Article

A Compression Resistant Steganography Based on Differential Manchester Code

Shuai Wang , Ning Zheng and Ming Xu *

School of Cyberspace, Hangzhou Dianzi University, Hangzhou 310018, China;
wang_hohhot95@hdu.edu.cn (S.W.); nzheng@hdu.edu.cn (N.Z.)

* Correspondence: mxu@hdu.edu.cn

Abstract: In the field of image steganography research, more and more attention is paid to the importance of stego image robustness. In order to make steganography accessible from laboratory to practical applications, it becomes critical that the stego images can resist JPEG compression from transmission channel. In this paper, an image steganography algorithm with strong robustness to resist JPEG compression is proposed. First, the robust cover elements are selected using the sign of DCT coefficients which are kept constant before and after JPEG compression. Additionally, a distortion function and a weighted cost adjustment method are designed to assign an appropriate cost to each candidate DCT coefficient. Finally, the message is embedded in the cover image which has minimal embedding distortion by flipping the signs of DCT coefficients, while differential Manchester code is applied to the element positions to obtain the location feature. Compared with the prior art, our algorithm has better undetectability and stronger robustness, and it can resist the attacks from the social network platforms such as Facebook, Twitter, and WeChat.

Keywords: robust steganography; JPEG compression; differential Manchester code; sign of DCT coefficients

Citation: Wang, S.; Zheng, N.; Xu, M. A Compression Resistant Steganography Based on Differential Manchester Code. *Symmetry* **2021**, *13*, 165. <https://doi.org/10.3390/sym13020165>

Received: 30 November 2020

Accepted: 17 January 2021

Published: 21 January 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Steganography is a branch and art of information hiding technology. Different from digital watermarking technology whose purpose is to protect copyrights [1], steganography needs to embed the secret message into multimedia communication covers such as text [2], image [3], audio [4], and video [5], in a way that is imperceptible to human perception. At present, image steganography has become the most popular research direction.

The essence of image steganography is to make small changes to pixels or coefficients in the spatial domain or frequency domain of the cover image so that the secret message can be embedded, and these slight changes will not cause huge changes in the image for people visually. Digital image steganography algorithms are usually divided into two categories: non-adaptive steganography and adaptive steganography. Early common non-adaptive steganography algorithms that can only be applied in the spatial domain include Least Significant Bit replacement (LSBr), random $\pm K$ steganography, and Least Significant Bit matching (LSBm), while the adaptive steganography algorithms can be applied in the spatial and frequency domains. The current classic adaptive steganography algorithms include HUGO [6], WOW [7], HILL [8], and S-UNIWARD [9] in the spatial domain. In addition, the current classic frequency domain steganography algorithms include UED [10], UERD [11], J-UNIWARD [9], and SI-UNIWARD [9]. In other words, the performance of adaptive strategy has been increasingly improved by minimizing non-adaptive distortion in steganography [12–14].

However, these adaptive steganography algorithms usually cannot completely and correctly extract the secret message when the stego image suffers attacks, such as JPEG compression, cropping, scaling, and noise addition. The essence of steganography is to transmit the secret message, and the complete extraction of the secret message is the task

of steganography. Therefore, the technique of steganography needs to move from the laboratory to the real world for a wide range of applications, meaning that the robustness of it should be mainly addressed.

Zhang et al. proposed a series of steganography algorithms based on robust domain [15–17]. The similarity of these algorithms is to find the stable characteristic relationship of the cover image before and after compression, such as the size or position relationship among DCT coefficients within or between blocks. Tao et al. [18] proposed a new type of robust steganography algorithm through the intermediate images based on coefficient adjustment. Based on transmission channel matching, Zhao et al. [19] repeatedly compress the image to reduce the influence of JPEG compression on the channel and directly improve the robustness of stego images. In addition, Zhu et al. also proposed a robust steganography method by flipping the sign of DCT coefficients [20]. Recently, based on the generalized Dither modulation and extended embedding domain [21], and based on the strategy of adaptive dither adjustment [22], the robust steganography algorithm has been further developed. Encouragingly, the above algorithms perform very well in resisting JPEG compression attacks, and at the same time guarantees a high accuracy of correct extraction of the secret message.

However, most steganographic methods have poor performance of resisting JPEG compression in social networks. Therefore, we designed a robust steganographic scheme with location features that can resist some JPEG compressions like random compression, repeated compression, and force compression. The contributions of the paper are as follows:

1. We design a robust steganography framework which embeds secret message by locating the robust domain of the image. This framework generates the location features by combining the position of the non-zero DCT coefficients and differential Manchester code. These procedures improve the robustness performance, which contributes to the complete extraction of the secret message after JPEG compression attacks.
2. We analyze and improve the sign steganography algorithm, propose a weighted cost adjustment method, and further optimize the cost selection of cover elements. Numerical experiments empirically demonstrate that our proposed embedding algorithm largely improves the robustness performance of stego image when facing the challenge of JPEG compression attack.
3. Our algorithm can be applied to a variety of social media platforms and can resist channel processing operations that only include JPEG compression attacks. At the same time, the secret message can still be completely extracted after repeated compression or force compression.

The rest of this paper is organized as follows. In Section 2, several popular robust steganography methods are classified and introduced. The principle of differential Manchester code and weighted cost adjustment are introduced in detail in Section 3. In Section 4, experiments and results are given, and conclusions are drawn in Section 5.

2. Related Works

In the current study, JPEG compression is widely used in simulated channels in the laboratory. A variety of complicated attacks in real social platforms will lead to the failure of some robust steganography algorithms that have performed well in the laboratory. Of the two categories of robust steganographic algorithms, we believe that the method relying on the image's robust domain has better robustness when applied to the real social platforms. It is a superior way to find stable relationships in the image to embed the message.

To the best of our knowledge, in order to fulfill the task of robust steganography, most current studies are prone to sacrifice embedding capabilities to improve the safety of covert communication. Furthermore, image steganography studies typically use four metrics to measure the overall performance of different algorithms:

- Capacity: the stego image carries the maximum number of the secret bits.

- Imperceptibility: the stego image is indistinguishable from the cover image in human visual perception.
- Undetectability: the stego image can bypass the detection from modern advanced steganalysis.
- Robustness: the stego image is capable of preserving the embedded message when suffering different post-processing attacks.

Most of the current robust steganography has not been tested by real network. In fact, the robustness of some steganographic algorithms performs well in laboratory, but fails to meet expectations in real social platforms. The performance in real platforms also belongs to “Robustness”, so we conducted experiments in social platforms such as Facebook, Twitter, and WeChat.

In the following, we give a general overview of several typical robust steganography methods and describe in detail the Flipping of the Sign of DCT Coefficients algorithm proposed by Zhu et al.

2.1. Symmetry Methods

In this section, we offer six popular robust steganography methods as the symmetry method, which are included in our classification of current robust steganography algorithms above. The specific ideas of the six methods are as follows:

- DCRAS method [15]: In this method, the embedding domain is constructed through the connection between DCT coefficients. DCRAS divides the DCT coefficients into several 8×8 DCT blocks for the mean value relationship between the DCT coefficients at a certain position. The corresponding position in the three adjacent DCT blocks is basically stable before and after JPEG compression. The authors construct the embedding domain based on this characteristic to extract the cover element from it. At the same time, the secret message is encoded with the RS error correction code and embedded in the cover element to obtain the stego element.
- FRAS method [16]: In this method, the authors first use Harris–Laplacian features [23] to extract the robust domain, then use channel simulation to attack each domain to find the domains suitable for embedding, and finally use the DCRAS algorithm to construct the embedding domain in these domains and extract the cover elements to strike the balance between undetectability and robustness. FRAS combines feature domain selection, and the DCRAS algorithm is better than DCRAS in terms of safety and robustness.
- DMAS method [24]: In this method, according to the JPEG compression quantization table, the authors use Dither modulation to construct the embedding domain. This algorithm uses the principle of compression to construct a more stable embedding domain, which reduces the modification range of the DCT coefficients when embedding the message. Thus, its robustness and undetectability are further improved compared with the two algorithms above.
- TCM method [19]: In this method, the authors devised a steganographic method based on matching channels. TCM (Transport Channel Matching), as the main idea of this method, depends on repeated compression of the stego images to reduce errors caused by quantization and rounding. The original image processed by TCM is embedded as the cover image. The algorithm adopts error correction code and cost function to generate the temporary stego image, and uses the transmission channel to repeatedly compress the stego image. In this process, the error correction code will be operated iteratively when the message is incorrectly extracted, until the correctly extracted stego image is generated or the threshold value of the error correcting is met.
- Zernike moment method [25]: Pixels between the original neighboring pixels after scaling attack will be lost or increased. In this method, Zhang et al. proposed a countermeasure based on Zernike moments which can map the characteristic pixels of an image, and has rotation invariance as a set of orthogonal matrices. In other words, scaling the image will not change Zernike moments. The sender extracts the

Zernike moments of the image, uses the method of Dither modulation to embed the secret message into the normalized Zernike moments, and then reconstructs the stego image.

- MCCI method [18]: In this method, Tao et al. proposed a new robust steganography framework. The main idea behind the method is to modify and adjust the differences between the DCT coefficients of the original image and the stego image. First, we compress the image and use any frequency domain steganography method to generate the stego image. The intermediate image is obtained by adjusting the DCT coefficients of the original image depending on the stego image; then, the intermediate image is compressed again through the previous channel to obtain a new stego image. The method ensures that the image generated by the intermediate image compression is the same as the stego image, so the secret message can be completely extracted.

2.2. Prior Work

The algorithm proposed in this paper is an improvement of the algorithm presented in Zhu et al. in [20]. They proposed a method in order to construct embedding domains based on Flipping the Sign of DCT Coefficients (FSDC). In this section, we introduce the FSDC algorithm, including the principle, the cost function design, the location feature generation, and the embedding method. These contents are also applied to the methods proposed in this paper.

The main idea of FSDC algorithm is shown in Figure 1, the gray block represents the DCT coefficient of a certain 8×8 pixels of the cover image, and the blue block represents the DCT coefficient that becomes zero after the JPEG pre-compression. The red block is the DCT coefficient with strong robustness, which is modified to generate the stego element. In addition, we will save the non-zero AC coefficients and the corresponding positions according to the cover elements as a preliminary location feature.

Figure 1. The main idea of the Flipping the Sign of DCT Coefficients algorithm proposed in [20].

The first step in the construction of robust domain is to obtain the cover element from the cover image I with a quality factor of Q . It is known that JPEG compression operations can make some DCT coefficients become zero, so we need to further filter out the non-zero AC coefficients in I from those that do not become zero after the JPEG compression attack. Therefore, we perform a pre-compression operation on I using a quality factor of Q' to obtain a compressed image of I' before extracting the cover elements, where Q' is smaller than Q . The compressed image I' is only a reference for the initial selection of the cover element which takes no part in the rest of the process of stego image generation.

By means of a pre-compression operation, we can obtain the cover elements as shown by Equation (1):

$$\mathbf{Map}_{[c_{i,j}, L_{i,j}]} = \begin{cases} [1, (i, j)], & x_{i,j} > 0 \\ [0, (i, j)], & x_{i,j} < 0 \\ \backslash, & x_{i,j} = 0 \end{cases} \quad (1)$$

where $x_{i,j}$ represents the DCT coefficients of the i th row and the j th column in the image $\mathbf{I}'$, $c_{i,j}$ and $L_{i,j}$ represent the cover elements of the i th row and the j th column and position coordinates, respectively. We select the coefficients that are non-zero DCT coefficients in the image $\mathbf{I}'$ and record them in the location feature **Map**.

In order to ensure that the stego image has the minimal changes compared to the cover image, the authors here use a steganographic algorithm for minimizing embedded distortion to obtain the cost of each cover element. The cost of embedding the generalized distortion function S-UNIWARD algorithm is the sum of the relative changes in the wavelet coefficients of the stego image, as shown in the equation:

$$D(\mathbf{I}, \mathbf{S}) \triangleq \sum_{k=1}^3 \sum_{i=1}^n \frac{|W_i^{(k)}(\mathbf{I}) - W_i^{(k)}(\mathbf{S})|}{\sigma + |W_i^{(k)}(\mathbf{I})|} \quad (2)$$

where $\mathbf{I}$ and $\mathbf{S}$ are the cover image and the stego image of the spatial domain; n is the total number of DCT coefficients of $\mathbf{I}$. $W_i^{(k)}(\mathbf{I})$, $k = \{1, 2, 3\}$, $i \in \{1, \dots, n\}$ is the i th wavelet coefficient of the first k subband of the first decomposition layer of the cover image filtered by the wavelet filter. $W_i^{(k)}(\mathbf{S})$ is the corresponding wavelet coefficient of the stego image; σ is a stable constant and $\sigma > 0$.

FSDC embeds the message based on the stability of the sign of DCT coefficient, so we use the J-UNIWARD algorithm to decompress the image into the spatial domain using Equation (3):

$$D(\mathbf{I}, \mathbf{S}) \triangleq D(J^{-1}(\mathbf{I}), J^{-1}(\mathbf{S})) \quad (3)$$

where $J^{-1}(\mathbf{I})$, and $J^{-1}(\mathbf{S})$ denotes the decompression of the image into the spatial domain. In order to satisfy the need to minimize embedding distortion, we should select the appropriate embedding region according to the texture complexity of the image, so J-UNIWARD pre-filters the image in multiple directions to identify textured areas, which are generally less costly to distort.

The cost of distortion can be compared to the sum of the additive costs of each pixel, and we use ρ_i^J to represent the cost of modifying the coefficient x_i to y_i , so that the Equation (3) can be expressed as:

$$D(\mathbf{I}, \mathbf{S}) \triangleq \sum_{i=1}^n \rho_i^J(\mathbf{I}, y_i) \quad (4)$$

where ρ^J represents the sum of the costs.

Based on the invariance principle of the sign of the DCT coefficient, we have flipped some coefficients in order to embed the message. In this process, modifying non-zero coefficients with smaller absolute values will result in less distortion, so we will select DCT coefficients from the cover elements with absolute values less than or equal to τ . At the same time, we also need to determine the range of the cost of this location with ρ^J , where a threshold value of μ is set. We need to select coefficients whose cost is less than μ , indicating that this location is in a complex textured region of the spatial image.

The equation of the steganographic algorithm that determines the minimal embedding distortion cost is as follows:

$$\rho'(x_{i,j}) = \begin{cases} |x_{i,j}|, & |x_{i,j}| \leq \tau, \rho^J(x_{i,j}) < \mu \\ |x_{i,j}|^\sigma, & |x_{i,j}| > \tau, \rho^J(x_{i,j}) < \mu \\ \rho^J(x_{i,j}), & \rho^J(x_{i,j}) \geq \mu, \end{cases} \quad (5)$$

where μ and τ represent threshold values. We use the absolute value of the DCT coefficients as the new cost of this position. When the absolute value of the coefficients is less than or equal to τ while ρ^J is less than μ , the elements will be appropriate. For the elements that are within the texture area but have a value above the threshold, we use $\sigma > 1$ to increase the cost of the element. Similarly, when the element is on the smooth region, we retain its large distortion function cost.

In order to minimize the average embedding distortion in the cover image, we use STCs encoding when acquiring the cover elements. STCs encoding is close to the theoretical minimum of additive average embedding distortion and is the preferred solution for many steganographic algorithms. Equation (6) for embedding message into a cover image using STCs encoding is as follows:

$$\mathbf{S} = \text{Emb}(\mathbf{I}, \mathbf{m}) = \arg \min_{\mathbf{S} \in C(\mathbf{m})} D(\mathbf{I}, \mathbf{S}) \quad (6)$$

where $\mathbf{m}$ represents the secret message, $C(\mathbf{m}) = \{z \in \{0, 1\}^n | \mathbb{H}z = \mathbf{m}\}$ is the corresponding chaperone of $\mathbf{m}$ and the parity matrix $\mathbb{H} \in \{0, 1\}^{m \times n}$ is a submatrix $\mathbf{H}$ stitched together in a diagonal cascade with the matrix dimension $h \times w$. The submatrix $\mathbf{H}$ is represented as a key parameter that is shared by the sender and the receiver. By means of a distortion function and a weighted cost adjustment method, we obtain the cost of each DCT coefficient in the cover element. The STCs encoding then minimizes the distortion between the cover image and the stego image within the limits of both.

After obtaining the cover element $\mathbf{c}$, the embedding cost ρ , the secret message $\mathbf{m}$, and the key matrix $\mathbf{H}$, Equation (7) is used to obtain the stego element $\mathbf{s}$:

$$\mathbf{s} = F_{STCs}(\mathbf{c}, \rho(\mathbf{I}), \mathbf{m}, \mathbf{H}). \quad (7)$$

Then, we flipped the sign of DCT coefficients in the cover image $\mathbf{I}$ according to the stego element $\mathbf{s}$ to generate the stego image $\mathbf{S}$:

$$y_{i,j} = \begin{cases} x_{i,j}, & \mathbf{s}_{i,j} > 0 \\ -x_{i,j}, & \mathbf{s}_{i,j} < 0 \end{cases} \quad (8)$$

where $x_{i,j}$ and $y_{i,j}$ are the DCT coefficients in the i th row and the j th column of the cover image and the stego image, respectively. By the means of Equation (8), we flipped the sign of DCT coefficients on the cover image to produce the stego image. When $\mathbf{s}_{i,j} > 0$, the DCT coefficients in this position become positive; otherwise, they become negative.

3. Proposed Method

This paper proposes optimizations and improvements based on the FSDC algorithm, the algorithm is called Differential Manchester Code with Sign Steganography (DMCSS), and the framework is shown in Figure 2. We retain the ideas of flipping the sign of the DCT coefficients and location features by the FSDC algorithm, and propose the differential Manchester code and weighted cost adjustment method. The algorithm proposed in this paper further improves the location features and the cost function so that DMCSS is further improved over the FSDC algorithm.

Figure 2. A framework for robust steganographic algorithms based on a differential Manchester code.

3.1. Motivation

By studying the FSDC algorithm, we found that its undetectability and robustness could be further improved. As to the undetectability, FSDC uses the value of the DCT coefficients as the cost at this position (see Equation (5)). This design does not accurately identify the texture region in which the DCT coefficients are located, so we proposed a weighted cost adjustment method to further optimize the original cost. Regarding the robustness, FSDC guarantees the correct extraction rate of the message by flipping the sign of DCT coefficients with the location features. However, in the Zhu's paper, FSDC cannot guarantee the correct extraction of all the message because some DCT coefficients become zero. Therefore, we proposed the differential Manchester code to further optimize the location features, which greatly improves the robustness of the stego images. In addition to this, we retain other methods of FSDC, such as the method of image pre-processing and the method of using the J-UNIWARD algorithm that obtains the complex texture regions.

We need to make greater use of location features. Through the principle of digital codes, we use the idea of the differential Manchester code to enhance the stability of the location features. Figure 3 illustrates the conversion of three types of digital data to digital codes. It is well known that digital data are stored as binary bits in a computer and needs to be converted to a digital signal before it can be propagated on a channel. The most basic is the baseband digital signal, which changes level according to the code element, marking 1 with a high level and 0 with a low level, while, in IEEE 802.3 LANs, the Manchester coding method is used, which provides for jumping from high to low marking 1 and from low to high marking 0. This rule can also be reversed, and is mainly mutually recognized by the two communicating parties. In IEEE 802.5 LANs, the differential Manchester code has been proposed in order to improve the immunity of Manchester coding to interference. In Figure 3, solid arrows indicate no jumps and hollow arrows indicate jumps.

Figure 3. Digital data to digital signal coding.

3.2. Weighted Cost Adjustment

According to Equation (5), Zhu et al. combine the absolute magnitude of the DCT coefficient with the complex texture region of image. By setting a threshold between the absolute value of the DCT coefficient and the cost of J-UNIWARD distortion, the value of the DCT coefficient that is smaller than both is directly taken as the cost of the coefficient, but then the cost of J-UNIWARD distortion will only function as a range restriction. However, we would prefer to provide more information for our new cost selection.

Assume that the absolute value of both coefficients x_1 and x_2 is 1, but their distortion function cost $\rho'(x_1)$ is much smaller than $\rho'(x_2)$. According to Equation (5), both x_1 and x_2 have a final cost of 1. In fact, however, we prefer x_1 with a smaller distortion cost of $\rho'(x_1)$ in the image, so here we introduce a weighted cost adjustment algorithm to further refine the selection of the costs:

$$\rho(x_{i,j}) = \rho'(x_{i,j}) \times (\rho^J(x_{i,j})/10 + \delta) \% \quad (9)$$

where we use the cost ρ' obtained from Equation (5) and the J-UNIWARD cost ρ^J at the corresponding position to obtain the weighted cost ρ . We use δ to control the parameter adjustment. Experimentally, we use $\delta = 50$ as a parameter for the algorithm.

3.3. Differential Manchester Code

Based on the idea of differential Manchester code, we improve the steganographic algorithm to enhance robustness. Since we only need to record the sign of the stego element instead of their values, the use of differential Manchester code strengthens the correlation between the coefficients and makes the DCT coefficients normalized. At the same time, the storage size of the location feature can be reduced to some extent, which further improves the security. The location feature only represents the position of the non-zero DCT coefficient in the cover image, and not all positions are embedded information. In the proposed algorithm, the cover image only embeds the secret message and the location feature can be made public. In the communication process, the secret message can be extracted

only when the receiver knows the algorithm, the location feature, the corresponding stego image, and the $\mathbf{H}$ matrix.

Suppose we want to send many stego images to different receivers, we can communicate with the receiver in advance to determine the $\mathbf{H}$ matrix, and then generate the stego images and location features. We make location features files public on platforms, such as Discuz, Cloud Storage, and GitHub. When communicating with the receiver in advance, we can agree on a rule to let the receiver to obtain his own files from the numerous location feature files on the platform. For example, the file whose name is “6” at the end of the file name is the location feature file of receiver A; the file whose file name is a multiple of 4 is the location feature file of receiver B, and so on. Finally, we send the stego images through social media; for example, sending one-to-one or one-to-many by Wechat or Twitter (of course, a rule needs to be agreed in advance so that the receiver knows which images belong to him). Although the way that we make location features public in platform will reduce security, we can improve the security of covert communications as much as possible by setting communication rules, adding obfuscated files in the stego images or the location features.

3.3.1. Differential Manchester Encoding

First, we update the set of coordinates $\mathbf{Map}$ in Equation (1) according to Equation (8) by replacing the cover element $\mathbf{c}$ with the DCT coefficient y of the cover image to obtain the set of cover coordinates $\mathbf{Map}_{[y_{i,j}, L_{i,j}]}$:

$$\mathbf{c}_{i,j} = y_{i,j} \quad (10)$$

The differential Manchester code requires us to uniformly specify the first $y_{(1)}$ of the sequence of DCT coefficients in the coordinate set $\mathbf{Map}_{[y_{i,j}, L_{i,j}]}$:

$$DM_{(1)} = \begin{cases} 1, & y_{(1)} > 0 \\ 0, & y_{(1)} < 0 \end{cases} \quad (11)$$

where $y_{(g)}$ represents the g th bit of the sequence of DCT coefficients, which is independent of the position of $y_{i,j}$. Taking into account the current position of the coefficients in $\mathbf{Map}$, $DM_{(g)}$ represents the g th bit of the differential Manchester code sequence. In the algorithm, $DM_{(1)}$ is determined by the sign of one DCT coefficient $y_{(1)}$. When $y_{(1)} > 0$, $DM_{(1)} = 1$, and when $y_{(1)} < 0$, $DM_{(1)} = 0$. Starting from $DM_{(2)}$, every acquisition of $DM_{(g)}$ needs to combine $y_{(g)}$ with $y_{(g-1)}$. The equation is as follows:

$$DM_{(g)} = \begin{cases} 1, & y_{(g)}/y_{(g-1)} < 0 \\ 0, & y_{(g)}/y_{(g-1)} > 0 \end{cases} \quad (12)$$

where $g > 1$. When $y_{(g)}$ and $y_{(g-1)}$ are of the opposite sign, $DM_{(g)} = 1$, and $DM_{(g)} = 0$ when they have the same sign. Furthermore, we need to replace the DCT coefficient sequence y with the obtained DM sequence in $\mathbf{Map}$ to obtain the final location feature. As the differential Manchester code sequence is obtained according to the position relationship in the queue and the unchangeability of the position after JPEG compression, the algorithm provides a further guarantee of the complete extraction of message. The following figure shows an example of using differential Manchester code.

As shown in Figure 4, the blue arrow indicates that the first bit in the DCT coefficient sequence is “−2” which is less than 0, determining $DM_{(1)} = 0$. The fourth bit “9” and the fifth bit “−5” pointed by the red arrow are of the opposite signs, determining $DM_{(5)} = 1$. The eighth bit “2” and ninth bit “4” pointed by the green arrow are of the same sign, determining $DM_{(9)} = 0$. By the above rules, we obtain the differential Manchester code label for the entire sequence of DCT coefficients.

DCT Coefficients	-2	57	-24	9	-5	-4	3	2	4	-2	14
Differential Manchester Encoding	0	1	1	1	1	0	1	0	0	1	1

Figure 4. Example of differential Manchester encoding.

3.3.2. Differential Manchester Decoding

The receiver receives the stego image transmitted over the channel and obtains a sequence of stego DCT coefficients according to the sequence L in the location feature **Map**. Noting that the DCT sequence generated here needs to be arranged strictly in the order of L , which is extracted as follows:

$$y_{i,j} = \begin{cases} y_{i,j}, & L_{i,j} \neq 0 \\ \text{wrong}, & L_{i,j} = 0 \end{cases} \quad (13)$$

In contrast to the embedding process, the sequence of stego DCT coefficients is extracted from the stego image according to the L in the location feature. The lossy DCT coefficient y is recovered from the compressed image. $y_{i,j}$ representing the DCT coefficient of the i th row and the j th column. When the DCT coefficient at this position becomes zero, we record this position as *wrong* to provide a marker for the message recovery using differential Manchester codes in the next subsection.

3.3.3. Correcting Incorrect Elements

For DCT coefficients marked as *wrong*, which can cause the extraction of the secret message to fail, we use the differential Manchester code in the location feature and the sign of the coefficient in front of the *wrong* position to determine the sign of this position. This equation is as follows:

$$s_{(g)} = \begin{cases} \Delta_{(g-1)}, & DM_{(g)} = 0 \\ -\Delta_{(g-1)}, & DM_{(g)} = 1 \end{cases} \quad (14)$$

where $\Delta_{(g)}$ denotes the sign of the g th bit of $y_{(g)}$. When $y_{(g)}$ is marked with *wrong*, we get $\Delta_{(g)}$ based on $DM_{(g)}$ and $y_{(g-1)}$. When $DM_{(g)}$ is 1, $y_{(g)}$ and $y_{(g-1)}$ are of the opposite sign and $\Delta_{(g)} = -\Delta_{(g-1)}$; conversely, when $DM_{(g)}$ is 0, $y_{(g)}$ and $y_{(g-1)}$ are of the same sign and $\Delta_{(g)} = \Delta_{(g-1)}$. Figure 5 shows an example of using differential Manchester decoding.

DCT Coefficients	-2	57	-24	9	-5	-4	3	2	4	-2	14
Differential Manchester	0	1	1	1	1	0	1	0	0	1	1
Lossy DCT Coefficients	-2	57	-24	9	0	-4	3	0	0	0	14
Repair DCT Coefficient Sign	-2	57	-24	9	-	-4	3	+	+	-	14

Figure 5. Example of differential Manchester decoding.

We repair the complete sign elements by differential Manchester decoding and obtain the stego element from the sign information as follows:

$$s_{(g)} = \begin{cases} 1, & \Delta_{(g)} = "+" \ \&\& \ y_{(g)} > 0 \\ 0, & \Delta_{(g)} = "-" \ \&\& \ y_{(g)} < 0 \end{cases} \quad g \neq 1 \quad (15)$$

where we reset the coefficients according to their plus and minus signs, and for the $\Delta_{(g)}$ obtained by differential Manchester decoding, we set “+” to 1 and “−” to 0. In this case, the s sequence as the extracted stego element contains only binary bits {0, 1}.

4. Numerical Experiments

In this section, many experiments are designed to verify the performance of the algorithm proposed in this paper. First, parameters are determined which are used in subsequent experiments; then, several general performance metrics of the algorithm are evaluated, including capacity, imperceptibility, undetectability, and robustness.

4.1. Experiment Setup

The images used in experiments are from the image baseline dataset BOSSbase ver. 1.01 [26] contains 10,000 grayscale raw images in PGM format with 512×512 pixels. All images are compressed to generate the cover images whose quality factors are 75, 85, and 95. The cover images adopt different steganography algorithms to generate the stego images with payloads of 0.01 to 0.1. Meanwhile, we use STCs [27] code to minimize the distortion embedding. Several types of representative robust steganography methods are selected to compare with DMCSS, which include DCRAS, FRAS, DMAS, TCM and MCCI.

DCRAS, FRAS, and DMAS are similar to the DMCSS in terms of the category of steganographic methods which all belong to steganography based on the robust domain of the cover image. TCM is a robust steganography method that matches the side-information of the channel, which requires to repeatedly compress the image to obtain the channel information. While MCCI is different from the two categories above, which studies the relationship between the stego image and the cover image, and uses the transmission channel as a steganography tool to complete the covert communication process. Finally, the J-UNIWARD algorithm is also adopted, which is a classic and popular steganography method with good undetectability performance. Although this method does not have robustness, it can be used as a better reference among other performance metrics.

Table 1 shows the general settings of the experiments in this paper. Some settings will be adjusted in subsequent sections. The details will be illustrated and explained in the following parts.

Table 1. Experiment Setup.

Cover Image		Steganography Algorithm	Attack Process	
			JPEG Compression	Steganalysis
Image source	BOSSbase 1.01	DMCSS		
Image size	512×512	DCRAS		PSNR
Number of images	10,000	FRAS	75	CCPEV
Image format	JPEG	DMAS	85	CCJRM
Quality factor	75, 85, 95	TCM	95	
Payload	0.01 to 0.1	MCCI		

4.2. Determination of Parameters

In this section, we need to determine the values of τ and μ in Equation (5). They are the threshold values to determine the cost. In Equation (5), the function of σ is to exponentially calculate the coefficients larger than τ in the texture region, which makes the coefficients have higher cost, so that the message is preferentially embedded in small DCT coefficients. In addition, experiments show that $\sigma = 3$ is the optimal choice for the algorithm. The smaller the parameter μ indicates that the point is in the region with more complex texture. In the FSDC algorithm, the author proves through experiments that $\mu = 500$ is the optimal choice, so we use this conclusion.

The parameter τ is used to determine the size of the DCT coefficients. We use $QF = 60$ to pre-compress 100 cover images whose QFs are 75. Figure 6 is a histogram of the average distribution of non-zero DCT coefficients of 100 compressed images. According to the

figure, most of the non-zero DCT coefficients are between $[-10, 10]$, where coefficients with absolute values of $\{1, 2, 3\}$ occupies the majority. The payload set in the experiment is between 0.01 and 0.1; at this time, the number of DCT coefficients with a value of $\{1, 2, 3\}$ is sufficient to meet the embedding requirement, so we set $\tau = 3$. In fact, through subsequent experiments, it can be concluded that the absolute value of most embedding coefficients is 1 or that of a few embedding coefficients is less than or equal to 3.

Figure 6. The frequency diagram of DCT coefficients.

4.3. Capacity Analysis

The capacity refers to the maximum number of secret bits embedded into the image. The capacity in the frequency domain is the number of non-zero AC coefficients, and capacity in the spatial domain is the number of pixels. In this section, we compress 10,000 pgm images into the cover images with $QF = \{75, 85, 95\}$. At the same time, we compress the image to simulate channel attacks with $QF = 75$. As the image quality factor decreases, the number of non-zero AC coefficients also decreases. Here, we select two representative algorithms, namely TCM and MCCI, to compare with DMCSS. All three of these cover two common types of robust steganography and a special robust steganography method. Among the three methods, the capacity of MCCI is the number of non-zero AC coefficients of the cover image, while the capacity of the other two methods is restricted by the corresponding algorithms (see Table 2).

The number of non-zero AC coefficients changes when QF changes. Accordingly, MCCI has the largest capacity, where all non-zero AC coefficients can be used as the cover elements for embedding. While both TCM and DMCSS have to select robust DCT coefficients as the cover elements for embedding. Because TCM needs to repeatedly compress the images with the same QF for reserving the invariance of DCT coefficients, a similar capacity appears among the original cover images with different QFs. As a matter of fact, TCM and DMCSS sacrifice to some extent capacity for guaranteeing robustness.

Table 2. Comparison of the average capacity performance of the three algorithms.

Method	Quality Factor	75	85	95
	DMCSS	41,529	28,748	31,525
	TCM	33,387	33,362	33,440
	MCCI	41,592	56,206	98,065

4.4. Imperceptibility Performance

The stego image should remain perceptual distortion as small as possible, referring to imperceptibility. In the experiments for this section, we use three algorithms to perform steganographic embedding on cover images whose $QF = \{75, 85, 95\}$ with the payload of 0.1, and obtain nine sets of stego images. Here, the Peak Signal to Noise Ratio (PSNR) is used as the criterion of imperceptibility. The higher the PSNR value is, the less perceptual distortion the stego image suffers.

Table 3 is the comparison among the average PSNR of TCM, MCCI, and DMCSS. In the comparison of three groups, MCCI and TCM have higher average PSNR performance. As QF increases, their average PSNR also increases.

Table 3. Comparison of the average PSNR of the three algorithms.

Quality Factor	Method	DMCSS	TCM	MCCI
75		46.8188	56.6977	57.4343
85		42.7146	56.6986	59.2747
95		38.5466	56.6959	62.6871

In addition, Figure 7 shows a comparison of the normalized histograms of 10,000 sets of PSNR under the three sets of QFs, which shows that the frequency of DMCSS is lower than the other two algorithms. When $QF = 75$, the histograms of TCM and MCCI are close to overlap, which means that the PSNR values of the two are very similar at the moment. It is worth noting that the PSNR values of the TCM algorithm have almost not changed in experiments. As for this phenomenon, it is not difficult to come to a reasonable explanation, that is, the DMCSS algorithm is inferior to the other two algorithms in terms of imperceptibility, which is due to the difference of the embedding principle. Most algorithms adopt the embedding method of ternary STCs with ± 1 embedding, which usually adds 1 or subtracts 1 from the DCT coefficient to embed the secret message. However, the DMCSS algorithm proposed in this paper takes advantage of the stability of the sign to complete the embedding, and the modification of the DCT coefficients needs to flip the sign. Compared to the ± 1 embedding, the operation makes twice the change from “−1” to “+1”, so the PSNR value is also smaller than the other two algorithms.

Figure 8 shows a cover image and the stego images generated using three algorithms. It can be found that the four images are similar visually, which means that, although the DMCSS algorithm is slightly inferior to other comparison algorithms in terms of PSNR, the difference between the three is basically imperceptible.

Figure 7. The normalized histogram comparison of 10,000 PSNR values.

Figure 8. The cover image and the stego images generated by using three algorithms.

4.5. Undetectability Performance

In this section, we evaluate the undetectability of the algorithm through steganalysis features. In the feature extraction method, we use rich model features CCPEV-548D [28] (PEV features enhanced by Cartesian calibration) and CCJRM-22510D [29] (Cartesian Calibrated JPEG domain rich model) which are two feature extraction methods. In addition, the “D” represents the feature dimension extracted from each image. At the same time, we use the ensemble classifier [30] to train features to test undetectability. Finally, P_E is usually regarded as an evaluation indicator to judge the effect of steganography, which represents the classification error rate, defined as follows:

$$P_E = \min \frac{P_{FA} + P_{MD}}{2} \quad (16)$$

where P_{FA} and P_{MD} represent the false alarm rate and missed detection rate, respectively. In this paper, we use the “out-of-bag” (OOB) error rate E_{OOB} instead of P_E to evaluate the undetectability performance. E_{OOB} is the unbiased estimated of P_E [11], which is also usually used as the undetectable evaluation standard in the field of steganalysis. The higher the value of E_{OOB} , the stronger the undetectability performance of the algorithm. In the training process, we choose half of the cover images and the stego images as the training set, and the other half as the test set. In addition, it is worth noting that the cover images and the stego images in the two sets should correspond one to one.

In this section, we mainly compare the undetectability performance when $QF = 75$. Under the premise of the same payloads, as the quality factor increases, the undetectability of the stego image will decrease. This is because the image with high quality factor has more non-zero DCT coefficients and has a wider degree of selectivity for the embedding position. DMCSS is based on flipping the sign of DCT coefficients to embed the message. For an image with a large quality factor, the flipping of sign with a larger absolute value will cause obvious distortion of the image, which is not suitable for embedding. On the contrary, for cover images with medium and small quality factors, their DCT coefficients

are relatively smaller, and the flipping of sign will also have a smaller cost, so their undetectability will be better than the images with large quality factors.

In other words, our algorithm is more suitable for cover images with medium and small quality factor to embed the secret message. Therefore, for the subsequent experiments in this section, we will uniformly adopt the cover images with $QF = 75$ for a variety of steganographic algorithms embedding and the stego image undetectability analysis.

We use DCRAS, FRAS, and DMAS algorithms to compare the undetectability performance with the DMCSS proposed in this paper. All of these four methods use CCPEV to extract features and then use the same classifier for training. Since the four algorithms are based on the robust steganography algorithm of the embedded domain, they belong to the vertical comparison. At the same time, we use CCPEV and CCJRM to compare with the undetectability performance of DMCSS under different feature extraction methods.

Figure 9 is a vertical comparison of the undetectability of the four algorithms of DMCSS, DCRAS, FRAS, and DMAS. According to the figure, it can be found that the DMCSS algorithm has the best undetectability at all payloads compared to other algorithms. When payload is 0.1, FRAS and DMAS algorithms can hardly resist the detection of steganalysis. The E_{OOB} of DMCSS is close to 0.2, which is 87% higher than DMAS. Therefore, DMCSS is superior to other methods based on the robust domains.

Figure 9. Illustration of undetectability of DMCSS, DCRAS, FRAS, and DMAS with CCPEV features.

Among the three algorithms of DCRAS, FRAS, and DMAS, the undetectable performance of the DCRAS algorithm is the worst, and its stego image can hardly resist steganalysis when the payload is large. FRAS optimizes the selection of the cover elements in the robust domain on the basis of DCRAS, so the undetectability has been further improved. It can be observed that DMAS performs best based on the embedding domain constructed by Dither modulation. In this case, the modification of the DCT coefficients due to embedding tends to be smaller compared to other methods.

Figure 10 shows the undetectability performance of DMCSS using CCPEV and CCJRM feature extraction algorithms. According to the broken line of the figure, it can be concluded that the steganalysis capability of CCJRM feature is stronger than that of the CCPEV feature. When the embedding rate is low, the undetectability of the two is close to 0.5, which means that the classifier can hardly distinguish between the stego image and the cover image. As the payload increases, the changes and distortions of the image also increase,

so the undetectability decreases. When the highest payload is 0.1, although the features extracted by CCJRM reduce the undetectability to 0.1592, the value is still higher than the undetectability of other algorithms using CCPEV in Figure 9, which proves that the undetectability of DMCSS is relatively good.

Figure 10. Illustration of undetectability of DMCSS when adopting CCPEV and CCJRM.

4.6. Robustness Performance

In this section, we need to evaluate whether the stego image can guarantee the integrity and accuracy of the secret message extracted after attacks (such as JPEG compression) during the transmission process. We propose to adopt the extraction error rate for comparing the robustness performance of these methods, which can be calculated by:

$$R_{\text{error}} = \frac{n_{\text{error}}}{n_{\text{msg}}} \quad (17)$$

where n_{error} denotes the number of incorrectly extracted bits, and n_{msg} is the total number of secret bits. In addition, the smaller the R_{error} , the stronger the robustness. At the same time, we define the number of correct extraction, which represents the total number of images that can completely reconstruct the secret message from all the stego images in the data set.

We compare DMCSS with the two algorithms of TCM and MCCI because the three algorithms represent robust steganography based on the image embedding domain, robust steganography based on channel side-information, and special steganography method, respectively. Therefore, such an experimental setup can horizontally compare the robustness of our method.

As shown in Table 4, DMCSS and MCCI have good robustness, so the secret message can be extracted after the compression of stego image. Due to the limitation of the TCM algorithm, some DCT coefficients still cannot eliminate the error after repeatedly compressing the image, making a few stego images fail to correctly extract the secret message after compression. However, as a whole, TCM has very few images that fail. Figure 11 shows the average R_{error} of the three methods. Similarly, due to the differential Manchester encoding method of DMCSS and the “compression and embedding” method of MCCI, they both have perfect robustness. Although the TCM algorithm has a high extraction error rate, the TCM still has unstable DCT coefficients that cannot be eliminated after repeated

channel matching, such elements will increase the overall extraction error rate. However, it is undeniable that, combined with Table 4 and Figure 11, TCM is still a superior robust steganography algorithm.

Table 4. The number ($\times 10^{-3}$) of images that can completely reconstruct secret bits.

Quality Factor	Methods	Payloads									
		0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.08	0.1
QF = 75	DMCSS	10	10	10	10	10	10	10	10	10	10
	TCM	9.983	9.986	9.984	9.983	9.980	9.981	9.982	9.983	9.981	9.980
	MCCI	10	10	10	10	10	10	10	10	10	10
QF = 85	DMCSS	10	10	10	10	10	10	10	10	10	10
	TCM	9.973	9.971	9.973	9.970	9.965	9.963	9.962	9.960	9.959	9.958
	MCCI	10	10	10	10	10	10	10	10	10	10
QF = 95	DMCSS	10	10	10	10	10	10	10	10	10	10
	TCM	9.923	9.920	9.914	9.917	9.913	9.906	9.905	9.903	9.897	9.895
	MCCI	10	10	10	10	10	10	10	10	10	10

Figure 11. Performance of resisting JPEG compression.

Table 5 is the average R_{error} of the stego images of MCCI and DMCSS after secondary compression. In fact, the MCCI algorithm compresses the intermediate images to generate the stego image, that is to say, the undetectability of MCCI is related to the cost function it adopts (J-UNIWARD is adopted in this paper). When there is no other interference in the channel, MCCI will always have the perfect robustness, which also makes the stego image unable to be compressed twice. For example, the stego images generated by using the cost function J-UNIWARD and the MCCI algorithm are equivalent to J-UNIWARD steganography directly performed on the cover image, which will have both perfect robustness and J-UNIWARD's high undetectability. However, when we compress it twice, it is equivalent to directly compressing the stego image generated by the J-UNIWARD algorithm. Because of the absence of the robustness in J-UNIWARD algorithm, its R_{error} will reach 0.5, which means that the secret message is unable to be extracted. For DMCSS, due to the existence of differential Manchester code, multiple compressions will still ensure its robustness.

Table 5. The R_{error} of MCCI and DMCSS after JPEG compression attack with $QF = 85$.

Methods	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.1
MCCI	0.5001	0.4997	0.4998	0.5002	0.4997	0.5001	0.4999	0.5001	0.5001	0.4999
DMCSS	0	0	0	0	0	0	0	0	0	0

In order to further explore the robustness of DMCSS, as shown in Figure 12, we randomly select a stego image with payload = 0.1, and perform a JPEG compression attack with $QF = \{75, 45, 25\}$, and then extract the secret message to compare the bits in order to get their n_{error} . As a result, all n_{error} are 0, which means that the DMCSS algorithm through differential Manchester encoding has good robustness after all compression attacks. The current mainstream steganography algorithm does not consider the phenomenon of force compressed image, but this algorithm can resist unknown JPEG attacks in various degrees, providing a reference for force compression research.

Figure 12. Illustration of stego image resists large compression.

4.7. Practical Performance

In this section, we will evaluate the practical performance of five methods on social network platforms, such as Facebook, Twitter, and WeChat. In addition, two popular adaptive steganographic schemes called J-UNIWARD and UERD are also compared. We randomly select 50 images in the BOSSbase at first, and compress them with $QF = 85$. Then, we embed the secret message with the payload of 0.1. All the stego images are uploaded to the social network platform, and then downloaded. The robustness performance is measured by R_c and R_{error} with Equation (17). R_c denotes the ratio of the number of images in which the secret message can be completely extracted to the number of all stego images.

As shown in Table 6, it can be seen that the secret messages can be completely extracted from 50 stego images generated by DMCSS, which means that DMCSS can resist the post-processing attacks from Facebook. However, other algorithms that perform well in a simulated environment can hardly resist Facebook's attacks. As we expected, the two adaptive steganography algorithms (J-UNIWARD and UERD) that have no robustness are not effective for secret communication in real social network platforms.

To our knowledge, it hardly holds true that the social network platform, such as Facebook, only adopts simply once JPEG compression, where the metabolic lossy channel probably remarkably impacts the robust steganography heavily relying on the transmission channel such as MCCI and TCM. Prior to embedding, MCCI or TCM need to learn the property of the transmission channel, while the learned property will be invalid as the channel continuously changes. On the contrary, DMCSS has no dependence on the channel and is immune to changes in the transmission channel, resulting in the best performance among the several methods. In fact, in our early observation (at the time both MCCI and TCM were published in 2018), Facebook indeed fixes the QF as 71. In 2019, the strategy of Facebook

was changed, in which original images with QF not larger than 85 are re-compressed with QF 71; otherwise, the re-compression is proceeded with various QFs (see [21]). However, in our recent observation (until the date of paper submission), Facebook completely breaks the prior rule of compression that continuously changes QF even though the images have the same original QF before uploading. Meanwhile, when the downloaded images from Facebook are uploaded again, the compression strategy is still unpredictable.

In addition, we conduct experiments on Twitter. When comparing the transmitted image via repeatedly uploading and downloading, it can be clearly observed that the image is post-processed by updating the strategy of compression. As Table 7 shows, TCM fails to complete the task of robust steganography. This is because the requirement of transmission channel cannot be met as multiple uploading and downloading. On the contrary, DMCSS without heavily relying on the channel performs best. Compared with the results on Facebook, DMCSS performance did not change, and the secret message could be extracted completely.

Table 6. The robustness performance on Facebook.

Method Results	DMCSS	MCCI	TCM	J-UNIWARD	UERD
R_c	50/50	0/50	0/50	0/50	0/50
R_{error}	0	0.3537	0.5013	0.4593	0.4600

Table 7. The robustness performance on Twitter.

Method Results	DMCSS	MCCI	TCM	J-UNIWARD	UERD
R_c	50/50	0/50	fail	0/50	0/50
R_{error}	0	0.4226	fail	0.4991	0.4993

Finally, it is proposed to conduct our experiments on WeChat, which is one of the most popular instant-messaging App in China. We consider two models for covert communication that is original model and non-original model. In the original model, the image is sent with selection of original image; in the non-original model, the image is directly sent with default selection. As Table 8 shows, among all compared methods, the secret bits can be perfectly extracted from all testing images in the original model. Unfortunately, as Table 9 illustrates, in the non-original model, most methods except DMCSS fail to transmit the stego images on WeChat. This is because WeChat non-maliciously attacks the images by using multiple post-processing manners except compression. However, this effect has no effect on the differential Manchester location feature, so DMCSS can still extract the secret message completely.

On the social network platform, the transmission channel that is characterized as black box behaves in the more complicated system than the channel in the simulated settings which only consider image compression attack. In virtue of our empirical analysis in this subsection, in the case that the channel is not completely acquainted, it is suggested to adopt the scheme relying on robust domain such as DMCSS.

Table 8. The robustness performance on WeChat (original image).

Method Results	DMCSS	MCCI	TCM	J-UNIWARD	UERD
R_c	50/50	50/50	50/50	50/50	50/50
R_{error}	0	0	0	0	0

Table 9. The robustness performance on WeChat (non-original image).

Method Results	DMCSS	MCCI	TCM	J-UNIWARD	UERD
R_c	50/50	0/50	0/50	0/50	0/50
R_{error}	0	0.4807	0.5008	0.4977	0.4979

5. Conclusions

In this paper, we proposed a novel robust steganography algorithm called DMCSS, by optimizing the Flipping the Sign of DCT Coefficients algorithm, which can resist a variety of JPEG compression attacks. Relying on the sign invariance of the DCT coefficients before and after JPEG compression, we successfully accomplish robust steganography even when the transmission channel has a compression attack. It is noteworthy that we comprehensively evaluate the effectiveness of the algorithm, indicating imperceptibility, capacity, undetectability, and robustness. Compared to the FSDC, we further improved the robustness and undetectability of DMCSS by differential Manchester code of the location features and optimization of the cost function. Just as we have learned, few literature works can be found in the current research on robustness for real social platforms. The robustness of stego images plays an important role in transferring steganography from the laboratory to the real world.

Compared with some algorithms, our proposed algorithm shows excellent robustness both in the clear channel and the dirty channel. At the same time, the cover image of DMCSS is resistant to both repeated compression and force compression, which helps the application of steganography on real social platforms. However, we have to acknowledge that the undetectability of stego images with high payloads is a limitation of our robust steganography. In future research, we need to improve the robust steganography algorithm in terms of undetectability.

Author Contributions: Conceptualization, S.W.; Methodology, S.W.; Visualization, S.W.; Writing—original draft, S.W.; Writing—review and editing, N.Z. and M.X. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Natural Science Foundation of China under Grant No. 61803135, the Key Research and Development Plan Project of Zhejiang Province under Grant No. 2017C01065.

Conflicts of Interest: The authors declare no conflict of interest.

Abbreviations

The following abbreviations are used in this manuscript:

LSBr	Least Significant Bit replacement
LSBm	Least Significant Bit matching
HUGO	Highly Undetectable steGo
WOW	Wavelet Obtained Weights
S-UNIWARD	Side Information UNiversal WAVElet Relative Distortion
FRAS	Feature Regions based Adaptive Steganography
DMAS	Dither modulation based Adaptive Steganography
DCRAS	Coefficients Relationship based Adaptive Steganography
JPEG	Joint Photographic Experts Group
QF	quality factor
TCM	Transmission Channel Matching
MCCI	Modifying Coefficients of Cover Image
DMCSS	Differential Manchester Code with Sign Steganography
CCPEV	PEV features enhanced by Cartesian Calibration
CCJRM	Cartesian Calibrated JPEG domain Rich Model
FSDC	Flipping the Sign of DCT Coefficients
STC	syndrome-trellis codes
PSNR	Peak Signal-to-Noise Ratio
OOB	out-of-bag

References

- Asikuzzaman, M.; Pickering, M.R. An overview of digital video watermarking. *IEEE Trans. Circuits Syst. Video Technol.* **2017**, *28*, 2131–2153. [\[CrossRef\]](#)
- Liu, T.Y.; Tsai, W.H. A new steganographic method for data hiding in microsoft word documents by a change tracking technique. *IEEE Trans. Inf. Forensics Secur.* **2007**, *2*, 24–30. [\[CrossRef\]](#)
- Li, B.; Tan, S.; Wang, M.; Huang, J. Investigation on cost assignment in spatial image steganography. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 1264–1277. [\[CrossRef\]](#)
- Huang, Y.; Liu, C.; Tang, S.; Bai, S. Steganography integration into a low-bit rate speech codec. *IEEE Trans. Inf. Forensics Secur.* **2012**, *7*, 1865–1875. [\[CrossRef\]](#)
- Xu, D.; Wang, R.; Shi, Y.Q. Data hiding in encrypted H. 264/AVC video streams by codeword substitution. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 596–606. [\[CrossRef\]](#)
- Pevný, T.; Filler, T.; Bas, P. Using high-dimensional image models to perform highly undetectable steganography. In *International Workshop on Information Hiding*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 161–177.
- Holub, V.; Fridrich, J. Designing steganographic distortion using directional filters. In *Proceedings of the 2012 IEEE International Workshop on Information Forensics and Security (WIFS)*, Tenerife, Spain, 2–5 December 2012; pp. 234–239.
- Li, B.; Wang, M.; Huang, J.; Li, X. A new cost function for spatial image steganography. In *Proceedings of the 2014 IEEE International Conference on Image Processing (ICIP)*, Paris, France, 27–30 October 2014; pp. 4206–4210.
- Holub, V.; Fridrich, J. Digital image steganography using universal distortion. In *Proceedings of the First, ACM Workshop on Information Hiding and Multimedia Security*, Montpellier, France, 17–19 June 2013; pp. 59–68.
- Guo, L.; Ni, J.; Shi, Y.Q. Uniform embedding for efficient JPEG steganography. *IEEE Trans. Inf. Forensics Secur.* **2014**, *9*, 814–825.
- Guo, L.; Ni, J.; Su, W.; Tang, C.; Shi, Y.Q. Using statistical image model for JPEG steganography: Uniform embedding revisited. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 2669–2680. [\[CrossRef\]](#)
- Denemark, T.; Fridrich, J. Improving steganographic security by synchronizing the selection channel. In *Proceedings of the 3rd ACM Workshop on Information Hiding and Multimedia Security*, Portland, OR, USA, 17–19 June 2015; pp. 5–14.
- Li, B.; Wang, M.; Li, X.; Tan, S.; Huang, J. A strategy of clustering modification directions in spatial image steganography. *IEEE Trans. Inf. Forensics Secur.* **2015**, *10*, 1905–1917.
- Zhang, W.; Zhang, Z.; Zhang, L.; Li, H.; Yu, N. Decomposing joint distortion for adaptive steganography. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *27*, 2274–2280. [\[CrossRef\]](#)
- Zhang, Y.; Luo, X.; Yang, C.; Ye, D.; Liu, F. A JPEG-compression resistant adaptive steganography based on relative relationship between DCT coefficients. In *Proceedings of the 2015 10th International Conference on Availability, Reliability and Security*, Toulouse, France, 24–27 August 2015; pp. 461–466.
- Zhang, Y.; Luo, X.; Yang, C.; Liu, F. Joint JPEG compression and detection resistant performance enhancement for adaptive steganography using feature regions selection. *Multimed. Tools Appl.* **2017**, *76*, 3649–3668. [\[CrossRef\]](#)
- Zhang, Y.; Qin, C.; Zhang, W.; Liu, F.; Luo, X. On the fault-tolerant performance for a class of robust image steganography. *Signal Process.* **2018**, *146*, 99–111. [\[CrossRef\]](#)

18. Tao, J.; Li, S.; Zhang, X.; Wang, Z. Towards robust image steganography. *IEEE Trans. Circuits Syst. Video Technol.* **2018**, *29*, 594–600. [[CrossRef](#)]
19. Zhao, Z.; Guan, Q.; Zhang, H.; Zhao, X. Improving the robustness of adaptive steganographic algorithms based on transport channel matching. *IEEE Trans. Inf. Forensics Secur.* **2018**, *14*, 1843–1856. [[CrossRef](#)]
20. Zhu, Z.; Zheng, N.; Qiao, T.; Xu, M. Robust Steganography by Modifying Sign of DCT Coefficients. *IEEE Access* **2019**, *7*, 168613–168628. [[CrossRef](#)]
21. Yu, X.; Chen, K.; Wang, Y.; Li, W.; Zhang, W.; Yu, N. Robust adaptive steganography based on generalized dither modulation and expanded embedding domain. *Signal Process.* **2020**, *168*, 107343. [[CrossRef](#)]
22. Li, F.; Wu, K.; Qin, C.; Lei, J. Anti-compression JPEG steganography over repetitive compression networks. *Signal Process.* **2020**, *170*, 107454. [[CrossRef](#)]
23. Tsai, J.S.; Huang, W.B.; Kuo, Y.H.; Horng, M.F. Joint robustness and security enhancement for feature-based image watermarking using invariant feature regions. *Signal Process.* **2012**, *92*, 1431–1445. [[CrossRef](#)]
24. Zhang, Y.; Zhu, X.; Qin, C.; Yang, C.; Luo, X. Dither modulation based adaptive steganography resisting jpeg compression and statistic detection. *Multimed. Tools Appl.* **2018**, *77*, 17913–17935. [[CrossRef](#)]
25. Tahmasbi, A.; Saki, F.; Shokouhi, S.B. An effective breast mass diagnosis system using zernike moments. In Proceedings of the 2010 17th Iranian Conference of Biomedical Engineering (ICBME), Isfahan, Iran, 3–4 November 2010; pp. 1–4.
26. Bas, P.; Filler, T.; Pevný, T. “Break our steganographic system”: The ins and outs of organizing BOSS. In *International Workshop on Information Hiding*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 59–70.
27. Filler, T.; Judas, J.; Fridrich, J. Minimizing additive distortion in steganography using syndrome-trellis codes. *IEEE Trans. Inf. Forensics Secur.* **2011**, *6*, 920–935. [[CrossRef](#)]
28. Pevný, T.; Fridrich, J. Merging Markov and DCT features for multi-class JPEG steganalysis. In Proceedings of the Electronic Imaging 2007, San Jose, CA, USA, 28 January–1 February 2007; p. 650503.
29. Kodovsky, J.; Fridrich, J. Steganalysis of JPEG images using rich models. In Proceedings of the IS&T/SPIE Electronic Imaging, Burlingame, CA, USA, 22–26 January 2012; p. 83030A.
30. Kodovsky, J.; Fridrich, J.; Holub, V. Ensemble classifiers for steganalysis of digital media. *IEEE Trans. Inf. Forensics Secur.* **2011**, *7*, 432–444. [[CrossRef](#)]