This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access

Robust Steganography by Modifying Sign of DCT Coefficients

ZHIQIANG ZHU[1], NING ZHENG[1], TONG QIAO[1,2], AND MING XU[1]
[1]School of Cyberspace, Hangzhou Dianzi University, Hangzhou 310018, China
[2]Zhengzhou Science and Technology Institute, Zhengzhou 450001, China

Corresponding author: Tong Qiao (e-mail: tong.qiao@hdu.edu.cn)

This work was funded by the Cyberspace Security Major Program in National Key Research and Development Plan of China under grant No. 2016YFB0800201, the Natural Science Foundation of China under grant No. 61572165, 61702150, and 61803135, the Public Research Project of Zhejiang Province under grant No. LGG19F020015, the Key Research and Development Plan Project of Zhejiang Province under grant No. 2017C01065.

ABSTRACT Modern adaptive image steganography with minimizing a distortion function has high performance of undetectability. However, when an image with hidden information is attacked by JPEG compression, its robustness cannot be guaranteed, that remarkably limits its extension from the lab to the real world. In this paper, a novel image steganographic algorithm is proposed that is robust to JPEG compression. First, by using the sign of DCT coefficients, that remains unchangeable before and after JPEG compression, we select the candidate coefficients for resisting JPEG compression. Second, the designed distortion function assigns cost for each candidate DCT coefficient. Finally, relying on both error correction code and Syndrome-Trellis Codes, an encoded message is embedded into the cover image with minimum embedding distortion. Compared with prior arts, extensive experimental results highlight both undetectability and robustness of our proposed algorithm.

INDEX TERMS Robust steganography, distortion function, JPEG compression, sign of DCT coefficients.

I. INTRODUCTION

STEGANOGRAPHY is an application of undetectable transmission of secret information in a carrier [1] [2]. Digital image is the most popular media on the Internet and the ideal carrier for hiding message, also the focus of multimedia forensics (see [3]–[9] for instance). Image steganography has made great progress in recent years[1]. However, images cannot always be lossless during transmission. In the practical scenario, such as being transmitted on social network, most of modern steganographic methods cannot correctly extract the secret message embedded in the image. Because images probably have been compressed on the server provider. Therefore, in this context, it is very meaningful to design a steganographic algorithm that can resist JPEG compression.

In the task of image steganography, secret bits are hidden into an image to avoid the eavesdropper's suspicion. In current community of image steganography, adaptive steganography has become a research priority due to its superior undetectability [10]. The most effective adaptive algorithm is based on the framework of minimizing embedding distortion, which defines the distortion as sum of embedding cost at each individual changed element. Then STCs (Syndrome-Trellis Codes) [11] adaptively embed the secret message into a cover image based on the embedding cost to minimize the total distortion.

To aim at improving the undetectability of current methodology, many current adaptive steganographic algorithms are proposed, such as spatial domain steganographic algorithms HUGO (Highly Undetectable steGo) [12], WOW (Wavelet Obtained Weights) [13], S-uniward (Spatial universal wavelet relative distortion) [14], HILL (HIgh-pass, Low-pass, and Low-pass) [15]; JPEG domain steganographic algorithms J-uniward (JPEG universal wavelet relative distortion) [14], UED (Uniform Embedding Distortion) [16]. The above algorithms attempt to embed the message in an imperceptible manner so that the stego image is similar to its corresponding cover image visually and statistically.

Meanwhile, in order to resist steganography, many steganalysis algorithms have been proposed, involving specific ones such as [17]–[19] and universal ones such as [20]–[23]. Relying on universal steganalytic features, the methods mainly model the inner relation of neighboring pixels in the image to determine whether the image is embedded with the secret message. For example, [20] merges two different DCT feature sets, and reduces the dimensions to obtain

[1]For simplicity, in this context, steganography refers to as hiding information in an image.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access

new and more effective features. SPAM (Subtractive Pixels Adjacency Model) [21] uses Markov chains to model different adjacent pixels, together with the sample probability transition matrix as the steganalytic feature. SRM (Spatial Rich Model) [22] calculates the co-occurrence matrix for four neighboring pixels with various sub-models. Due to that the excellent performance of the ensemble classifier [23] is combined with the powerful steganalytic features, the accuracy of steganalysis is further improved. Besides, the study of locating hidden bits has also been advanced [24]. Steganography and steganalysis are mutually antagonistic and reinforcing, both of which have been greatly developed.

However, to our knowledge, most of the current steganographic methods have poor performance of resisting JPEG compression. Therefore, it is proposed to investigate the design of robust steganography resisting JPEG compression. The contributions of this paper are as follows:

- To enrich the traditional framework of image steganography, we novelly propose the general practical application of robust steganography, and mainly analyze the distinguishable characteristics among robust steganography, adaptive (or traditional) steganography, and robust watermarking.
- It is proposed to establish the framework of the robust steganographic scheme using sign of DCT coefficients, that is capable of resisting JPEG compression attack.
- By designing the function of embedding cost, we select the cover elements from the texture regions to ensure the minimum distortion caused by hiding bits.
- Relying on the rule of STCs, together with the strategy of error correcting (prior to embedding), we further guarantee the undetectability and robustness of the proposed steganographic scheme.

The remainder of this paper is organized as follows. In Section II, the related works about current robust steganography are presented. In Section III, a typical image steganographic system is described, and the definition of robust steganography is addressed. In Section IV, the robust steganographic algorithm resisting JPEG compression is proposed. In Section V, the extensive experiments are provided. Concluding remarks are drawn in Section VI.

II. RELATED WORKS

Combining the strength of both adaptive image steganography (undetectability) and robust watermarking (compression resistance), the authors of [25] opened the new way of designing a robust steganographic scheme, involving embedding and extracting procedure.

In the embedding procedure, the technique of robust watermarking is used to determine the embedded domain, and select the cover elements. In virtue of some regular features, such as the location relationship existing among inter or intra DCT coefficient blocks, cover elements consisting of binary bits are acquired in the embedded domain. Note that when JPEG compression happens, the values of cover elements basically remain stable. Subsequently, a distortion function of adaptive steganography is used for assigning cost to each cover element. Meanwhile the secret message is encoded by an ECC (error correction code) such as RS (Reed-Solomon) to further improve its robustness. Next, relying on STCs encoding, one can embed the RS-coded secret message into the cover elements. Finally, a robust image with stego elements is obtained.

In the extracting procedure, referring to as the inverse process of embedding, the stego elements are first extracted from the robust image. Then STCs decoding is adopted to extract the secret message encoded by RS encoder. Finally, dependent of RS decoder, the secret message is decoded to obtain the original information.

However, not all robust watermarking is applicable to this framework. To our knowledge, when a watermarked image suffers potential JPEG compression attack, possibly only a fuzzy contour of embedded watermark is extracted, and meanwhile most of the watermark details disappear. In this context, one has to guarantee that the consistency of the cover elements before and after JPEG compression. Therefore, the selection of cover elements in the embedded domain directly determines the performance of robust steganography. In prior arts, the studies mainly focus on how to select the optimal technique of robust watermarking for improving the robustness of resisting JPEG compression.

In DCRAS (DCT Coefficients Relationship based Adaptive Steganography) [26], the embedded domain is constructed by using the relationship among DCT coefficients [27], which are not or less impacted by JPEG compression. The embedded domain is determined by the relationship between the coefficients in an 8×8 DCT block and the mean of the coefficients at the same position of three adjacent blocks. A cover element is extracted by comparing the magnitudes of the two coefficients. In FRAS (Feature Regions based Adaptive Steganography) [28], on the basis of the embedded domain constructed by DCRAS, the authors use the Harris-Laplacian feature (see [29], [30]) to construct the regions, striking the balance between the JPEG compression resistance and undetectability. In DMAS (Dither Modulation based Adaptive Steganography) [31], dependent of the quantization tables, the authors utilize DCT coefficients based dither modulation methods [32] to construct the embedded domain. Besides, the authors of [25] establish a burst error model based on the Poisson distribution, that reduces the fault tolerance performance of the above three methods. In addition, scaling attacks can be resisted by constructing embedded domain. The authors of [33] use the scaling invariance of the Zernike moment to resist scaling attacks. The embedded domain of this method is a set of Zernike moments, but its maximum size is limited, and only a few hundred bits can be embedded.

Recently, in order to tackle JPEG compression, based on the transmission channel matching, the authors of [34] repeatedly compress the image to reduce the impact caused by JPEG compression of the channel, that directly improves the robustness of the transmitted stego image. In [35], in

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access

virtue of the similar characteristic between stego image generation and the principle of JPEG compression, the cover image is transferred to an intermediate image, leading to that the channel compressed version of the intermediate image is same as the stego image. Although both methods can achieve high undetectability and robustness performance, the methods cannot be applied to unknown channels. The methods in [34] and [35] need to know the compression quality factor of the dirty channel before embedding. However, the proposed algorithm does not need to know the exact quality factor of the dirty channel. What is more, the stego images generated by methods in [34] and [35] cannot resist JPEG compression with multiple quality factors, which limits their wide application. However, the proposed algorithm can resist JPEG compression with multiple quality factors.

Without generality, most current arts (see [26], [28], [31]) mainly rely on the robust watermarking to establish the embedded domain, which indeed can achieve the good performance of resisting JPEG compression, but the performance of undetectability needs to be improved. In addition, prior algorithms mainly consider the case that a stego image with the hidden message is compressed with the same quality factor (as itself) rather than a different one. However, in practice, a stego image is probably compressed with different quality factors. To overcome the limitations of prior arts, based on the sign of DCT coefficients, we propose a novel embedding scheme for robust steganography, which improves the performance of both JPEG compression resistance and undetectability.

III. DESCRIPTION OF TYPICAL IMAGE STEGANOGRAPHIC SYSTEM

A typical image steganographic system in practice is illustrated in Figure 1. Steganography, in which two parties communicate in secret over a public channel, can be described as prisoners' problem [36]. Alice and Bob, two prisoners, intend to conspire together a plan for "prison break". The only communication channel that they can use, however, is unfortunately monitored by a warden Wendy. The fundamental problem of establishing a steganographic system is that the stego image with the hidden message (e.g. the plan for prison break), should be exchanged freely between Alice and Bob while Wendy can not observe any abnormal image in that channel. Additionally, Alice needs to design a secure steganographic mechanism involving embedding and extracting algorithm with a unique key, shared only with Bob, of course, not known by Wendy.

In prior arts, it always holds true that the transmitted image cannot suffer attacks from the public channel used by Alice and Bob. However, in this paper, it is proposed to challenge that assumption. Let us divide the public channel into clean and dirty one. The clean channel keeps the transmitted image untouched, that is only be monitored or attacked by Wendy. On the opposite, the dirty channel probably attacks the image, that is possibly not generated by Wendy, but from the server provider[2]. Consequently, it conducts some post-processing operations (such as JPEG compression, image re-sampling, noise adding, image filtering) for images transmitted over a dirty public channel. In fact, most of modern steganographic algorithms are prone to consider the transmission of a secret message in the clean channel. However, in the practical scenario, many public channels are dirty, in which the social network platform (Facebook and WeChat for instance) act as the server provider.

Facebook and WeChat are current popular social platforms with a large number of users (see [37], [38]). However, due to the limitation of storage, bandwidth and security, those social networks prefer conducting post-processing operations on uploaded images (see [34], [39]), leading to that a steganographer has to consider more factors in practical applications [40]. In our empirical experiment, it is observed that a JPEG image uploaded on Facebook is unavoidably twice-compressed with a fixed quality factor (QF), 71 for instance. Meanwhile, an image in other formats is twice-compressed with another fixed QF, 84 for instance. Besides, the server from WeChat probably performs more complicated post-processing operations, that remarkably impact the effectiveness of current non-robust steganography suitable for the clean channel.

In the dirty channel, to generate a stego image, Alice adopts the embedding algorithm (that is designed for the clean channel) to hide the secret message (shuffled by a secret key) in the cover image, and then uploads the stego image to her social network profile. After downloading the stego image, Bob uses the shared secret key and the extracting algorithm to obtain the secret message. Unfortunately, when Bob decodes the secret message, he cannot extract the original message, or possibly restore the misleading information. Because the image downloaded from the social network has been post-processed. Therefore, the plan of prison break is failed. More importantly, Bob even cannot figure out whether the stego image is maliciously modified by the active steganalyzer Wendy, or unavoidably attacked by the innocuous server provider. If the robust steganographic algorithm (that is designed for the dirty channel) with the ability of resisting the attack from the server provider is used, the two prisoners can immediately make a decision that the public channel for secret communication has been attacked by the warden Wendy. In addition, the server provider may also be malicious. When the server provider is not the safe party, our steganographic communication is likely to be detected by the server provider. The server provider may adopt different strategies to break down steganographic communication. In this case, we have to take corresponding measures to improve the robustness of the algorithm. Thus, it is of great practical significance to study the steganographic algorithm in the dirty channel. In virtue of the analysis, we address the following definition of robust

[2]In this context, we assume that the server provider is the safe party, that shows no interest of the two prisoners.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access

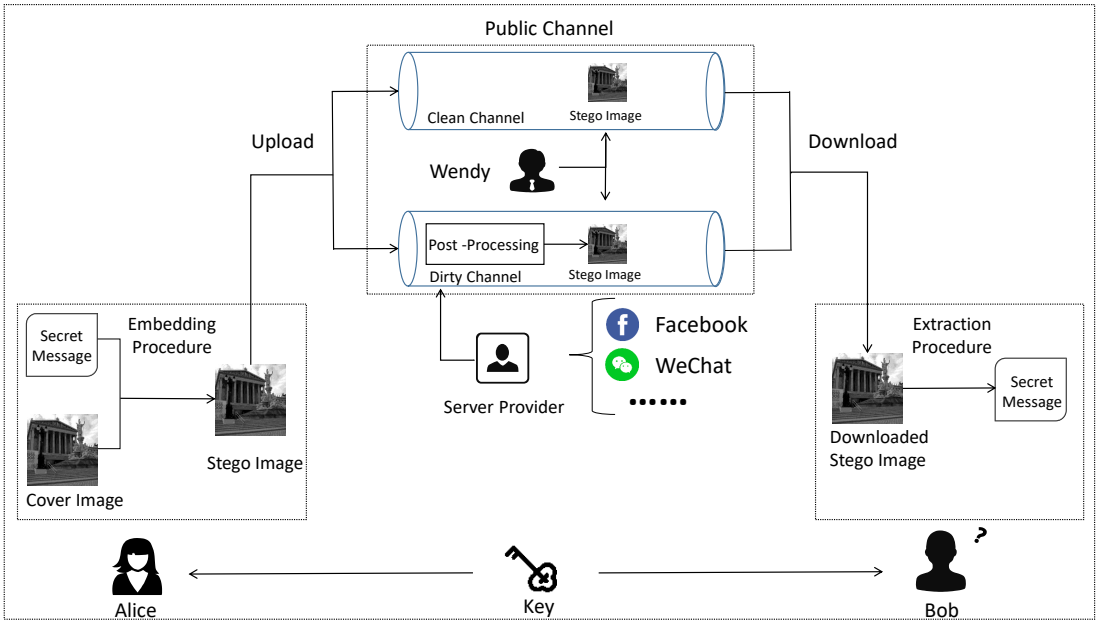


FIGURE 1. Illustration of typical image steganographic system in practice.

steganography:

Definition 1. *In a dirty channel (more practical public channel), to complete a task of secret communication, the goal of robust steganography is to fool a steganalyzer's detection on the premise of being able to resist post-processing attacks.*

In general, to achieve the goal of the proposed robust steganography, the designed algorithm has to satisfy four following requirements:

- Imperceptibility: the stego image is perceptually indistinguishable from the cover image.
- Undetectability: the stego image has the ability of escaping the detection from the reliable steganalysis.
- Capacity: the stego image carries the amount of the embedded data.
- Robustness: the stego image has the ability of preserving the embedded data when suffering various known or unknown post-processing attacks.

In the community of information hiding, even though both traditional steganography (in the clean channel) and robust watermarking have the similar requirements, they are kind of different from robust steganography (in the dirty channel). For clarity and simplicity, the similarities and differences of them are illustrated in Figure 2, where each requirement with dark color falling over the edge of the graph is assigned more weights than that closing to the center.

The most ideal scheme for information hiding should be superior to any other one in every aspect of performance. However, to our knowledge, it hardly holds true that the performance of imperceptibility, undetectability, capacity, and robustness is improved at the same time, meaning that

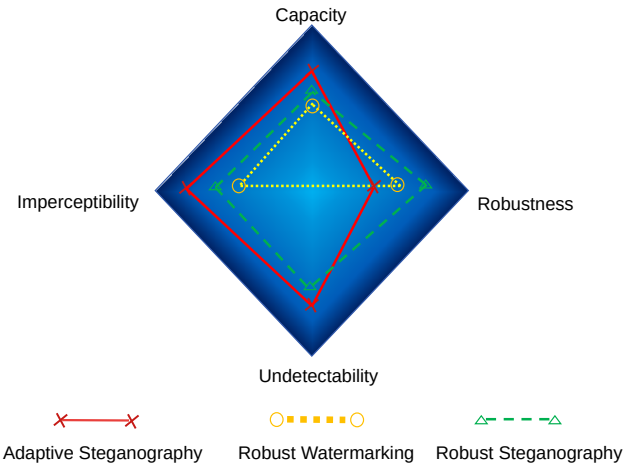


FIGURE 2. Illustration of requirements of three information hiding schemes: traditional steganography, robust watermarking, and robust steganography.

the increment of one side inevitably leads to the decrease of the other side. Hence, it is reasonable that one designs the different hiding schemes based on different requirements.

In steganography, the cover image serves as a bait, regardless of secret messages [1]. It is extremely important to ensure that no traces of hidden data are perceptual in the stego image, referring to the requirements of imperceptibility and undetectability. Furthermore, steganography as a means of secret communication needs to allow the transmission of large amounts of secret data. The prior-art traditional steganography, illustrated by the red solid line in Figure 2, mainly focuses on imperceptibility, undetectability and

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access

capacity, that are higher than that of robust watermarking. However, the traditional steganographic algorithm ignores that in the dirty channel, it has no ability to deal with various attacks, leading to its considerable lower robustness.

On the contrary, robust watermarking aims to protect copyrights of digital contents, mainly addressing robustness and imperceptibility, illustrated by the yellow triangle in Figure 2. It should be noted that the watermark (hidden data) usually carries additional information about the image content, such as the image owner, receiver or sender. Thus the existence of the watermark should be public. In this scenario, the robust watermarking algorithm does not require neither undetectability nor much capacity. When a watermarked image suffers a potential attack, possibly only the few bits of the extracted watermark is capable of testifying the ownership.

However, in the design of robust steganography, the extracted secret data from a stego image requires not only to be undetected, but also be perfectly correct to the receiver. Therefore, robust steganography should perform more robustly than robust watermarking. To strike the balances of different requirements, the proposed robust steganography combines the advantages of traditional adaptive steganography and robust watermarking, illustrated by the green dotted line in Figure 2, that is slightly to reduce undetectability and capacity, and appropriately enhance robustness.

In this paper, let us deal with the most important problem of robustness, referring to as JPEG compression resistance, that leads to the design of our proposed robust steganography. As a result, the secret message in the generated stego image can be correctly extracted not only in the clean channel, but also in the dirty channel.

IV. PROPOSED METHOD

Boldface symbols stand for matrices, vectors and sets. $\mathbf{x} = (x_1, x_2, ..., x_n)$ and $\mathbf{y} = (y_1, y_2, ..., y_n)$ stand for a cover image and a stego image. $\mathbf{c} = (c_1, c_2, ..., c_n)$ and $\mathbf{s} = (s_1, s_2, ..., s_n)$ stand for cover elements and stego elements.

For clarity, let us illustrate the framework of the proposed robust steganography in Figure 3. When a JPEG image is used for the robust steganography, the operations of Huffman decoding and inverse quantization help us acquire the original source data, that is quantized DCT coefficients. Next, we have to determine the embedded domain, that is constructed by the coefficients (or pixels), regions, or relative relationships which are not or less affected by JPEG compression. Since that the JPEG compression resistance is considered in our proposed scheme, let us define the embedded domain as the sign of DCT coefficients in this context. It should be noted that the cover elements, that remain relevantly stable before and after JPEG compression, are extracted in the embedded domain. Currently, the most successful approach for image steganography is content adaptive, meaning that the embedding strategy is designed based on the model of minimizing distortion between the cover and the its stego version, leading to the selection of cover elements. Next, the stego elements, that are actually another description of the secret message, are generated by using STCs encoding. Finally, the stego image is completed by using our designed embedding rule, where stego elements replace cover elements with flipping the sign of DCT coefficients. It should be noted that to further improve the robustness of the designed steganographic algorithm, the error correction code is adopted.

Embedding procedure:

1) Pre-process the cover image. Read a cover JPEG image and perform Huffman decoding in order to acquire the quantized DCT coefficients.
2) Extract cover elements. The cover elements are extracted in the embedded domain, which is constructed by the sign of DCT coefficients. To optimize the selection of cover elements, we re-compress the cover JPEG image for excluding the zero-value DCT coefficients.
3) Error correction encoding. Encode the message $\mathbf{m}$ with ECC (error correction code) to obtain the encoded message $\mathbf{m_e}$.
4) Calculate the embedding cost. The embedding cost measures the amount of distortion caused by changing a selected cover element. The embedding cost of cover elements is determined according to the characteristics of the embedded domain.
5) Embed the message based on STCs. Embed the encoded message $\mathbf{m_e}$ in the cover elements with STCs and obtain the stego elements $\mathbf{s}$.
6) Generate the stego image. Modify the embedded domain according to the stego elements $\mathbf{s}$ and generate the stego image $\mathbf{y}$.

Extraction procedure:

1) Pre-process the stego image. Huffman decoding and inverse quantization.
2) Extract stego elements. Relying on the shared key, the corresponding stego elements $\mathbf{s}$ are extracted from the embedded domain.
3) STCs decoding. Perform STCs decoding to extract the encoded message $\mathbf{m_e}$ from stego elements $\mathbf{s}$.
4) Error correction decoding. The encoded message $\mathbf{m_e}$ is decoded to get the corresponding secret message $\mathbf{m}$.

In this section, we give the overall framework of the proposed algorithm, and the algorithm is elaborated in details. First, we select the cover elements in the embedded domain. Second, the distortion function and STCs are introduced. Relying on the characteristics of the embedded domain, together with the distortion function, the embedding cost of the proposed algorithm is calculated. Next, the stego elements are obtained with the help of ECC and STCs encoding. Finally, the stego image is acquired by using the designed embedding rule.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access

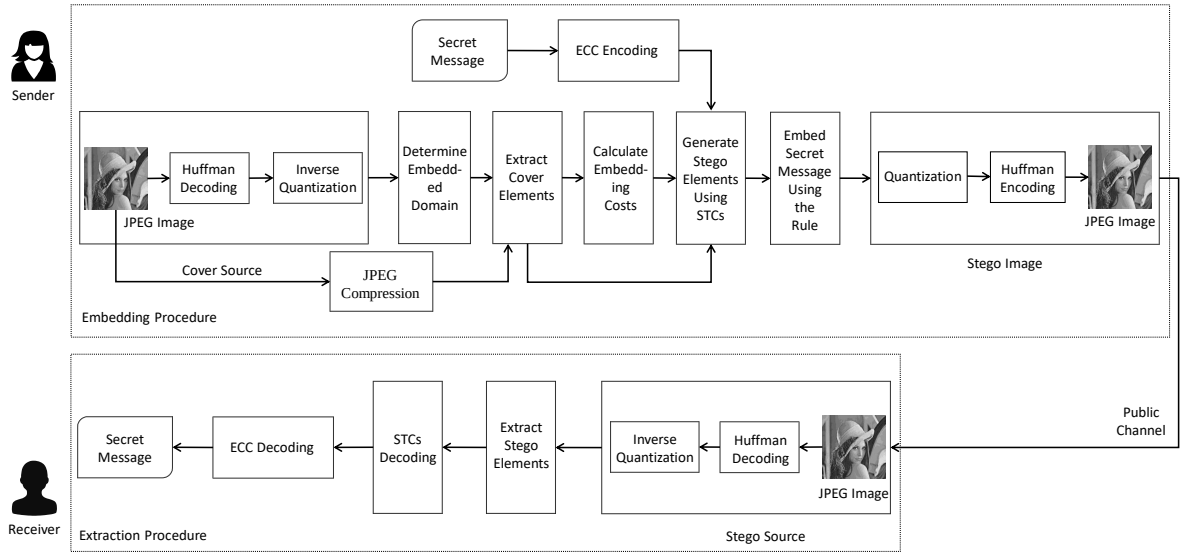


FIGURE 3. Illustration of our proposed robust steganographic framework.

A. SELECTION OF COVER ELEMENTS IN THE EMBEDDED DOMAIN

In this context, we propose a new algorithm to determine the embedded domain, that is the sign of DCT coefficients. Because most of the sign of DCT coefficients remain stable before and after JPEG compression. Besides, note that we exclude zero-value DCT coefficients. Immediately, a cover element $c_{i,g}$ can be formulated by:

$$c_{i,g} = \begin{cases} 1, & x_{i,g} > 0 \\ 0, & x_{i,g} < 0 \end{cases} \tag{1}$$

where g represents an index of a DCT block, and i represents the i-th position in the DCT block as zig-zag order. $x_{i,g}$ denotes a DCT coefficient of the i-th position in the g-th DCT block.

In fact, one cannot guarantee that each cover element in the embedded domain (that is the sign of the DCT coefficient) has good robustness. Possibly, some DCT coefficients become zero after JPEG compression, directly resulting in the invalidation of the corresponding cover elements. Thus, in the embedded domain, we further select the non-zero DCT coefficients after JPEG compression with a given quality factor (QF). In the following, let us optimize the scheme of selecting a cover element, that is formulated as:

$$c'_{i,g} = c_{i,g}, \quad \text{if} \quad x'_{i,g} \neq 0 \tag{2}$$

where $x'_{i,g}$ represents the DCT coefficient of the cover JPEG image after twice compression. $c'_{i,g}$ represents the cover element after optimization using Eq. (2). Note that the locations of cover elements need to be shared as a key.

In practice, as the quantization step increases, the amount of the cover elements $c'_{i,g}$ becomes small. Because the number of non-zero DCT coefficients gradually diminishes. Thus, the QF of JPEG compression impacts the total number of cover elements, that directly determines the capacity of the steganographic algorithm. In that scenario, with reducing the capacity for embedding, one can further improve the performance of JPEG compression resistance using our proposed algorithm.

B. INTRODUCTION TO DISTORTION FUNCTION

Modern image steganography mainly focuses on the design of the function with minimizing distortion caused by embedding [14]. The principle is that the distortion of the image can reach the minimum in utilization of the designed function, among which the uniward distortion function is expressed by the sum of relative changes of all wavelet coefficients with respect to the cover image:

$$D(\mathbf{x}, \mathbf{y}) \triangleq \sum_{k=1}^{3} \sum_{i=1}^{n} \frac{|W_i^{(k)}(\mathbf{x}) - |W_i^{(k)}(\mathbf{y})|}{\sigma + |W_i^{(k)}(\mathbf{x})|} \tag{3}$$

where $\mathbf{x}$ and $\mathbf{y}$ are a pair of cover and stego spatial images. n is the number of coefficients in the image. $W_i^k(\mathbf{x})$, $W_i^k(\mathbf{y})$, k = {1, 2, 3}, i ∈ {1,..., n} are from i-th wavelet coefficient in the k-th subband of the first decomposition level. $\sigma > 0$ is a constant stabilizing the numerical calculations. For J-uniward (a description of uniward in the JPEG domain), the distortion between DCT coefficient $\mathbf{x}$ and $\mathbf{y}$ requires decompressing the JPEG image into the spatial domain and calculating it, that can be expressed as follows:

$$D(\mathbf{x}, \mathbf{y}) \triangleq D(J^{-1}(\mathbf{x}), J^{-1}(\mathbf{y})) \tag{4}$$

where $J^{-1}(\mathbf{x})$, $J^{-1}(\mathbf{y})$ are the decompressed image of the JPEG file to the spatial domain.

The embedding distortion is computed as a sum of relative changes of coefficients in a directional wavelet filter bank decomposition. To minimize the distortion, embedding changes are encouraged in regions where all directions are

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access

complex. Thus, the more complex the region of the inquiry image is, the lower the cost is.

Furthermore, regardless of the interaction among adjacent coefficients, the total distortion can be described as the additive approximation using D to compute the cost ρ_i^{uni} of changing each x_i, that is formulated by:

$$D(\mathbf{x},\mathbf{y}) \triangleq \sum_{i=1}^{n} \rho_i^{uni}(\mathbf{x}, y_i) \tag{5}$$

where $\rho_i^{uni}(\mathbf{x}, y_i)$ denotes the cost of $\mathbf{x}$ with only its i-th changed element.

When the cover $\mathbf{x}$ is confirmed, the $D(\mathbf{x},\mathbf{y})$ can be simplified as $D(\mathbf{y})$. Assuming that the embedding algorithm changes $\mathbf{x}$ to $\mathbf{y} \in \mathcal{Y}$ with the probability $\pi(\mathbf{y}) = P(Y = \mathbf{y})$, the sender could send up to $H(\pi)$ bits on average with averaging the distortion $E_\pi(D)$ such that

$$H(\pi) = -\sum_{\mathbf{y}\in\mathcal{Y}} \pi(\mathbf{y}) \log \pi(\mathbf{y}) \tag{6}$$

$$E_\pi(D) = \sum_{\mathbf{y}\in\mathcal{Y}} \pi(\mathbf{y}) D(\mathbf{y}) \tag{7}$$

where $\mathbf{y} = (y_1, y_2, ..., y_n), \mathcal{Y} = I_1 \times I_2 \times \ldots \times I_n$ denotes Cartesian product, I_i represents the range of the embedding operation at element i, and $x_i \in I_i$. Here, a binary or ternary embedding operation is respectively denoted as $|I_i| = 2$ or $|I_i| = 3$. For example, in an 8-bit grayscale image, a binary operation $I_i = \{x_i, \bar{x}_i\}$, where the bar denotes the operation of flipping the LSB, means that 0 can only be flipped to 1; 1 only to 0. However, the ternary embedding operation is with $I_i = \{x_i - 1, x_i, x_i + 1\}$.

The sender wants to guarantee that the stego image has the minimum average distortion under the condition of fixed length L, that is expressed as:

$$\min_{\pi} \quad E_\pi(D) = \sum_{\mathbf{y}\in\mathcal{Y}} \pi(\mathbf{y}) D(\mathbf{y}) \tag{8}$$

$$\text{subject} \quad \text{to} \quad H(\pi) = -\sum_{\mathbf{y}\in\mathcal{Y}} \pi(\mathbf{y}) \log \pi(\mathbf{y}) = L \tag{9}$$

Fortunately, the STCs is designed to solve the above problem, that can approach to the theoretical rate-distortion bound. Based on the improvement of Standard-Trellis Coding, STCs minimizes the additive distortion. The secret message is embedded into the cover $\mathbf{x}$ by STCs to obtain the stego $\mathbf{y}$, which can be expressed as:

$$\mathbf{y} = Emb(\mathbf{x}, \mathbf{m}) = \arg\min_{\mathbf{y}\in C(\mathbf{m})} D(\mathbf{x}, \mathbf{y}) \tag{10}$$

where $C(\mathbf{m}) = \{z \in \{0,1\}^n | \mathbb{H}\mathbf{z} = \mathbf{m}\}$ is the coset corresponding to the syndrome $\mathbf{m}$, The parity-check matrix $\mathbb{H} \in \{0,1\}^{m\times n}$ of a binary Syndrome-Trellis Code of the length n and the codimension m is obtained by placing a small submatrix $\widehat{\mathbb{H}}$ of size $h \times w$ along the main diagonal [11]. The submatrix $\widehat{\mathbb{H}}$ acts as an input parameter, also shared between the sender and the receiver.

The distortion function (see Eq. (5)) is used to assign the cost to each DCT coefficient in the embedded domain, and meanwhile the STCs can adaptively minimize the distortion between $\mathbf{x}$ and $\mathbf{y}$ under the constraint of the distortion function. In the practical embedding, for clarity, it is proposed to use Eq. (14) to realize the calculation of Eq. (10). Due to the fact that the modern content-adaptive steganography with the minimal embedding distortion can obtain relatively high performance of undetectability, in our designed algorithm, the establishment of the embedding cost is based on Eq. (5).

C. CALCULATION OF EMBEDDING COST

To our knowledge, the STCs serves as the optimal choice of designing embedding schemes due to its fairly good performance. In our proposed robust steganography, the cover element is changed by flipping the sign of the corresponding DCT coefficient (see details in Sec. IV-A). In this case, the small absolute value of the DCT coefficient leads to the small embedding cost. In order to minimize the embedding cost, the cost of the texture region should be less than that of the smooth region, which maintains the high dimensional statistical model of cover image, and enhances the undetectability of the stego image. The cost of the small absolute value of the DCT coefficient is less than that of the large absolute value. In short, the small DCT coefficient of the texture region has the low embedding cost that is more suitable for our embedding.

One can first calculate the pre-cost ρ^{uni} for each changed DCT coefficient using the Eqs. (4) and (5). In fact, the pre-cost ρ^{uni} helps us select the texture region of an image. In our designed calculation of the embedding cost, if $\rho^{uni}(x_{i,g})$ is smaller than the threshold β, the cost of the DCT coefficient $x_{i,g}$ for changing is its absolute value. In this scenario, the smaller the absolute value of the DCT coefficient, the more suitable it is for embedding.

Therefore, the embedding cost of the DCT coefficient $x_{i,g}$ with respect to the modifying magnitude $\rho(x_{i,g}, m)$ can be defined by:
if m = 0

$$\rho(x_{i,g}, m) = \begin{cases} 0, & x_{i,g} < 0 \\ |x_{i,g}|, & 0 < x_{i,g} \le \alpha, \rho^{uni}(x_{i,g}) < \beta \\ |x_{i,g}|^\gamma, & x_{i,g} > \alpha, \rho^{uni}(x_{i,g}) < \beta \\ \rho^{uni}(x_{i,g}), & x_{i,g} > 0, \rho^{uni}(x_{i,g}) \ge \beta, \end{cases} \tag{11}$$

if m = 1

$$\rho(x_{i,g}, m) = \begin{cases} 0, & x_{i,g} > 0 \\ |x_{i,g}|, & -\alpha \le x_{i,g} < 0, \rho^{uni}(x_{i,g}) < \beta \\ |x_{i,g}|^\gamma, & x_{i,g} < -\alpha, \rho^{uni}(x_{i,g}) < \beta \\ \rho^{uni}(x_{i,g}), & x_{i,g} < 0, \rho^{uni}(x_{i,g}) \ge \beta \end{cases} \tag{12}$$

where β denotes the threshold used to select the DCT coefficients of the texture region, and α is used to control the size of the preferential embedding DCT coefficients. In practice, the proposed algorithm optimally selects the small

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access

DCT coefficients in the texture region for embedding. Note that when the DCT coefficient is greater than α, γ (generally larger than one) is used to increase the cost of the location while avoiding the undesirable embedding.

In the case of $m = 0$, when the DCT coefficient is negative, the embedding cost $\rho(x_{i,g}, m)$ equals to 0. Based on our embedding rule (see Sec. IV-E), the DCT coefficient dose not change even the embedding happens. Conversely, when the DCT coefficient is positive, its sign is flipped: if $\rho^{uni}(x_{i,g})$ is smaller than β, and the DCT coefficient $x_{i,g}$ is smaller than α, the embedding cost $\rho(x_{i,g}, m)$ equals to $|x_{i,g}|$; when $\rho^{uni}(x_{i,g})$ is smaller than β, and the DCT coefficient $x_{i,g}$ is larger than α, $\rho(x_{i,g}, m)$ equals to $|x_{i,g}|^{\gamma}$. if the other scenario happens, the embedding cost $\rho(x_{i,g}, m)$ is replaced by $\rho^{uni}(x_{i,g})$. Similarly, in the case of $m = 1$, the embedding cost can be expressed by Eq. (12).

D. ACQUISITION OF STEGO ELEMENTS

To further improve the self-correction ability of the secret message $\mathbf{m} = (m_1, m_2, ..., m_n)$, we can use the ECC to encode it. In order to obtain the encoded message $\mathbf{m}_e$, the formula is as follows:

$$\mathbf{m}_e = \mathrm{F}_{ECC}(\mathbf{m}) \tag{13}$$

where $\mathrm{F}_{ECC}(\cdot)$ stands for a ECC function, and $\mathbf{m}_e$ stands for the encoded message.

In our design of stego elements, the ECC can effectively correct error bits hidden in the stego image. We intend to adopt the binary BCH and RS, both of which are simple and easy to implement. For example, when the message consists of $t \times k$ bits, RS (n, k) can encode k symbols (each symbol contains t bits) to generate n symbols, which can correct $(n - k)/2$ error symbols. Besides, it should be noted that RS with stronger error correction ability is a type of non-binary BCH. With the help of the ECC encoding, the stego elements $\mathbf{s}$ can be obtained by the following formula:

$$\mathbf{s} = \mathrm{F}_{STCs}(\mathbf{c}', \rho(\mathbf{x}, \mathbf{m}_e), \mathbf{m}_e) \tag{14}$$

where $\mathbf{c}'$ denotes cover elements after optimization (see Eq. (2)). By using STCs encoding, $\mathbf{m}_e$ can be embedded into $\mathbf{c}'$ to generate the stego elements $\mathbf{s}$ under the "guidance" of the embedding cost ρ.

Note that the ECC plays an important role in the process of designing robust steganography. In fact, when the stego image is attacked by JPEG compression, the DCT sign of the stego image generated by using our proposed algorithm basically remains unchanged. However, a small portion of the DCT coefficients (with the large quantization step) possibly becomes zero. In this scenario, ECC can correct those error bits to some extent.

E. DESIGN OF EMBEDDING RULE

After extracting the cover elements, STCs is used to embed the binary encoded secret message into the cover elements to obtain the stego elements. Let us modify the DCT coefficients to replace the cover elements $\mathbf{c}'$ with the stego elements $\mathbf{s}$, and generate a stego image $\mathbf{y}$. Immediately, the embedding rule is formulated as:

$$y_{i,g} = \begin{cases} x_{i,g}, & c'_{i,g} = s_{i,g} \\ -x_{i,g}, & c'_{i,g} \neq s_{i,g} \end{cases} \tag{15}$$

where $c'_{i,g}$ represents the cover element of the i-th position in the g-th DCT block. $s_{i,g}$ represents the stego element. $x_{i,g}$ represents the DCT coefficient used for extracting the cover element in the cover image, and $y_{i,g}$ from the stego image. When $c'_{i,g} = s_{i,g}$, $y_{i,g} = x_{i,g}$, where the DCT coefficients $x_{i,g}$ remains unchanged; when $c'_{i,g} \neq s_{i,g}$, $y_{i,g} = -x_{i,g}$, where the sign of $x_{i,g}$ is changed. Finally, the stego image is obtained.

V. EXPERIMENTAL RESULTS AND ANALYSIS

In this section, it is proposed to verify the performance of the proposed algorithm, involving capacity, imperceptivity, undetectability, and robustness. The randomly-generated binary bitstream, serving as the secret message, is used for embedding. With different payloads, we generate the stego images from the given cover images to test the effectiveness of our proposed robust steganography. Specifically, the extensive experiments include: 1) determination of embedding cost parameters; 2) evaluation of capacity; 3) evaluation of imperceptibility performance; 4) evaluation of undetectability performance; 5) evaluation of robustness performance; 6) comparison with prior arts.

A. EXPERIMENTAL SETUPS

We generate the stego images using the payload from 0.01 to 0.1 bpnzAC (bits per non-zero AC coefficient), and the quality factor from 65 to 95, which are implemented in Matlab R2017a. Note that the original images are acquired from Bossbase1.01 [41], that is a benchmark dataset in the community of multimedia security. The benchmark J-uniward algorithm is a very classic and mostly used steganography algorithm in the JPEG domain. Therefore, the proposed algorithm is compared with the J-uniward algorithm. In addition, the proposed algorithm is also compared with some other robust steganography algorithms, such as DCRAS [26], FRAS [28], DMAS [31], algorithm in [34] and algorithm in [35]. Table 1 lists the experimental image and algorithm statistic.

B. DETERMINATION OF EMBEDDING COST PARAMETERS

In our experiments, the payloads, ranging from 0.01 to 0.1, decides the amount of the used DCT coefficients. In general, the absolute value of DCT coefficients, not larger than 10, are available for embedding, leading to the empirical parameter $\alpha = 10$. In this scenario, the embedding cost equals to the absolute value of the DCT coefficient itself (see Eq. (11)). Without of generality, we also consider another scenario. When the DCT coefficients,

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access

TABLE 1. Experimental statistic.

Image source	BOSSbase 1.01 dataset [41]
Image color	Grayscale
Image size	512×512
Image format	JPEG
Quality factor	65, 75, 85, 95
Number of original images	10000
Payload	$0.01 \sim 0.1$ bpnzac
Steganography	Our proposed algorithm, J-uniward [14], DCRAS [26], FRAS [28], DMAS [31], algorithm in [34] and algorithm in [35]
Steganalysis	Ensemble classifier with CCPEV [20] and CCJRM features [42]

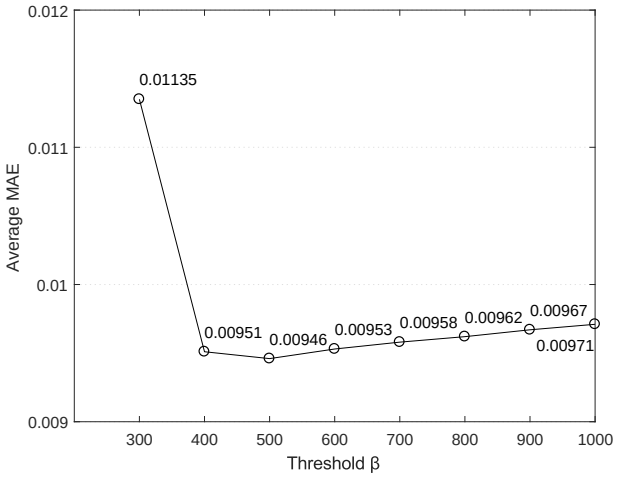


FIGURE 4. The average MAE of the feature set between cover and stego images with using different thresholds β.

whose absolute value are larger than α, are required for embedding, the exponential parameter $\gamma > 1$ is used to amplify the embedding cost of those DCT coefficients in the texture region, in which $\gamma = 3$ is our optimal choice based on the extensive empirical experiments.

In order to determine the threshold β, 2000 cover images with QF 75 are randomly selected. Relying on the proposed algorithm, the threshold varying from 300 to 1000 are utilized to generate the stego images with 0.1 payloads. For each inquiry image including a cover one and its stego version, let us calculate the 548-dimensional CCPEV steganalytic feature vector. It is proposed to adopt MAE (Mean Absolute Error) to measure the distance of the feature set between cover and stego images. The smaller the distance is, the more difficult the stego image is to be detected. The average MAE of 2000 images with different thresholds are illustrated in Figure 4. When the threshold equals to 300, the average MAE reaches to 0.01135, about 0.002 higher than the others; when the threshold is greater than 500, the average MAE remains stable with only a slight ascending trend. Thus, in the design of the robust steganography, the threshold $\beta = 500$ is our optimal choice.

C. CAPACITY PERFORMANCE

In this section, the embedding capacity of the proposed algorithm is evaluated. Capacity refers to the maximum number of secret bits that a steganographic algorithm can embed in the cover image. In the J-uniward algorithm, the capacity is the number of non-zero AC coefficients of the cover image. In the proposed method, the capacity is the number of cover elements. In order to improve the robustness of the proposed algorithm, the cover elements are extracted from the non-zero AC coefficients of the compressed cover image. Therefore, the number of cover elements is the same as the number of non-zero AC coefficients of cover image compressed with a given QF (see Eq. (2)).

First, all the 10000 images in the database are used to generate cover images with QF = {85, 95}. Then all cover images are compressed with given QF 60 to extract cover elements. Finally, we calculate the capacity of the proposed algorithm and the J-uniward algorithm in each cover image. Table 2 shows the average capacity of the proposed algorithm and J-uniward algorithm.

TABLE 2. Average capacity of the proposed algorithm and J-uniward algorithm [14].

Embedding algorithm \ Quality factor	85	95
Proposed algorithm	28748	31525
J-uniward [14]	56235	98095

As Table 2 reports, with increasing the quality factor of the cover image, the capacity of the J-uniward gradually increases. When the quality factor of cover image is 95, the J-uniward algorithm has the highest average capacity. Similarly, with increasing the quality factor of the cover image, the capacity of the proposed algorithm gradually increases. When the quality factor of cover image is 95, the proposed algorithm has the relatively large capacity, which is about 1/3 that of the J-uniward algorithm. When cover images with quality factor 85 and 95 are compressed with QF 60, the number of non-zero AC coefficients of images with quality factor 95 is larger, more non-zero AC coefficients are retained after compression, resulting in greater embedding capacity of images with quality factor 95. It is worth noting that we can change the capacity by adjusting the size of the given QF to meet different embedding requirements.

D. IMPERCEPTIBILITY PERFORMANCE

To avoid any suspicion caused by steganography, one has to guarantee the imperceptibility of a stego image. Here, taking Lena, Barbara, Goldhill, and Peppers images for example, we intend to validate the imperceptibility of the acquired image using our proposed robust steganographic algorithm.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access

The stego images are obtained by embedding secret bits with 0.1 payloads. Furthermore, to compare with the prior-art J-uniward algorithm, the PSNR (Peak Signal to Noise Ratio) and SSIM (Structural SIMilarity index) serve as the evaluation metric.

TABLE 3. PSNR comparison of 4 images obtained respectively by using the proposed scheme and J-uniward algorithm [14].

Images \ Embedding algorithm	Proposed scheme	J-uniward [14]
Barbara	42.4368	53.3915
Goldhill	43.9344	55.0221
Lena	46.8689	56.3775
Peppers	46.8496	56.1087

TABLE 4. SSIM comparison of 4 images obtained respectively by using the proposed scheme and J-uniward algorithm [14].

Images \ Embedding algorithm	Proposed scheme	J-uniward [14]
Barbara	0.9974	0.9997
Goldhill	0.9974	0.9997
Lena	0.9977	0.9998
Peppers	0.9975	0.9998

As Table 3 reports, the PSNR values generated by the J-uniward algorithm are basically at the interval [53, 56]. While the images generated by the proposed algorithm are at the interval [42, 47]. Since the J-uniward algorithm slightly changes the DCT coefficient, that is far smaller than the change of the DCT coefficient caused by sign flipping, the stego image using the proposed algorithm cannot acquire high PSNR as J-uniward. Nevertheless, the PSNR of the stego image using our robust steganography is still acceptable, whose imperceptibility performance is also very good. In addition, Table 4 illustrates that the SSIM values of the stego images generated by the proposed algorithm are above 0.997, which are very close to the SSIM values generated by the J-uniward algorithm. The closer the SSIM value is to 1, the better the imperceptibility of the algorithm. The results of Table 4 further verify that our proposed algorithm has the strong imperceptibility, comparable to that of the prior art. Moreover, Figure 5 illustrates the exemplary images using two different algorithms, where Barbara, Goldhill, Lena, and Peppers images are displayed from left to right. Whatever embedding scheme is adopted, it is observed that the stego image is visually indistinguishable from its corresponding cover one.

E. UNDETECTABILITY PERFORMANCE

In this section, we conduct a modern steganalysis detector , an ensemble classifier [23] with the CCPEV [20] or CCJRM [42] features, to verify the undetectability of our proposed robust steganography, also its comparison with prior arts such as DCRAS [26], FRAS [28], and DMAS [31]. 2000 stego images are generated by using 2000 original images randomly selected from the BOSSbase dataset.

(a) Cover images

(b) Stego images generated by the J-uniward algorithm

(c) Stego images generated by the proposed algorithm

FIGURE 5. Stego images results generated by the proposed algorithm and J-uniward algorithm.

The half of the stego and cover images are used for training the ensemble classifier, the remaining images are used for testing. The undetectability of the proposed algorithm is quantified using the ensemble's "out-of-bag" (OOB) error E_{OOB}, which is an unbiased estimate of the testing error "average" over multiple bootstrap samples of the image source during training.

First, compared to the prior-art J-uniward algorithm, let us evaluate the undetectability of our algorithm with or without ECC. 0.05 payloads are used for embedding. BCH (15,11,1) and RS (31,15) (see details in Sec. IV-D), respectively used for ECC encoding. Cover images with QF = {75, 85, 95} are used to generate stego images. Then the detection errors E_{OOB} of stego images are shown in Table 5.

When the quality factor is fixed, the J-uniward algorithm performs better than our proposed schemes, especially dealing with the scenario of high QFs such as 85 and 95. Because the J-uniward algorithm only slightly modifies DCT coefficients. Besides, with increasing the quality factor, the undetectability of the J-uniward algorithm is further improved. In fact, with increasing the quality factor, the redundancy of the compressed image is increased. Meanwhile, the high correlation among pixels of an inquiry image unavoidably disturbs the differences caused by embedding.

When the JPEG images with the quality factor 75 or 85, for our proposed robust steganography, RS encoding slightly improves its undetectability while BCH encoding slightly degrades its performance. Unfortunately, when the quality factor 95 is used for testing, any proposed scheme cannot guarantee the undetectability. Note that the secret bits are embedded by modifying the sign of DCT coefficients. Therefore, with increasing QF, the larger DCT coefficient and the incremental amount of modified elements directly result in high detection probability. Nevertheless, it is worthwhile to improve the robustness of steganography at the cost of to some extent undetectability performance. In the following experiments, it is proposed to adopt JPEG cover images with QF 75.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access

TABLE 5. Illustration of the detection error E_{OOB} with different quality factors.

Embedding algorithm / Quality factor	Proposed scheme without ECC	Proposed scheme with BCH	Proposed scheme with RS	J-uniward [14]
75	0.3385	0.3315	0.3401	0.4981
85	0.0739	0.0669	0.0772	0.4994
95	0.0034	0.0024	0.0029	0.4996
Average	0.1386	0.1336	0.1401	0.4990

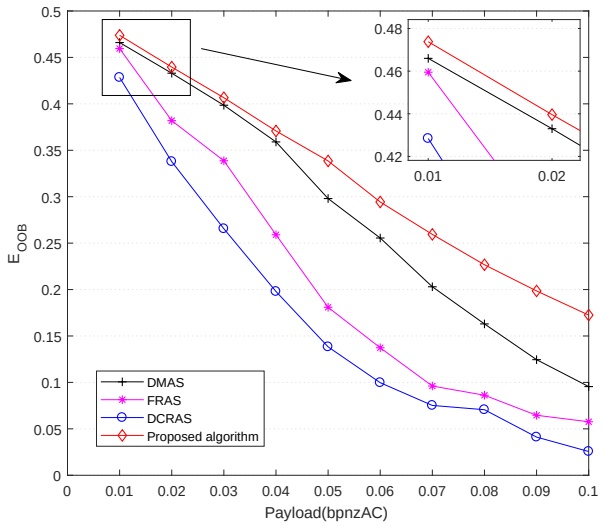


FIGURE 6. Illustration of the detection error E_{OOB} of the proposed algorithm, DCRAS, FRAS and DMAS.

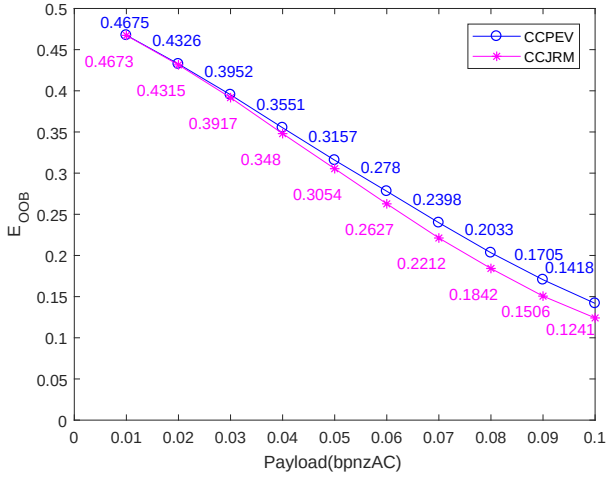


FIGURE 7. Illustration of the detection error E_{OOB} when adopting two different detectors.

To compare with the prior-art steganographic algorithms, we generate JPEG images with QF 75 using the payload ranging from 0.01 to 0.1. As Figure 6 illustrates, the undetectability performance of all algorithms is gradually improved with decreasing the payload. When the payload is lower than 0.04, the detection error E_{OOB} of the proposed algorithm is marginally better than DMAS, and remarkably better than the other two robust steganographic schemes. As the payloads continuously increases, the undetectability differences between ours and the others are further enlarged.

In order to further verify the undetectability of the proposed method, we compare two popular feature sets, referring to as CCPEV and CCJRM, to establish an ensemble classifier for evaluating our proposed robust steganography. All the 10000 images in the database are used to generate cover images with QF 75. Next, the corresponding stego images with different payloads are generated using the proposed method. We randomly select 5000 images to train an ensemble classifier while the rest is used for testing. Figure 7 reports the E_{OOB} of the proposed algorithm for both CCPEV and CCJRM feature-based detector. It can be observed that in the case of low payload, the E_{OOB} of the proposed algorithm against two detectors basically remains the same. However, with increasing the payload (larger than 0.05), the undetectability of the algorithm against the CCPEV detector is slightly better than the CCJRM detector. It is worth noting that compared with the results of Figure 6, as the number of training sets increases, the undetectability of the proposed algorithm against the CCPEV detector is marginally decreased, nevertheless still acceptable.

F. ROBUSTNESS PERFORMANCE

In this section, it is proposed to verify the robustness performance of the proposed algorithm. Let us define the extraction error rate as $R_{error} = \frac{N_{error}}{N_m}$, where the N_{error} denotes the number of error bits, and N_m is the length of the secret message, that is the total number of the embedded bits (see Figures 9, 10, 11 and Table 6). It is worth noting that the embedded secret message is a pseudo-random bit sequence. In the practical application, we can embed a semantical text for instance. One character consists of eight bits. In the worst case, each error bit represents one character error, the extraction rate of the character error is 8 times that of the bit error extraction rate. In the optimal case, the eight bits from one character are incorrectly extracted at the same time. The extraction rate of the character error is the same as the bit error extraction rate. Also, we count the total number of the stego images from which the secret message can be perfectly extracted (see Figure 8).

All the 10000 cover images in the database are used for embedding. Here, our proposed three schemes are compared, referring to as the scheme without ECC, the scheme

with BCH (15,11,1), and the scheme with RS (31,15). We select different payloads ranging from 0.01 to 0.1 to generate stego images. It should be noted that each stego image with QF 75 is twice-compressed with QF = {65, 75, 85}, that simulates the JPEG compression attack in the dirty channel. In this scenario, let us use the metric R_{error} to evaluate if the secret message can be perfectly extracted.

Figure 8 demonstrates the extraction results of the secret bits from the stego images. Obviously, it can be observed that the scheme with BCH or RS performs much better than the scheme without ECC. With decreasing payload, the robustness performance is gradually improved. In particular, when the payload is not larger than 0.04, if the scheme with RS is adopted, the secret bits from the stego image can be extracted better, even the stego images are attacked by JPEG compression with QF = {75, 85}. Besides, note that with decreasing QF, the robustness performance is unavoidably degraded because high compression rate brings more zero-value DCT coefficients in which the secret bits are probably hidden. Furthermore, as the number of zero-value coefficients is further increased, as well as the number of the error bits, the ECC probably becomes invalid. In this scenario, the robustness of our proposed algorithm is remarkably degraded.

Based on our analysis, the proposed algorithm scheme with RS (31,15) can exert its utmost ability to ensure the integrity of the embedded secret message, which is undoubtedly our optimal choice for robust steganography. Besides, it should be noted that the scheme with RS (31,15) can correct more images than the scheme with BCH (15,11,1), empirically indicating that the stronger the error correction ability is, the more robustly the proposed algorithm performs.

Next, to further evaluate the robustness performance, both our proposed algorithm and the J-uniward steganographic algorithm are compared by using the metric R_{error}. Similarly, 10000 stego images, with different payloads ranging from 0.01 to 0.1, are generated using QF 75, and decompressed using QF = {65, 75, 85}. The average R_{error} is illustrated in Figures 9, 10 and 11.

As Figure 9 displays, the J-uniward algorithm basically invalids in the case that the generated stego images suffers JPEG compression attack in the dirty channel. When stego images are compressed with QF 75, due to the same QF for double compression, the DCT coefficients nearly remain unchanged, leading to the lower average R_{error} than that of the other cases. And the average R_{error} of the J-uniward algorithm is not lower than 10.04% with all given payloads. As the payload increases, the average R_{error} of the proposed algorithm has slightly increased trend. However, the average R_{error} is not higher than 0.26%. Although the undetectability performance of the J-uniward algorithm is stronger than the proposed algorithm, it is very vulnerable to JPEG compression and cannot be applied in the dirty channel, while the proposed algorithm has high robustness performance under the premise of undetectability.

As Figure 10 reports, if the algorithms with BCH are adopted, the average R_{error} of both the J-uniward algorithm and our algorithm is decreased. When stego images are attacked by JPEG compression with QF equal to 65 or 85, with the help of BCH, the average R_{error} of the J-uniward algorithm decreases from 50% to around 36%. Although the robustness performance of J-uniward algorithm been improved, its average extraction error rate is still too high to be applied in the dirty channel. However, in the face of JPEG attack with QF = {65, 75, 85}, the average R_{error} of our proposed algorithm is not higher than 0.3%. In fact, if we reduce the message length or use the more efficient ECC, such as RS, we intend to further improve the robustness performance of the proposed algorithm.

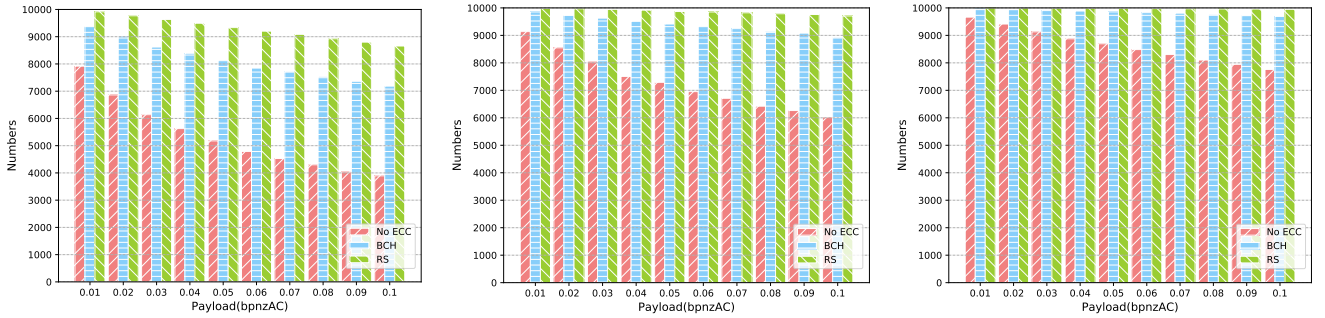
Unfortunately, when the stego images are attacked by JPEG compression with QF equal to 65 or 85, RS cannot help the J-uniward algorithm to decrease its R_{error} (See Figure 11). RS is a non-binary ECC, that has the strong ability to correct burst error[3]. When stego images are attacked, the error bits are more dispersed in multiple symbols, which is beyond the error-correction ability of RS. For RS (31,15) as an example, each 5-bit data constitutes a decimal symbol, which can correct up to 8 symbol errors. When a stego image generated by the J-uniward algorithm is attacked by JPEG compression, most errors are more than 8 symbols. Even though the J-uniward algorithm with RS is used, the failure of error correction still happens. On the contrary, when the proposed scheme without ECC is adopted, the extracted bits have only a few and dispersed errors. Thus, the proposed scheme with RS is capable of correcting those error symbols.

Next, the robustness performance of our scheme with RS, algorithm in [34] and algorithm in [35] is compared. Algorithms in [34] and [35] use QF 75 as quality factors of channel. Table 6 shows the R_{error} of different algorithms resisting JPEG compression with QF = {65, 75, 85}. When stego images are attacked by JPEG compression with QF 75, all three algorithms have low R_{error}, and the R_{error} of the proposed methods is not higher than 0.41×10^{-3}, which is much lower than the R_{error} of algorithm in [34]. When the stego images are attacked by JPEG compression with QF 65 or 85, algorithms in [34] and [35] are basically invalid, while the proposed algorithm still has outstanding robustness performance. It is worth noting that the AVG (average value) of our scheme with RS is not higher than 1.04×10^{-3}, which is much lower than the other two robust algorithms. In addition, Figure 12 intuitively shows that algorithms in [34] and [35] can not resist JPEG compression with multiple quality factors. On the contrary, the proposed algorithm has great advantages in resisting multiple JPEG compression.

In [34], the authors define the rate of extraction error $R_e^{'}$ = n_{error} / n_m, where the n_{error} denotes the number of

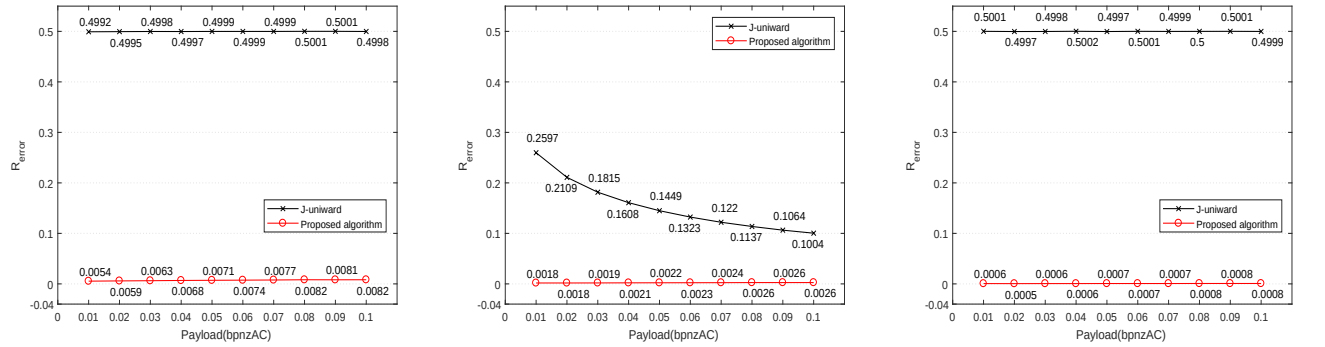
[3]A burst error is a string of corrupt data, measured as the length between the first and last error signals.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access



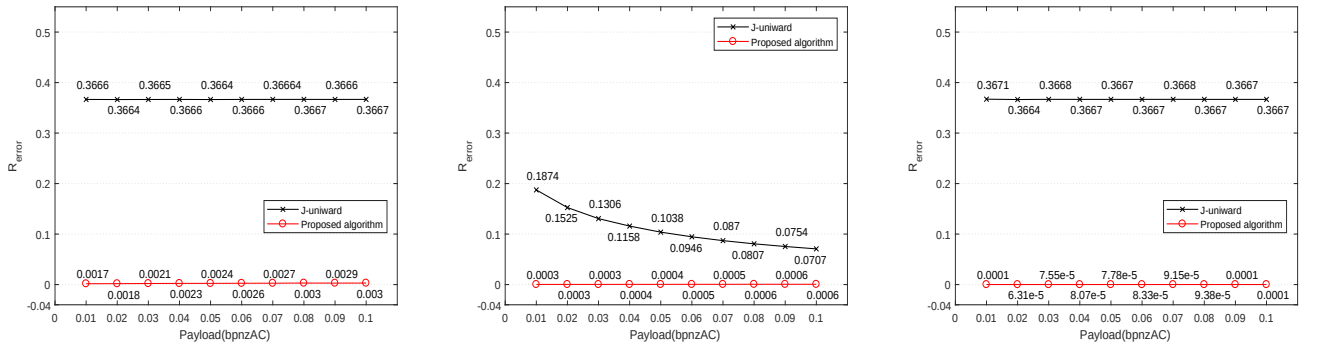
(a) QF = 65 (b) QF = 75 (c) QF = 85

FIGURE 8. Illustration of the total number of stego images, where the secret bits can be correctly extracted.



(a) QF = 65 (b) QF = 75 (c) QF = 85

FIGURE 9. R_{error} of two algorithms without ECC resisting JPEG compression.



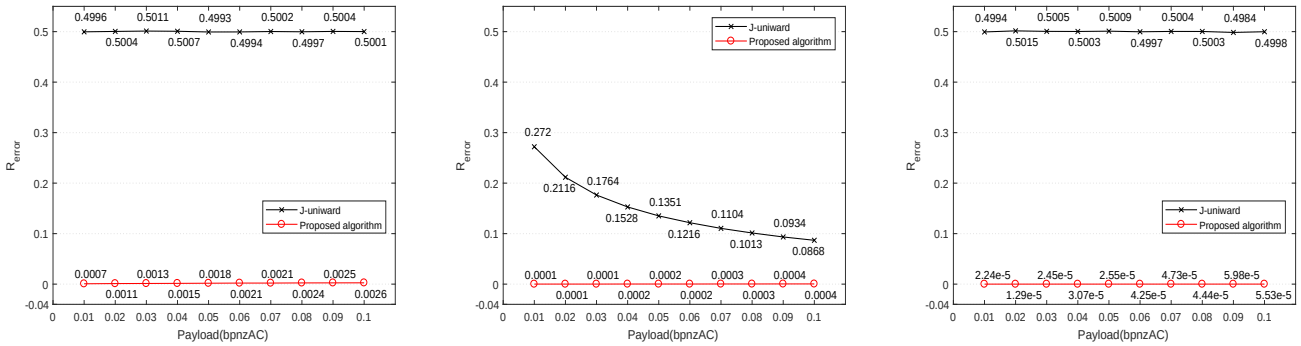
(a) QF = 65 (b) QF = 75 (c) QF = 85

FIGURE 10. R_{error} of two algorithms with BCH resisting JPEG compression.

error bits, and n_m is the length of the embedded message. We use the same criteria to compare the robustness of the proposed method with algorithm in [34]. All the 10000 images in the database are used to generate cover images with QF 75. Next, the corresponding stego images with different payloads are generated using our scheme with RS and algorithm in [34]. Figure 13 shows the $R_e^{'}$ of different algorithms resisting JPEG compression with QF 75.

As Figure 13 illustrates, the robustness performance of both algorithms is gradually improved with increasing the payload. When the payload arrives at around 0.03, the difference of $R_e^{'}$ between the two algorithms is close to zero. When the payload is not lower than 0.03, the $R_e^{'}$ of proposed algorithm is lower than the algorithm in [34], and the $R_e^{'}$ of proposed algorithm decreases sharply than the algorithm in [34]. When the payload is 0.1, the $R_e^{'}$ of the algorithm in [34] maintains at about 2%. However, the $R_e^{'}$ of the proposed algorithm is 1.43% and still has a significant

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access



(a) QF = 65 (b) QF = 75 (c) QF = 85

FIGURE 11. R_{error} of two algorithms with RS resisting JPEG compression.

TABLE 6. R_{error} of our scheme with RS, algorithm in [34] and algorithm in [35]($\times 10^{-3}$).

Embedding algorithm	QF	Payloads									
		0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.1
algorithm in [34]	65	501	499	500	502	501	498	498	500	497	503
	75	11.9	12.2	12.6	12.7	12.8	12.5	12.3	12.4	12.2	12.3
	85	498	499	500	497	501	501	503	499	499	500
	AVG	337	336	337	337	338	337	337	337	336	338
algorithm in [35]	65	499	497	502	499	501	498	498	500	501	500
	75	0	0	0	0	0	0	0	0	0	0
	85	500	499	497	498	501	500	499	502	499	499
	AVG	333	332	333	332	334	332	332	334	333	333
Our scheme with RS	65	0.78	1.11	1.34	1.53	1.89	2.05	2.18	2.43	2.49	2.66
	75	0.15	0.15	0.17	0.21	0.28	0.28	0.31	0.38	0.40	0.41
	85	0.02	0.01	0.02	0.03	0.02	0.04	0.04	0.04	0.05	0.05
	AVG	0.32	0.42	0.51	0.59	0.73	0.79	0.95	0.95	0.98	1.04

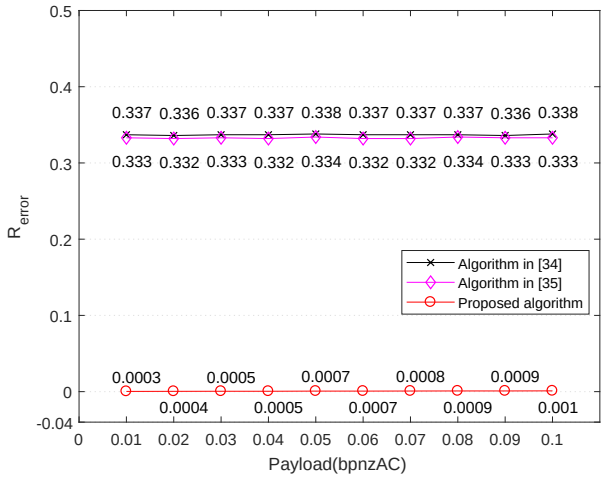


FIGURE 12. The AVG R_{error} of our scheme with RS, algorithm in [34] and algorithm in [35].

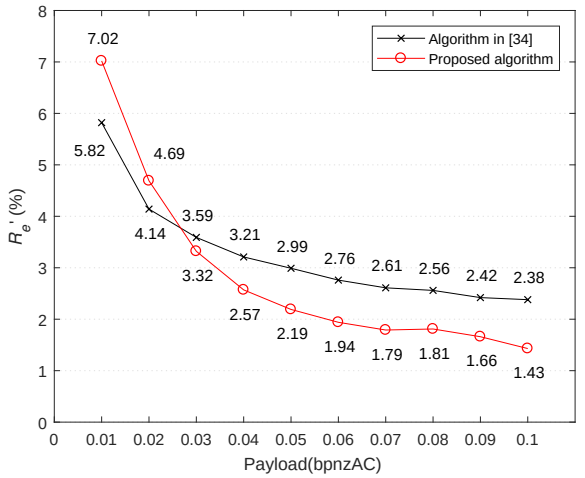


FIGURE 13. R_e' of our scheme with RS and algorithm in [34].

downward trend.

Next, to compare with prior-art robust steganographic algorithms, let us use the same dataset to evaluate the effectiveness of the proposed scheme. For instance, the 100 cover images are used for embedding. Table 7 shows the R_{error} of different algorithms under JPEG compres-

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2019.2953504, IEEE Access

TABLE 7. R_{error} of our scheme with RS, FRAS, DCRAS and DMAS($\times 10^{-3}$).

Embedding algorithm \ Payloads	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09	0.1	Average
FRAS [26]	0	0	0	0	0	0	0	0	0	2.1	0.21
DCRAS [28]	0	0	0	0	0	0	0	0	0.2	0.3	0.05
DMAS [31]	0	0	0	0	0.2	0.4	0.3	0.4	0.6	0.9	0.28
Our scheme with RS	0	0	0	0	0	0.06	0.05	0.09	0.04	0.05	0.03

sion with QF 75. When the payload is not larger than 0.04 bpnzAC, the R_{error} of all four methods equals to 0. As expected, with increasing the payload, the R_{error} gradually increases. It is worth noticing that the R_{error} of our scheme with RS always remains lower than (or equal to) 0.09×10^{-3}. Specifically, when the payload is greater than (or equal to) 0.08 bpnzAC, the R_{error} of DCRAS or DMAS method has an obvious increasing trend, while ours remains stable. When the payload is 0.1, our scheme with RS preforms obviously better than the others. In addition, based on different payloads, the average values by adopting our scheme with RS is the lowest among the four algorithms. In fact, such a low R_{error} has little impact on the correct interpretation of the secret message. Therefore, our proposed algorithm with RS not only guarantees its undetectability, but also enhances its robustness.

VI. CONCLUSION

In this paper, we analyze the distinguishable characteristics of robust steganography, adaptive (or traditional) steganography, and robust watermarking. Based on our specific analysis, a typical image steganographic system is proposed, in which both clean and dirty channel are first addressed. More importantly, a novel robust steganographic algorithm is designed, that can resist JPEG compression with multiple quality factors. Relying on the sign invariance of the selected DCT coefficients before and after JPEG compression, it is proposed to embed the secret message into the cover elements. Together with the embedding cost function, we select the cover elements from the texture regions to guarantee the minimum distortion caused by robust steganography. Furthermore, with the help of ECC and STCs, the robustness and undetectability of the proposed steganographic algorithm are further improved.

In comparison with adaptive steganography and robust watermarking, the proposed robust steganography perfectly strikes the balance between undetectability and robustness. Compared with J-uniward, stego images generated by our algorithm have a great improvement of robust performance. Compared with DCRAS, FRAS and DMAS, the proposed algorithm has better undetectability performance. In the future study, we will mainly focus on improving the embedding capacity with the prerequisite of ensuring both undetectability and robustness. In addition, we will further study the robustness of the proposed algorithm to resist more image post-processing attacks.

VII. ACKNOWLEDGEMENTS

The authors would like to thank Dr. Qingxiao Guan for his helpful discussions and suggestions on the details of [34]. We also thank Dr. Sheng Li and Dr. Zichi Wang for sharing the codes of method in [35]. What is more, we would like to thank the anonymous reviewers for their valuable comments and helpful suggestions.

REFERENCES

[1] J. Fridrich, Steganography in digital media: principles, algorithms, and applications. Cambridge, U.K.: Cambridge Univ. Press, 2009.
[2] B. Li, J. He, J. Huang, and Y. Q. Shi, "A survey on image steganography and steganalysis," Journal of Information Hiding and Multimedia Signal Processing, vol. 2, no. 2, pp. 142–172, 2011.
[3] T. Qiao, F. Retraint, R. Cogranne, and T. H. Thai, "Source camera device identification based on raw images," in 2015 IEEE international conference on image processing (ICIP). IEEE, 2015, pp. 3812–3816.
[4] T. Qiao, F. Retraint, R. Cogranne, and T. H. Thai, "Individual camera device identification from jpeg images," Signal Processing: Image Communication, vol. 52, pp. 74–86, 2017.
[5] T. Qiao, A. Zhu, and F. Retraint, "Exposing image resampling forgery by using linear parametric model," Multimedia Tools and Applications, vol. 77, no. 2, pp. 1501–1523, 2018.
[6] H. Yao, T. Qiao, M. Xu, and N. Zheng, "Robust multi-classifier for camera model identification based on convolution neural network," IEEE Access, vol. 6, pp. 24 973–24 982, 2018.
[7] T. Qiao, R. Shi, X. Luo, M. Xu, N. Zheng, and Y. Wu, "Statistical model-based detector via texture weight map: application in re-sampling authentication," IEEE Transactions on Multimedia, vol. 21, no. 5, pp. 1077–1092, 2018.
[8] T. Qiao and F. Retraint, "Identifying individual camera device from raw images," IEEE Access, vol. 6, pp. 78 038–78 054, 2018.
[9] Y. Zhao, N. Zheng, T. Qiao, and M. Xu, "Source camera identification via low dimensional prnu features," Multimedia Tools and Applications, vol. 78, no. 7, pp. 8247–8269, 2019.
[10] W. Zhou, W. Zhang, and N. Yu, "A new rule for cost reassignment in adaptive steganography," IEEE Transactions on Information Forensics and Security, vol. 12, no. 11, pp. 2654–2667, 2017.
[11] T. Filler, J. Judas, and J. Fridrich, "Minimizing additive distortion in steganography using syndrome-trellis codes," IEEE Transactions on Information Forensics and Security, vol. 6, no. 3, pp. 920–935, 2011.
[12] T. Filler and J. Fridrich, "Gibbs construction in steganography," IEEE Transactions on Information Forensics and Security, vol. 5, no. 4, pp. 705–720, 2010.
[13] V. Holub and J. J. Fridrich, "Designing steganographic distortion using directional filters." in 2012 IEEE International Workshop on Information Forensics and Security (WIFS), 2012, pp. 234–239.
[14] V. Holub, J. Fridrich, and T. Denemark, "Universal distortion function for steganography in an arbitrary domain," EURASIP Journal on Information Security, vol. 2014, no. 1, p. 1, 2014.
[15] B. Li, M. Wang, J. Huang, and X. Li, "A new cost function for spatial image steganography," in 2014 IEEE International Conference on Image Processing (ICIP), 2014, pp. 4206–4210.
[16] L. Guo, J. Ni, and Y.-Q. Shi, "An efficient jpeg steganographic scheme using uniform embedding." in 2012 IEEE International Workshop on Information Forensics and Security (WIFS), 2012, pp. 169–174.
[17] T. Qiao, C. Ziitmann, R. Cogranne, and F. Retraint, "Detection of jsteg algorithm using hypothesis testing theory and a statistical model with nuisance parameters," in Proceedings of the 2nd ACM workshop on

Information Hiding and Multimedia Security (IH & MMSec), 2014, pp. 3–13.

[18] T. Qiao, C. Zitzmann, F. Retraint, and R. Cogranne, "Statistical detection of jsteg steganography using hypothesis testing theory," in 2014 IEEE International Conference on Image Processing (ICIP), 2014, pp. 5517–5521.

[19] T. Qiao, F. Retraint, R. Cogranne, and C. Zitzmann, "Steganalysis of jsteg algorithm using hypothesis testing theory," EURASIP Journal on Information Security, vol. 2015, no. 1, pp. 1–16, 2015.

[20] T. Pevny and J. Fridrich, "Merging markov and dct features for multi-class jpeg steganalysis," in Security, Steganography, and Watermarking of Multimedia Contents IX, vol. 6505, 2007, p. 650503.

[21] T. Pevny, P. Bas, and J. Fridrich, "Steganalysis by subtractive pixel adjacency matrix," IEEE Transactions on information Forensics and Security, vol. 5, no. 2, pp. 215–224, 2010.

[22] J. Fridrich and J. Kodovsky, "Rich models for steganalysis of digital images," IEEE Transactions on Information Forensics and Security, vol. 7, no. 3, pp. 868–882, 2012.

[23] J. Kodovskỳ, J. Fridrich, and V. Holub, "Ensemble classifiers for steganalysis of digital media," IEEE Transactions on Information Forensics and Security, vol. 7, no. 2, pp. 432–444, 2012.

[24] Q. Liu, T. Qiao, M. Xu, and N. Zheng, "Fuzzy localization of steganographic flipped bits via modification map," IEEE Access, vol. 7, pp. 74 157–74 167, 2019.

[25] Y. Zhang, C. Qin, W. Zhang, F. Liu, and X. Luo, "On the fault-tolerant performance for a class of robust image steganography," Signal Processing, vol. 146, pp. 99–111, 2018.

[26] Y. Zhang, X. Luo, C. Yang, D. Ye, and F. Liu, "A jpeg-compression resistant adaptive steganography based on relative relationship between dct coefficients," in 2015 10th International Conference on Availability, Reliability and Security (ARES), 2015, pp. 461–466.

[27] W. Luo, G. L. Heileman, and C. E. Pizano, "Fast and robust watermarking of jpeg files," in Proceedings Fifth IEEE Southwest Symposium on Image Analysis and Interpretation, 2002, pp. 158–162.

[28] Y. Zhang, X. Luo, C. Yang, and F. Liu, "Joint jpeg compression and detection resistant performance enhancement for adaptive steganography using feature regions selection," Multimedia Tools and Applications, vol. 76, no. 3, pp. 3649–3668, 2017.

[29] K. Mikolajczyk and C. Schmid, "Scale & affine invariant interest point detectors," International Journal of Computer Vision, vol. 60, no. 1, pp. 63–86, 2004.

[30] J.-S. Tsai, W.-B. Huang, Y.-H. Kuo, and M.-F. Horng, "Joint robustness and security enhancement for feature-based image watermarking using invariant feature regions," Signal Processing, vol. 92, no. 6, pp. 1431–1445, 2012.

[31] Y. Zhang, X. Zhu, C. Qin, C. Yang, and X. Luo, "Dither modulation based adaptive steganography resisting jpeg compression and statistic detection," Multimedia Tools and Applications, vol. 77, no. 14, pp. 17 913–17 935, 2018.

[32] J. Xiao and Y. Wang, "Adaptive dither modulation image watermarking algorithm," Journal of Electronics & Information Technology, vol. 31, no. 3, pp. 552–555, 2009.

[33] Y. Zhang, X. Luo, Y. Guo, C. Qin, and F. Liu, "Zernike moment-based spatial image steganography resisting scaling attack and statistic detection," IEEE Access, vol. 7, pp. 24 282–24 289, 2019.

[34] Z. Zhao, Q. Guan, H. Zhang, and X. Zhao, "Improving the robustness of adaptive steganographic algorithms based on transport channel matching," IEEE Transactions on Information Forensics and Security, vol. 14, no. 7, pp. 1843–1856, 2018.

[35] J. Tao, S. Li, X. Zhang, and Z. Wang, "Towards robust image steganography," IEEE Transactions on Circuits and Systems for Video Technology, vol. 29, no. 2, pp. 594–600, 2019.

[36] G. J. Simmons, "The prisoners' problem and the subliminal channel," in Advances in Cryptology, 1984, pp. 51–67.

[37] F. Yan, M. Xu, T. Qiao, T. Wu, X. Yang, N. Zheng, and K.-K. R. Choo, "Identifying wechat red packets and fund transfers via analyzing encrypted network traffic," in 2018 17th IEEE International Conference On Trust, Security And Privacy In Computing And Communications/12th IEEE International Conference On Big Data Science And Engineering (TrustCom/BigDataSE), 2018, pp. 1426–1432.

[38] Y. Wang, N. Zheng, M. Xu, T. Qiao, Q. Zhang, F. Yan, and J. Xu, "Hierarchical identifier: Application to user privacy eavesdropping on mobile payment app," Sensors, vol. 19, no. 14, p. 3052, 2019.

[39] A. Wilson, P. Blunsom, and A. D. Ker, "Linguistic steganography on twitter: hierarchical language modeling with manual interaction," in Media Watermarking, Security, and Forensics 2014, vol. 9028, 2014, p. 902803.

[40] A. D. Ker, P. Bas, R. Böhme, R. Cogranne, S. Craver, T. Filler, J. Fridrich, and T. Pevnỳ, "Moving steganography and steganalysis from the laboratory into the real world," in Proceedings of the first ACM workshop on Information Hiding and Multimedia Security (IH & MMSec), 2013, pp. 45–58.

[41] P. Bas, T. Filler, and T. Pevnỳ, ""break our steganographic system": The ins and outs of organizing boss," in Proceedings of the 13th International Conference on Information Hiding (IH), 2011, pp. 59–70.

[42] J. Kodovskỳ and J. Fridrich, "Steganalysis of jpeg images using rich models," in Media Watermarking, Security, and Forensics 2012, vol. 8303, 2012, p. 83030A.