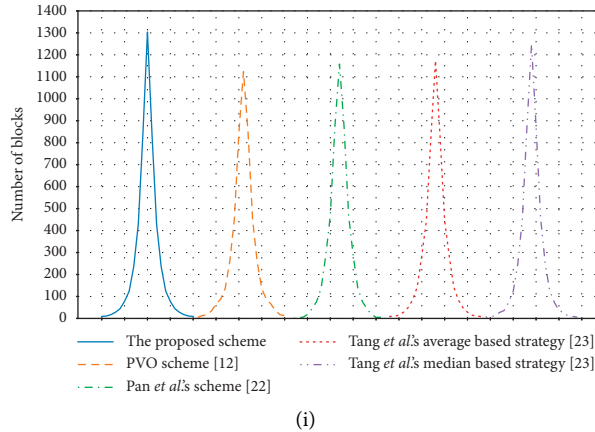


FIGURE 4: Comparison of PEH between our method and PVO scheme [12], Pan et al.'s scheme [22], and Tang et al.'s two schemes [23]: (a) Boat, (b) Man, (c) Airport, (d) Vanessa, (e) Philipp, (f) Mohammad, (g) 8FDE, (h) Aweek, and (i) RXtorace.

54.97, 54.29, 53.77, 53.26, and 52.77 dB in PVO scheme, 65.62, 60.56, 58.28, 56.96, 55.90, 55.06, 54.45, 53.87, 53.31 and 52.85 dB in Pan et al.'s scheme, and 65.97, 60.91, 58.71, 57.37, 56.30, 55.53, 54.85, 54.22, 53.69, 53.27 dB in Tang et al.'s average-based strategy, 66.37, 61.35, 59.18, 57.78, 56.74, 55.94, 55.21, 54.61, 54.14, and 53.68 dB in Tang et al.'s median based strategy. The reason is that in PVO scheme, only two internal pixels participate in complexity calculation, which is not accuracy enough to evaluate the real smoothness of blocks. Similar to PVO scheme, even though Pan et al.'s scheme considers both the internal and external pixels, the number participates in calculation is still rather limited. When it comes to Tang et al.'s schemes, more context pixels participating in calculation contribute to a better performance. However, those invalid ones are still treated equally, which is another factor to restrain the performance.

As a comparison, our proposed weighted average-based complexity calculation scheme not only ensures enough number of context pixels to participate in complexity calculation but also eliminates those invalid ones to improve performance. Furthermore, we assign different weights to each one of the valid context pixels to distinguish its contribution to the final complexity value. Thus, the result of complexity calculation is able to well reflect the real smoothness of the block, which is exhibited in the experimental results. Even though the maximum embedding capacity is achieved, thanks to the threshold value of block level complexity employed, our scheme is still better in imperceptibility. In this case, every one of the appropriate blocks, whose prediction error equals to 1 at the maximum side or -1 at the minimum side are selected to embed data. Take the test image airport for example, the PSNR value obtained by our scheme is 54.37 dB, higher than 52.77 dB in PVO scheme, 52.85 dB in Pan et al.'s scheme, 53.27 dB and 53.67 dB in Tang et al.'s two schemes.

It is worth noting that the results in real-world medical data set provide a novel way to expand the application of the proposed scheme. In fact, the proposed scheme, thanks to the high imperceptibility achieved, is able to be widely used in any occasions requiring high quality of cover signals, such

as the scenario of reversible data hiding in medical images. For privacy protection and convenience of reference, many medical images carry basic information such as patient identity, creation time, and so on. Since most physician consultations and medical records use printed case images, if the embedding of the information causes a significant distortion in image quality, it will inevitably affect the professional judgement of the physician and even endanger the health of patients. While, our method eliminates these risks.

5. Security Analysis

In this section, we take experiments to analyze the security of the proposed scheme. Specifically, we validate the reversibility at first, and then perform the universal steganalysis by comparing the normalized cross correlation (NCC), bit error rate (BER), and the targeted steganalysis by analyzing the regular singular (RS).

5.1. Reversibility Verification. In order to verify the reversibility of the proposed scheme, we extract secret data from the stego image and recover the cover image before comparing with the original one. Take Boat in Figure 3(a) for example, we divide it into 2×2 fixed size blocks at first, each of which is sorted according to the calculated complexity value before embedding 7,226 bits of data in order. On finishing data embedding, we perform the same procedures to the stego image and extract 7,226 bits successively. Then, the recovered image is obtained and shown in Figure 7(b), which is compared with the original one in Figure 7(a) under the exclusive OR operation. The result is shown in Figure 7(c), which is a binary image. It is clear that the result image is all black, which means the recovered image is completely consistent with the original one. Thus, the reversibility of our proposed scheme is validated.

5.2. Steganalysis. Steganalysis focuses on detecting whether a certain cover image has already been embedded, which is able to be utilized to evaluate the security of the proposed scheme.

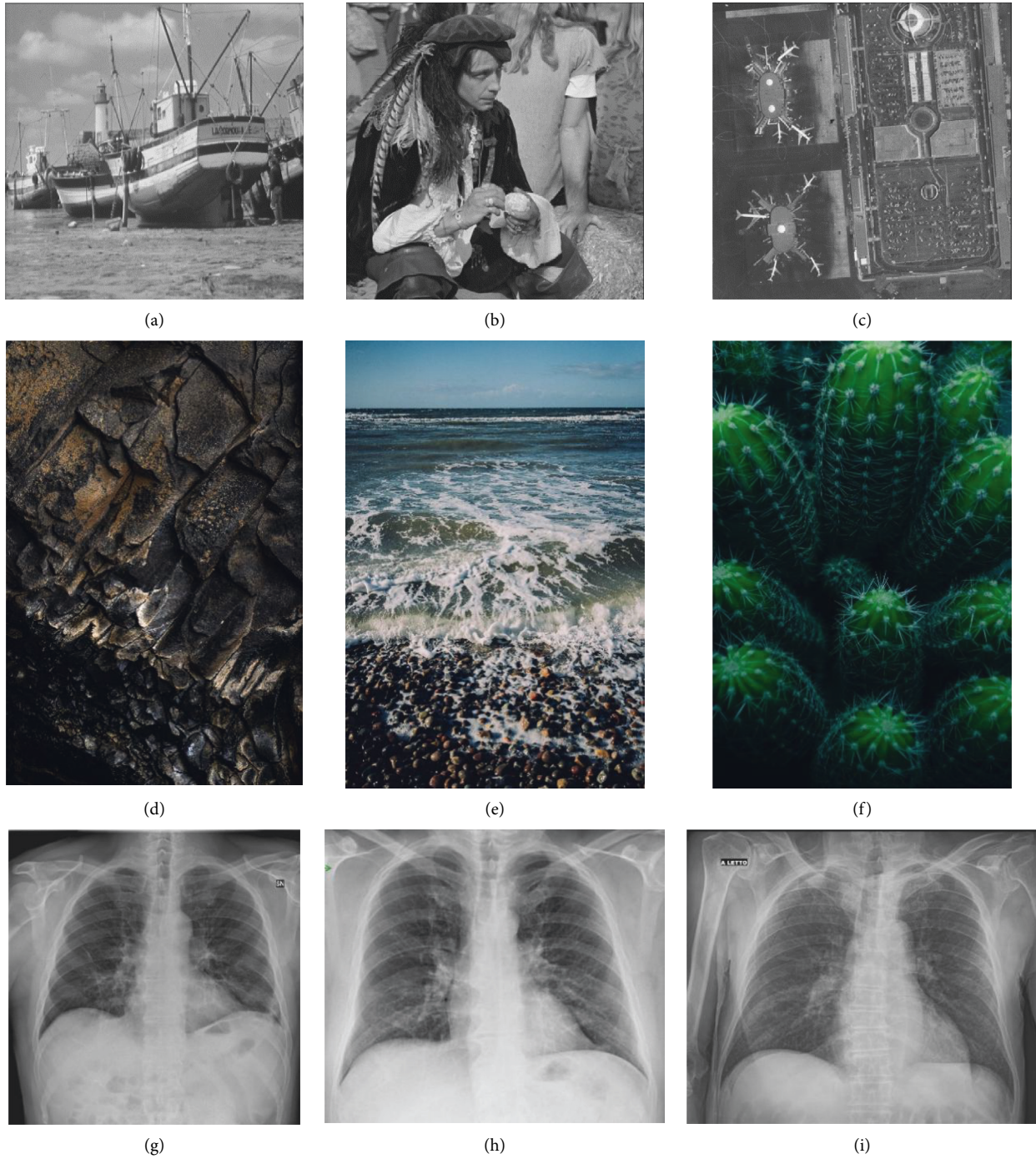
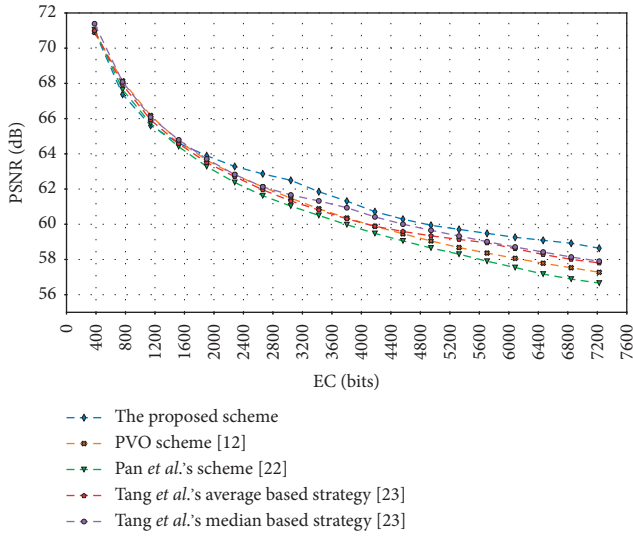


FIGURE 5: Stego images: (a) Boat, (b) Man, (c) Airport, (d) Vanessa, (e) Philipp, (f) Mohammad, (g) 8FDE, (h) Aweek, and (i) RXtorace.

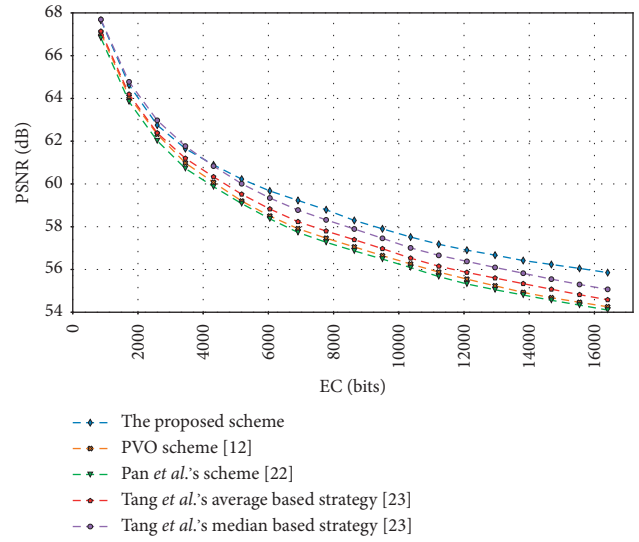
According to whether it is focused on a particular embedding algorithm, steganalysis can be classified into universal steganalysis and the targeted one. In particular, we employ NCC and BER to achieve universal detection and RS analysis to achieve targeted detection. NCC measures the correlation between the cover image and the stego one. The larger the value, the better the concealment of the designed steganography scheme since the two images are more similar. Besides, BER exhibits the difference between the cover image and the stego one at the pixel level. A smaller BER means less

modification is introduced to the cover image after data embedding. In addition, RS analysis checks the existence of secret data by analyzing the statistical characteristics of least significant bits of pixels after data embedding. The details of steganalysis from the three aspects are as follows.

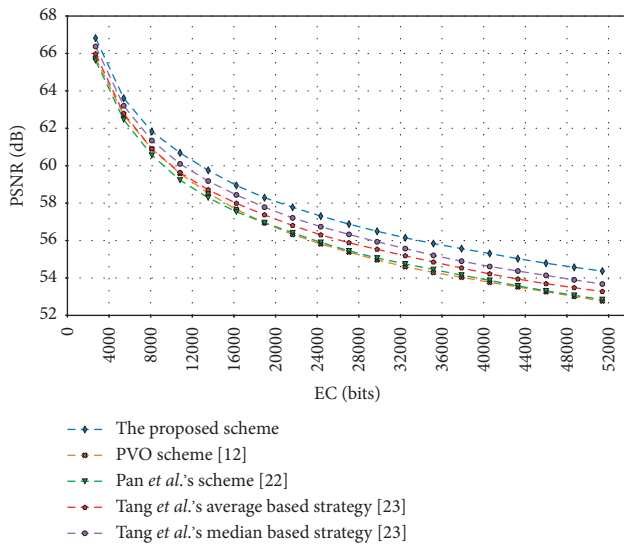
5.2.1. NCC. NCC is employed to evaluate similarity between the original cover image and the stego one at the maximum capacity. Suppose the size of both images is $M \times N$, the value is defined as



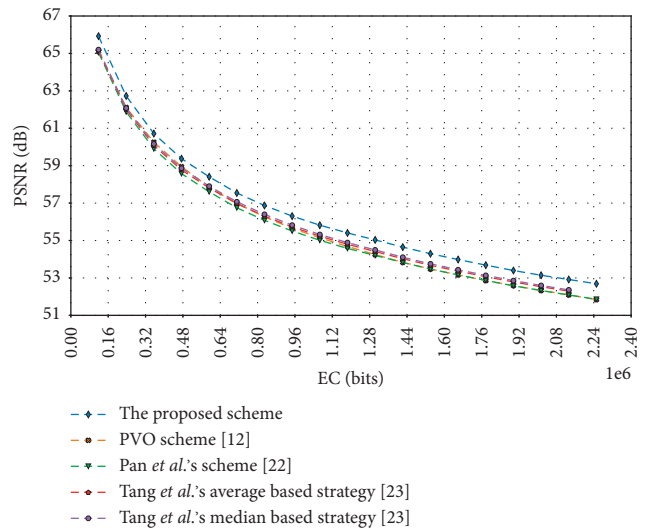
(a)



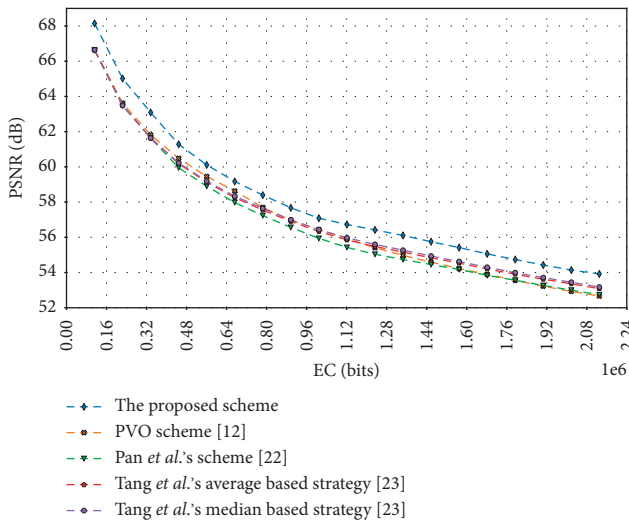
(b)



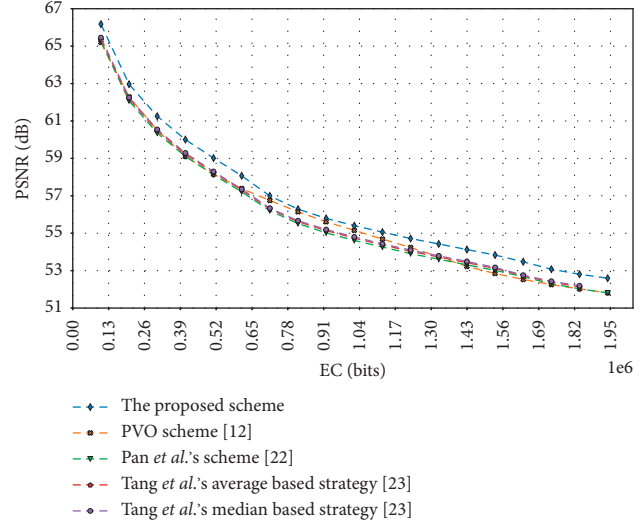
(c)



(d)



(e)



(f)

FIGURE 6: Continued.

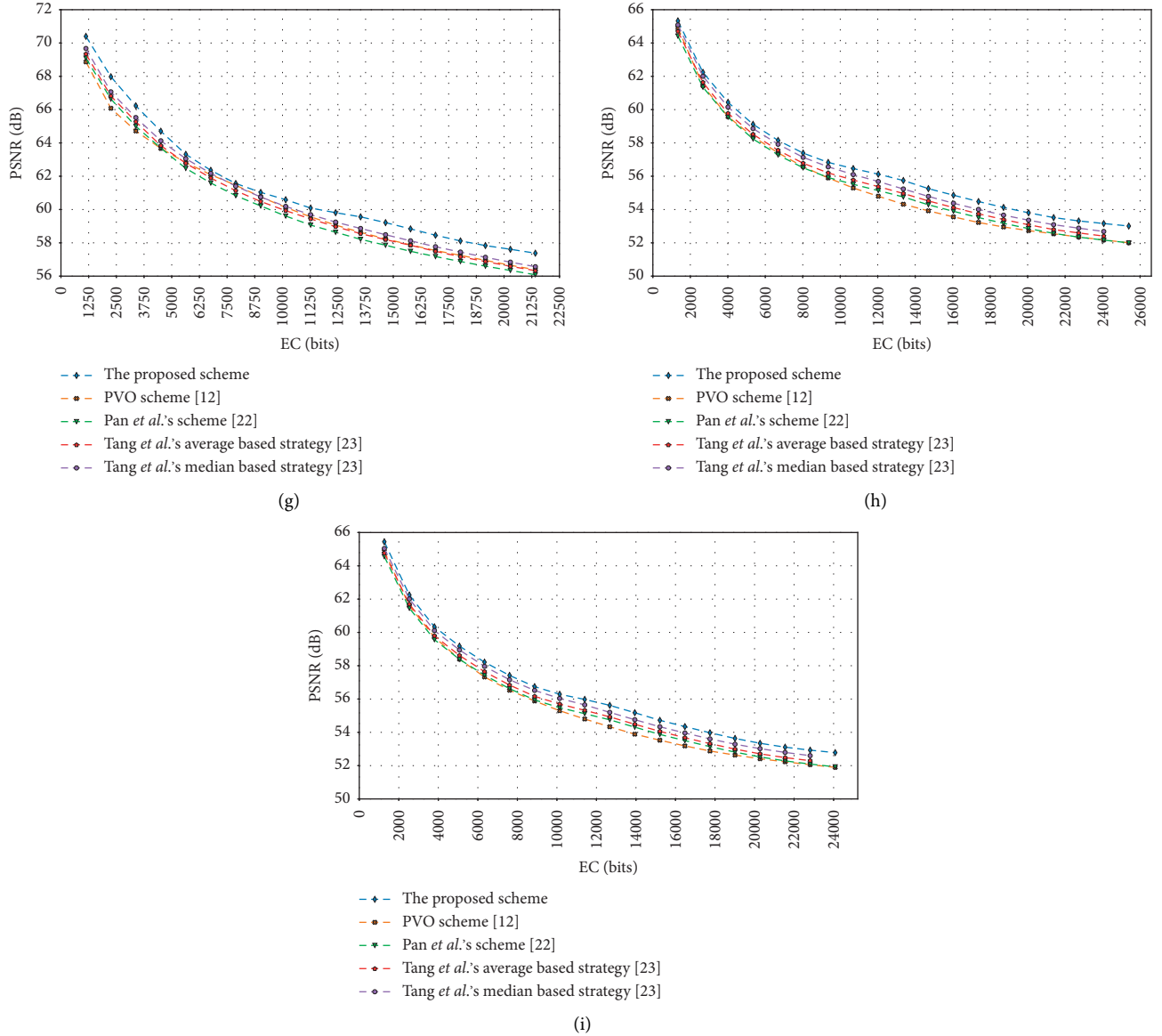


FIGURE 6: Comparison of imperceptibility between our method and PVO scheme [12], Pan et al.'s scheme [22], and Tang et al.'s two schemes [23]: (a) Boat, (b) Man, (c) Airport, (d) Vanessa, (e) Philipp, (f) Mohammad, (g) 8FDE, (h) Aweek, and (i) RXtorace.

$$\text{NCC} = \frac{\sum_{i=1}^M \sum_{j=1}^N (X(i, j) - \bar{X}(i, j))(Y(i, j) - \bar{Y}(i, j))}{\sqrt{\sum_{i=1}^M \sum_{j=1}^N (X(i, j) - \bar{X}(i, j))^2 \times \sum_{i=1}^M \sum_{j=1}^N (Y(i, j) - \bar{Y}(i, j))^2}} \quad (12)$$

In the definition, $X(i, j)$ and $Y(i, j)$ and $i \in [1, M]$, $j \in [1, N]$ represent pixel values at the i th row and j th column of the original cover image and stego image, respectively. It is clear that the value of NCC ranges in $[-1, 1]$. When NCC is equal to -1 , the two images are negatively correlated. On the contrary, when NCC approaches to 1, the two images are positively correlated, which means they are completely consistent. The closer this value is to 1, the more similar the two images are. For our proposed scheme, the NCCs of nine test images are shown in Table 2.

According to the table, the values of NCC range from 0.9998856 to 0.9999726, each of which approaches to 1. In particular, for test images Vanessa and Philipp, even though more secret bits are embedded, the values of NCC are still large enough. It means that the original cover image is similar to the stego one. It is to say, our scheme achieves high concealment after data embedding.

5.2.2. BER. In this part, we utilize BER to exhibit the difference between the cover image and the stego one at the pixel level in the case of completely embedded.



FIGURE 7: Verification of reversibility. (a) Original boat. (b) Recovered boat. (c) XOR result.

TABLE 2: Comparison of NCC with different images.

Image data sets	Images	NCC
USC-SIPI [33]	Boat	0.9999289
	Man	0.9999365
	Airport	0.9998856
UNSPLASH [34]	Vanessa	0.9999282
	Philipp	0.9999726
	Mohammad	0.9998986
ChestX-ray14 [35]	8FDE	0.9999376
	Aweek	0.9999555
	RXtorace	0.9999423

TABLE 3: Comparison of BER with different images.

Image data sets	Images	BER
USC-SIPI [33]	Boat	0.0275650
	Man	0.0625839
	Airport	0.0490179
UNSPLASH [34]	Vanessa	0.0938082
	Philipp	0.0990778
	Mohammad	0.0938662
ChestX-ray14 [35]	8FDE	0.0499127
	Aweek	0.1450843
	RXtorace	0.1279693

Specifically, BER measures the percentage of modified pixels during the process of data embedding, which is defined as

$$\text{BER} = \frac{MP}{M \times N}, \quad (13)$$

where MP is the number of modified pixels and $M \times N$ denotes the total number.

According to the results shown in Table 3, the values of BER for the nine test images range from 0.0275650 to 0.1450843, each of which approaches to 0. In particular, for test images Vanessa and Philipp with large maximum embedding capacity, even though more secret bits are embedded, the values of BER are as low as 0.0938082 and 0.0990778, since the invalid shifting is reduced according to our scheme. It means that few modifications are introduced

in fact. Thus, the concealment of the proposed scheme is guaranteed.

5.2.3. RS Analysis. RS analysis measures the statistical characteristics of least significant bits of pixels to determine whether secret data are embedded [36]. Take Boat with size of 512×512 in Figure 3(a) for example, we first embed 7,226 bits of data into the image, and then divide pixels into 65,536 disjoint pixel groups, each of which contains 4 pixels. Consider pixels x_1, x_2, x_3, x_4 in any one of the groups, the corresponding smoothness function is defined in (14). The smaller the obtained value, the stronger the correlation between adjacent pixels. Otherwise, the group of pixels is more likely to contain secret data.

$$f(x_1, x_2, x_3, x_4) = \sum_{i=1}^3 |x_{i+1} - x_i|. \quad (14)$$

Then define 3 flip functions on each one of the groups, according to the way to change original pixels. Specifically, in function F_0 , no operation is required to original pixels. In function F_1 , the least significant bits of original pixels are flipped over, so that the values of which are transformed between 2 and 3, 4 and 5, \dots , 254 and 255. As to function F_{-1} , the two least significant bits are entailed in the flipping process. Take four pixels in a certain group for example, we first generate four elements $m_1, m_2, m_3,$ and m_4 in a set M , each of which randomly takes a value among $\{-1, 0, 1\}$. In particular, we define M to be $\{1, 0, 1, 0\}$, and $-M$ to be $\{-1, 0, -1, 0\}$. Then calculate the values of corresponding

TABLE 4: Results of RS analysis.

Image data sets	Images	R_M	R_{-M}	S_M	S_{-M}
USC-SIPI [33]	Boat	33,178	32,504	20,966	21,617
	Man	35,402	35,906	17,279	17,155
	Airport	125,402	126,360	88,099	87,416
UNSPLASH [34]	Vanessa	3,848,600	3,855,095	811,246	813,817
	Philipp	3,629,664	3,628,511	723,583	722,901
	Mohammad	3,562,855	3,531,772	424,576	424,559
ChestX-ray14 [35]	8FDE	74,459	74,592	12,011	12,315
	Aweek	31,462	31,130	4,393	4,571
	RXtorace	30,445	30,687	7,543	7,227

smoothness functions $F_M = f(F_{m_1}(x_1), F_{m_2}(x_2), F_{m_3}(x_3), F_{m_4}(x_4))$ and $F_{-M} = f(F_{-m_1}(x_1), F_{-m_2}(x_2), F_{-m_3}(x_3), F_{-m_4}(x_4))$. If F_M is larger than $f(x_1, x_2, x_3, x_4)$, the group is classified as a regular one. On the other hand, if F_M is smaller, the group is considered to be a singular one. Otherwise, the group is defined to be an invariant group. Perform the similar process by comparing F_{-M} and $f(x_1, x_2, x_3, x_4)$. Record the number of regular and singular groups for both cases as R_M, R_{-M} , and S_M, S_{-M} respectively. According to the conclusion of Fridrich et al. [36] obtained by extensive experimental analyses, if R_M approximates to R_{-M} , and S_M approximates to S_{-M} , the least significant bits of pixels in the detected image do not contain secret data with high probability.

We perform RS analysis to the selected 9 test images and show the results in Table 4. According to the table, for each image, R_M approximates to R_{-M} , and S_M approximates to S_{-M} , which validates the security of our proposed scheme under RS analysis.

6. Conclusion

In this paper, we put forward a weighted average-based complexity calculation strategy for block selection-oriented reversible data hiding. Specifically, we first assign weights to each one of the pixels with the help of its adjacent neighbors, to ensure that the context pixel in a smooth area is assigned a larger weight. By embedding secret bits into blocks with smaller complexity value at first, the number of invalid shifting is able to be largely reduced in PVO-based reversible data hiding, thus the imperceptibility is guaranteed. Experimental results show that our proposed scheme is better in imperceptibility comparing with the state of the art when secret message is not fully embedded. Therefore, our scheme is suitable for any field that requires high-quality cover images.

Data Availability

The image data used to support the findings of this study are available from the USC-SIPI image data set (<https://sipi.usc.edu/database>), the UNSPLASH image data set (<https://www.unsplash.com>), and the ChestX-ray14 image data set (<https://www.kaggle.com/nih-chest-xrays/data/>).

Conflicts of Interest

The authors have no conflicts of interest to report regarding the present study.

Acknowledgments

This work was specially supported by Research Funds for NSD Construction, University of International Relations (2021GA08), National Natural Science Foundation of China (62172053; 62102113), Student Academic Research Training Project of University of International Relations (3262022SYJ04; 3262022SYJ12), Beijing Municipal Natural Science Foundation (4212031), and Fundamental Research Funds for the Central Universities, University of International Relations (3262021T13).

References

- [1] Z. C. Ni, Y. Q. Shi, and N. Ansar, "Reversible data hiding," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 16, no. 3, pp. 354–362, 2006.
- [2] X. Tang, L. N. Zhou, D. Liu, L. Boyu, and L. Xinyi, "Reversible data hiding based on improved rhombus predictor and prediction error expansion," in *Proceedings of the IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications*, (IEEE TrustCom), pp. 13–21, Guangzhou, China, December 2020.
- [3] L. N. Zhou, W. J. Shan, X. Tang, and H. Bingwei, "Reversible data hiding algorithm with high imperceptibility based on histogram shifting," in *Proceedings of the 13th IEEE Cyber, Physical and Social Computing (CPSCom)*, pp. 628–633, IEEE, Norway, Europe, November 2020.
- [4] L. Chu, H. W. Wu, and Y. L. Zeng, "Improved weighted average-based rhombus predictor in reversible data hiding using prediction error expansion," in *Proceedings of the IEEE the 6th International Conference on Computer and Communication Systems (ICCCS)*, pp. 6–11, IEEE, Chengdu, China, April 2021.
- [5] X. M. Liu, B. L. Zhang, and Y. C. Wen, "An improved image authentication method using QR code watermarking approach," in *Proceedings of the 8th International Conference on Artificial Intelligence and Security (ICAIS)*, Qinghai, China, December 2022.
- [6] Y. Q. Shi, Z. Ni, and D. Zou, "Lossless data hiding: fundamentals, algorithms and applications," in *Proceedings of the 34th IEEE International Symposium on Circuits and Systems (ISCAS)*, pp. 33–36, IEEE, Canada, CA, USA, May 2004.

- [7] R. Caldelli, F. Filippini, and R. Becarelli, "Reversible watermarking techniques: an overview and a classification," *EURASIP Journal on Information Security*, vol. 2010, Article ID 134546, 19 pages, 2010.
- [8] B. Ma, J. C. Hou, C. P. Wang, X. M. Wu, and Y. Q. Shi, "A reversible data hiding algorithm for audio files based on code division multiplexing," *Multimedia Tools and Applications*, vol. 80, no. 12, Article ID 17569, 2021.
- [9] X. M. Liu and X. Tang, "Image authentication using QR code watermarking approach based on image segmentation," in *Proceedings of the IEEE 19th International Conference on Trust, Security and Privacy in Computing and Communications*, (IEEE TrustCom), pp. 1572–1577, Guangzhou, China, December 2020.
- [10] X. Tang, L. N. Zhou, D. Liu, W. Shan, and Y. Zhang, "Border following-based reversible watermarking algorithm for images with resistance to histogram overflowing," *International Journal of Distributed Sensor Networks*, vol. 16, no. 5, Article ID 155014772091701, 2020.
- [11] X. Tang, D. Liu, and L. N. Zhou, "A high capacity reversible watermarking algorithm based on block-level prediction error histogram shifting," in *Proceedings of the 3rd ACM International Conference on Artificial Intelligence and Pattern Recognition*, (ACM AIPR), pp. 133–139, Xiamen, China, May 2020.
- [12] X. L. 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.
- [13] W. G. He, K. Zhou, J. Cai, L. Wang, and G. Xiong, "Reversible data hiding using multi-pass pixel value ordering and prediction-error expansion," *Journal of Visual Communication and Image Representation*, vol. 49, pp. 351–360, 2017.
- [14] B. Ou, X. Li, Y. Zhao, and R. Ni, "Reversible data hiding using invariant pixel-value-ordering and prediction-error expansion," *Signal Processing: Image Communication*, vol. 29, no. 7, pp. 760–772, 2014.
- [15] B. Ou, X. Li, and J. Wang, "High-fidelity reversible data hiding based on pixel-value-ordering and pairwise prediction-error expansion," *Journal of Visual Communication and Image Representation*, vol. 39, pp. 12–23, 2016.
- [16] F. Peng, X. Li, and B. Yang, "Improved PVO-based reversible data hiding," *Digital Signal Processing*, vol. 25, pp. 255–265, 2014.
- [17] H. Wu, X. Li, Y. Zhao, and R. Ni, "Improved reversible data hiding based on pvo and adaptive pairwise embedding," *Journal of Real-Time Image Processing*, vol. 16, no. 3, pp. 685–695, 2019.
- [18] X. Wang, J. Ding, and Q. Pei, "A novel reversible image data hiding scheme based on pixel value ordering and dynamic pixel block partition," *Information Sciences*, vol. 310, pp. 16–35, 2015.
- [19] C. Qin, W. Zhang, F. Cao, X. Zhang, and C. C. Chang, "Separable reversible data hiding in encrypted images via adaptive embedding strategy with block selection," *Signal Processing*, vol. 153, no. 12, pp. 109–122, 2018.
- [20] S. W. Weng, Y. Q. Shi, and W. Hong, "Dynamic improved pixel value ordering reversible data hiding," *Information Sciences*, vol. 489, pp. 136–154, 2019.
- [21] S. J. Xiang and G. Q. Ruan, "Efficient PVO-based reversible data hiding by selecting blocks with full-enclosing context," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 32, no. 5, pp. 2868–2880, 2022.
- [22] Z. B. Pan, X. Y. Gao, E. D. Gao, and G. Fan, "Adaptive complexity for pixel-value-ordering based reversible data hiding," *IEEE Signal Processing Letters*, vol. 27, pp. 915–919, 2020.
- [23] X. Tang, L. N. Zhou, and G. Tang, "Reversible data hiding based on improved block selection strategy and pixel value ordering," in *Proceedings of the IEEE the 6th Cyber Science and Technology Congress (CyberSciTech)*, pp. 619–627, IEEE, Canada, CA, USA, October 2021.
- [24] X. Tang, L. N. Zhou, and G. Tang, "Improved fluctuation derived block selection strategy in pixel value ordering based reversible data hiding," in *Proceedings of the 20th International Workshop on Digital-Forensics and Watermarking (IWDW)*, pp. 163–177, Springer, Beijing, China, January 2021.
- [25] B. Jana, D. Giri, and S. K. Mondal, "Dual-image based reversible data hiding scheme using pixel value difference expansion," in *Proceedings of the 3rd Information Systems Design and Intelligent Applications (INDIA)*, pp. 403–412, Springer, New Delhi, India, February 2016.
- [26] S. Mukherjee and B. Jana, "A novel method for high capacity reversible data hiding scheme using difference expansion," *International Journal of Natural Computing Research*, vol. 8, no. 4, pp. 13–27, 2019.
- [27] B. Jana, D. Giri, and S. Kumar Mondal, "Dual image based reversible data hiding scheme using (7, 4) hamming code," *Multimedia Tools and Applications*, vol. 77, no. 1, pp. 763–785, 2018.
- [28] B. Jana, "High payload reversible data hiding scheme using weighted matrix," *International Journal for Light and Electron Optics*, vol. 127, no. 6, pp. 3347–3358, 2016.
- [29] B. Jana, "Dual image based reversible data hiding scheme using weighted matrix," *International Journal of Electronics and Information Engineering*, vol. 5, no. 1, pp. 6–20, 2016.
- [30] P. Chowdhuri, B. Jana, and D. Giri, "Secured steganographic scheme for highly compressed color image using weighted matrix through DCT," *International Journal of Computers and Applications*, vol. 43, no. 1, pp. 38–49, 2021.
- [31] Y. J. Jia, Z. X. Yin, X. P. Zhang, and Y. Luo, "Reversible data hiding based on reducing invalid shifting of pixels in histogram shifting," *Signal Processing*, vol. 163, no. 5, pp. 238–246, 2019.
- [32] S. Kim, X. C. Qu, V. Sachnev, and H. J. Kim, "Skewed histogram shifting for reversible data hiding using a pair of extreme predictions," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 29, no. 11, pp. 3236–3246, 2019.
- [33] sipi, "The USC-SIPI Image Database," 1997, <http://sipi.usc.edu/database/>.
- [34] Unsplash, "The UNSPLASH Image Database," 2013, <https://www.unsplash.com/>.
- [35] Kaggle, "The ChestX-ray14 Image Database," 2015, <https://www.kaggle.com/nih-chest-xrays/data/>.
- [36] J. Fridrich, M. Goljan, and R. Du, "Invertible authentication watermark for JPEG images," in *Proceedings of the 2nd International Conference on Information Technology: Coding and Computing (ITCC)*, pp. 223–227, IEEE, Las Vegas, USA, August 2001.