Reversible Data Hiding in EtC Images with Flexible Access Privileges

Yusaku Kato and Shoko Imaizumi
Chiba University, Chiba, Japan
E-mail: y.kato@chiba-u.jp, imaizumi@chiba-u.jp

***Abstract*—In this paper, we propose a novel method of reversible data hiding in encrypted images derived by an encryption-then-compression (EtC) system. We call such encrypted images EtC images hereafter. There are two main contributions in the proposed method: flexible access privileges and high image quality. First, the previous reversible data hiding (RDH) method for EtC images has a constraint where a payload cannot be extracted in the case that decryption is conducted prior to data extraction. The proposed method solves the constraint by storing the encryption data in the output image, that is, the marked EtC image. Second, the pre-processing for data hiding is performed before encryption so that the block distortion of an image decrypted without data extraction, called a marked image, can be avoided. Through our experiments, we evaluate the effectiveness of the proposed method in terms of hiding capacity and marked-image quality. The original image could be restored without regard to the order between decryption and data extraction in the experiments.**

I. Introduction

With the recent development of social networking services and cloud services, copyright and privacy protection of shared images has become increasingly important. One image protection method is data hiding. Data hiding methods are divided into two types: reversible and irreversible. The former can fully recover original images by extracting embedded data (hereafter, payload) from marked images [1], [2]. This method is therefore strongly required in fields such as medical and satellite images. Recently, reversible data hiding in encrypted image (RDH-EI) methods have also been studied actively [3]–[8]. In RDH-EI methods, an image owner first encrypts the original image and sends the encrypted one to a third-party organization. The third-party organization receives the encrypted image and then embeds a payload into the image. The payload is supposed to be data for copyright, authentication, management, and so forth. Hereafter, we refer to the third-party organization as the data hider. The encryption and data hiding processes are performed by the image owner and the data hider independently. Therefore, the payload can be embedded into the encrypted image without disclosing the image content to the data hider.

Motomura et al. proposed a method for images encrypted by an encryption-then-compression (EtC) system [8]. The EtC system is based on block-wise encryption that enables the compression of encrypted images. Hereafter, we refer to images encrypted by the EtC system as EtC images and the method of [8] as the RDH in EtC images (RDH-EtCI) method. In the method, a payload is embedded into each block of an EtC image using an extended algorithm of the prediction error expansion with histogram shifting (PEE-HS) method [9]. An image owner performs only image encryption, while a data hider prepares the embeddable area and then embeds a payload. This provides a clear distinction between the encryption and data hiding processes. An encrypted image with a payload is called a marked encrypted-image. We can assign different access privileges to this image. Fig.1 illustrates four access privileges for marked encrypted-images. The RDH-EtCI method provides the three types of access privileges shown in Figs.1(a) to (c) but cannot give the privilege shown in Fig.1(d). In addition, block distortion may occur in the marked image, which is obtained by decrypting the marked encrypted-image, namely the marked EtC-image.

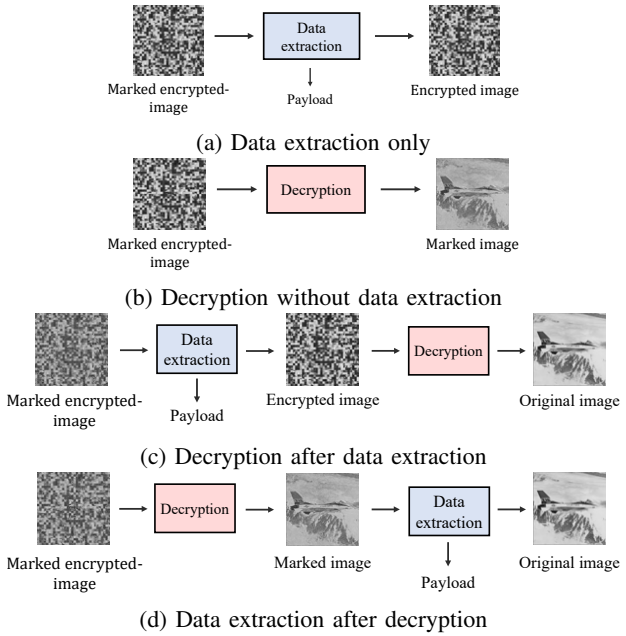


Fig. 1: Access privileges in RDH-EI.

In this paper, we achieve all four access privileges shown in Fig.1 and eliminate the block distortion in marked images caused by the RDH-EtCI method. The proposed method stores encryption data within the four corner pixels of each block in the EtC image. By referring to this data, the payload is extracted, and the original image can be restored. Additionally, pre-processing for data hiding is carried out before encryption to avoid block distortion in marked images. In our experiments,

979-8-3315-7206-8/25/$31.00

we evaluate the effectiveness of the proposed method in terms of the hiding capacity and marked-image quality and discuss the robustness of EtC images.

II. Reversible Data Hiding in Encrypted Images

In this section, we give an overview of RDH-EI and some previous works related to the proposed method.

A. Overview

In RDH, a payload is embedded into a plain image. In contrast, the payload is hidden into an encrypted image in RDH-EI [3]–[8]. An image owner first encrypts an original image and sends the encrypted image to a data hider. The data hider then embeds a payload in the encrypted image such as copyright, authentication, and management data.

RDH-EI methods can be broadly classified into two forms: reserving room before encryption (RRBE) and vacating room after encryption (VRAE). In the RRBE format, the image owner previously reserves the embeddable area before encryption. Sui et al. proposed a method with a hiding capacity of 3.39 bpp by using most significant bit (MSB) prediction and Huffman coding [3]. In more recent research, the hiding capacity has been improved to 4.17 bpp [4]. In the method of [4], an original image is divided into blocks; each block is then asymmetrically encoded on the basis of prediction errors.

In contrast to the RRBE format, in the VRAE format, the data hider vacates an embeddable area and embeds a payload into the encrypted image. Thus, the image owner does not need to allocate the embeddable area, and the data hider can conceal the area from the image owner. From the above perspective, the VRAE format is more practical than the RRBE format. Chen et al. proposed a method using MSB prediction for block-based encrypted images and achieved a hiding capacity of 3.00 bpp [5]. Xiao et al. further proposed another novel method for block-based encrypted images with a hiding capacity of 4.04 bpp [6]. In the method of [6], prediction errors are calculated for pixel values in each block and then compressed using the more effective of the two encoding schemes.

The methods described above, however, directly replace the pixel values of the encrypted image with a payload. Therefore, it is necessary to extract the payload prior to decryption in the restoration process.

B. Related Works

Most RDH-EI methods such as [3]–[6] have only the two types of access privileges shown in Figs.1(a) and (c). They cannot decrypt marked encrypted-images without data extraction and thus cannot provide marked images. This is because their restoration process assumes that a payload should be previously encrypted and that decryption is conducted after data extraction. However, it is also required to extract the payload from the marked image.

To address this issue, a method using bit-plane partition has been proposed [7]. This method achieves all four access privileges by performing the encryption and data hiding processes in two independent areas. The method further has a high hiding capacity of 2.50 bpp. Hereafter, we refer to this method as the RDH-bit-plane-based encrypted images (RDH-BPEI) method. The RDH-BPEI method is, however, classified with the RRBE format. Moreover, by previously defining the number of bit-planes used for the embeddable area as α, the hiding capacity is limited to α bpp or less.

On the other hand, the RDH-EtCI method [8] belongs to the VRAE format. Although VRAE methods generally have greater constraints on restoration than RRBE methods, the RDH-EtCI method can apply the three access privileges shown in Figs.1(a) to (c). Using the method, a hiding capacity of 2.45 bpp was achieved by embedding a payload into an EtC image using an extended algorithm of the PEE-HS method [9]. The method, however, does not support the other access privilege shown in Fig.1(d), where a payload is extracted from a marked image. Moreover, block distortion may be caused in the marked image.

In the next section, we propose a novel RDH-EtCI method to tackle the above issues. Specifically, we describe a process for storing encryption data. In addition, we modify the order of the pre-processing for data hiding.

III. Proposed Method

In this section, we extend the RDH-EtCI method [8] so that all the access privileges are available in the VRAE format. In the proposed method, the encryption data is stored in each block of an EtC image. This enables data extraction after decryption, which is the fourth access privilege. Additionally, we carry out the pre-processing for data hiding prior to encryption. The pre-processing is an essential process to prevent overflows and underflows in data hiding. This modification allows us to avoid block distortion in marked images. We explain the specific procedures as follows.

A. Marked EtC-Image Derivation

We obtain a marked EtC-image by embedding a payload into an EtC image. Fig.2 shows the flow of marked EtC-image derivation. First, differing from the RDH-EtCI method, our method carries out the pre-processing before encryption. The remaining process consists of three main steps: encryption using the EtC system, encryption-data storage, and data hiding. In the encryption process, we divide each main block into sub blocks in order to reduce the amount of encryption data. Each step is described below.

1) Encryption using EtC system: The proposed method introduces an extended EtC system using main and sub blocks [10]. This reduces the amount of the encryption data to be stored in the EtC image. First, a pre-processed image with $H \times W$ pixels is divided into main blocks with $B_m \times B_m$ pixels; each main block is further divided into sub blocks with $B_s \times B_s$ pixels. Thus, the numbers of main blocks N_m and sub blocks N_s are expressed by

$$N_m = \frac{H}{B_m} \times \frac{W}{B_m}, \tag{1}$$

$$N_s = \left(\frac{B_m}{B_s}\right)^2, \tag{2}$$

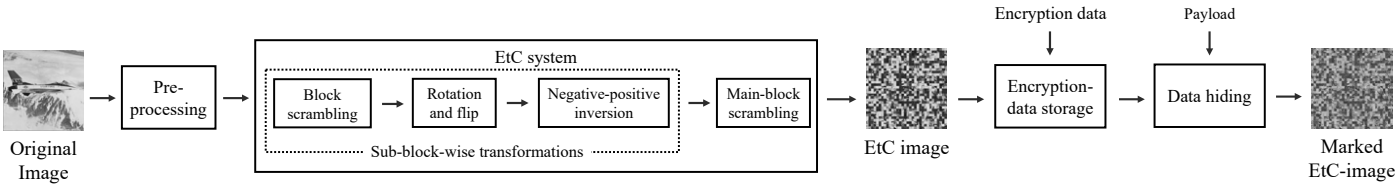
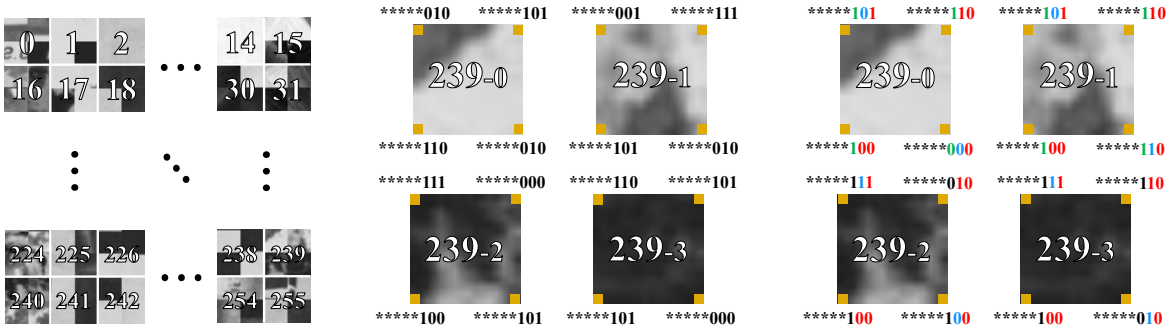


Fig. 2: Block diagram of marked EtC-image derivation.



(a) Block division of EtC image (b) Original values of four corner pixels in each sub block (c) Pixel values after data storage

Fig. 3: Example of encryption-data storage ($M_n = 239$).

respectively. Next, block scrambling within each main block, rotation/flip, and negative-positive inversion (hereafter, NP inversion) are applied to each sub block. For these processes, different keys are assigned to each main block. Finally, the main blocks are randomly permuted, and the EtC image is obtained.

2) Encryption-data storage: For each block of an EtC image, we store two types of encryption data: sub-block data and block number. These data will be contained within the four corner pixels in each sub block and used to extract a payload from the marked image. Specifically, sub-block data specifies the scan order of pixels within a sub block. The block number determines the order of sub and main blocks from which the payload is extracted. Fig.3 shows an example of encryption-data storage for an EtC image with $N_m = 256$ and $N_s = 4$. The specific procedure is given below.

Step 1-1: For N_m main blocks in the EtC image, assign a main-block number M_n, where $0 \leq M_n \leq N_m - 1$, in raster scan order (see Fig.3(a)).

Step 1-2: For each of N_s sub blocks in a main block, assign a sub-block number S_n, where $0 \leq S_n \leq N_s - 1$, in raster scan order.

Step 1-3: Store the sub-block data into the four corner pixels of each sub block. First, replace the least significant bit (LSB) of the top-left pixel with '1' and each LSB of the other three corner pixels with '0'. Next, replace the second lower bit of the top-right pixel with '1' and that of the bottom-left pixel with '0' (see the red-colored bits in Fig.3(c)).

Step 1-4: For the top-left and bottom-right pixels in each sub block, replace the second lower bits with each S_n (see the blue-colored bits in Fig.3(c)). Note that the bits of S_n were stored in the order of the top-left and bottom-right pixels in this example.

Step 1-5: Store M_n for each main block. Starting with the four corner pixels of the sub block with the smallest S_n, their third lower bits are replaced with the bits of M_n, which are stored in raster scan order (see the green-colored bits in Fig.3(c)). The number of bits to be replaced is defined by $\lceil log_2 N_m \rceil$. In the case of $\lceil log_2 N_m \rceil > 4N_s$, the fourth and above bits are also used for replacement.

3) Data hiding: In the data hiding process, in addition to the data embedded by a data hider (hereafter, pure payload), the data for image restoration from a marked image (hereafter, restoration data) should be embedded into all the sub blocks in an EtC image. The restoration data includes the original values of pixels modified by the pre-processing, a sequence of lower bits replaced by the encryption data, and so forth. The payload is embedded into pixels of each sub block in raster scan order using an extended algorithm of the PEE-HS method [9]. Here, the four corner pixels in each sub block are excluded from the hiding process. Data hiding is carried out in ascending order of M_n and S_n; the process starts from the sub block with $(M_n, S_n) = (0, 0)$ and ends at the sub block with $(M_n, S_n) = (N_m - 1, N_s - 1)$. Note that an arbitrary RDH method based on HS can be used for data hiding instead of the PEE-HS method.

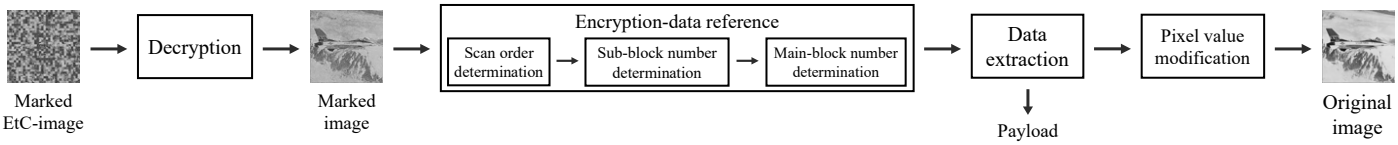
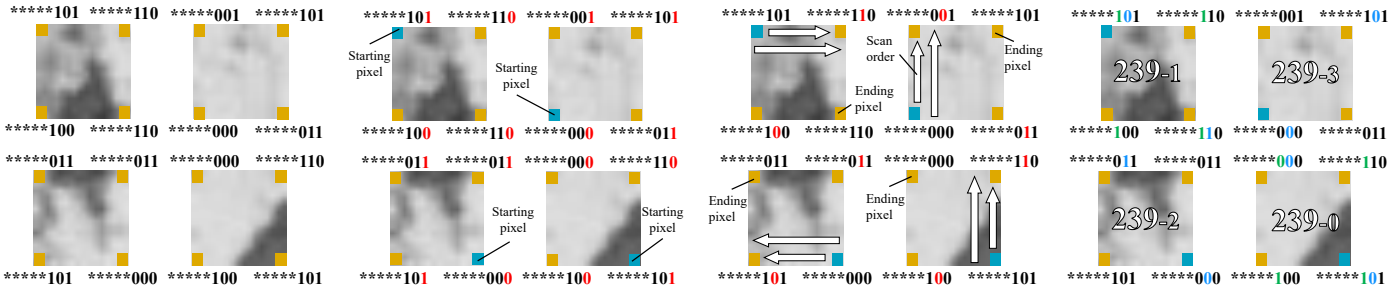


Fig. 4: Block diagram of data extraction after decryption.



(a) Decrypted values of four corner pixels (b) Starting pixels for scanning (c) Scan orders (d) Retrieved block numbers

Fig. 5: Example of encryption-data reference in each sub block ($M_n = 239$).

B. Image Restoration

The proposed method provides all four access privileges for marked EtC-images. First, the two privileges in Figs.1(a) and (c), where the payload is extracted prior to decryption, can be simply conducted in the reverse order of marked EtC-image derivation. The privilege in Fig.1(b), where decryption is performed without data extraction, can also be conducted in the same way as the RDH-EtCI method. Generally, decryption before data extraction derives marked images with low quality. Since the proposed method uses the EtC system for encryption as well as the EtCI method, there is still redundancy among pixel values within a block. Thus, RDH methods based on HS, which basically modify the lower bits of each pixel, can be used for EtC images. This allows us to obtain high-quality marked images even when marked EtC images are directly decrypted.

In contrast to the RDH-EtCI method, the proposed method also allows data extraction from marked images, which is the other access privilege. The process for this privilege is shown in Fig.4. The marked EtC-image is first decrypted; we then obtain the marked image. We then retrieve the encryption data, which allows us to extract the payload, from the marked image. The procedure for encryption-data reference is described in detail below.

Fig.5 outlines encryption-data reference and corresponds to Fig.3. Fig. 5(a) is a decrypted main block of $N_m = 239$, which is equal to the main block shown in Fig.3(c). To identify the scan order of pixels, we first refer to the sub-block data in the following steps.

Step 2-1: Refer to the LSBs of the four corner pixels in each sub block to identify if NP inversion is applied to each sub block. For instance, when a single ‘1’ and three ‘0’s are detected in a sub block, we determine that NP inversion has not been applied to the sub block. Conversely, when a single ‘0’ and three ‘1’s are detected, NP inversion has been applied.

Step 2-2: Identify a starting pixel for scanning in each sub block from the LSBs of the four corner pixels (see Fig.5(b)). In the case that NP inversion has not been applied, the pixel with the LSB of ‘1’ is the starting point, and vice versa.

Step 2-3: Refer to the second lower bits of the other two corner pixels in the same row or column as the starting pixel (see the red-colored bits in Fig.5(c)).

Step 2-4: From the above two bits, select one bit that is equal to the LSB of the starting pixel. The scan is done in the direction of the pixel with that single bit from the starting pixel.

Note that for sub blocks with NP inversion, a part of the payload from them will be correctly obtained by flipping the bit sequence to be extracted.

Next, we identify the main and sub block numbers to determine the order of data extraction. For each sub block in a main block, we first obtain a sub-block number S_n by referring to the second lower bits of the two corner pixels: the starting and ending pixels (see the blue-colored bits in Fig.5(d)). In Fig.5(d), for example, the top-left sub block did not undergo NP inversion. In this case, S_n is defined as $(01)_2 = 1$ without flipping the bits. In contrast, the bottom-left sub block passed through NP inversion. S_n, directly referred to from the starting and ending pixels, is expressed as ‘01’, but the bits will be flipped to $(10)_2 = 2$.

Further, main-block numbers are obtained on the basis of the sub-block numbers and scan order. We scan the four sub blocks in sub-block numerical order; the third lower bit of each corner

pixel is referred to in the previously obtained scan order shown in Fig.5(c). After $\lceil log_2 N_m \rceil$ of these bits are arranged from the MSB, the main-block number can be obtained. In the case of Fig.5(d), for example, N_m is 256, so $\lceil log_2 N_m \rceil$ becomes eight. Accordingly, by referring to the eight green-colored bits, which were first scanned, M_n is defined as $(11101111)_2 = 239$.

Finally, the payload is extracted in the block order defined as described above. The original image is also reconstructed by replacing the lower bits with the original ones contained in the payload.

C. Advantages of Proposed Method

The proposed method has two main advantages: flexibility of access privileges and inhibition of block distortion in marked images. The details of each advantage are described below.

First, the proposed method provides all four access privileges. In the RDH-EtCI method, we cannot extract the payload from the marked image obtained by directly decrypting a marked EtC-image. Direct decryption before data extraction results in loss of the pixel-scan order for later data extraction. In contrast, our method preserves the scan order within the marked image containing the encryption data. This enables the proposed method to achieve the fourth privilege, that is, data extraction from the marked image. In general, it is difficult to realize the fourth privilege with the VRAE format. The contribution of the proposed method is highly significant.

Second, the proposed method prevents block distortion in marked images. When the pre-processing is performed after encryption such as with the RDH-EtCI method, pixel values in the entire EtC image might be significantly changed. If the EtC image is decrypted in this case, differences between pixel values in sub blocks with and without NP inversion are enlarged. This causes block distortion and detracts from the quality of marked images. By conducting the pre-processing on the original image before encryption, the proposed method eliminates the effect of NP inversion and avoids generating block distortion.

IV. Experimental Results

The effectiveness of the proposed method was evaluated from three perspectives: hiding capacity, marked-image quality, and EtC-image security. We used 24 test images from the Kodak Lossless True Color Image Suite [12]. Each image was preliminarily converted from RGB to grayscale and then resized to 512×768 or 768×512 pixels by bicubic interpolation. The sizes of the main and sub blocks were 32×32 and 16×16 pixels; the numbers of main and sub blocks were 1,536 and 384, respectively. Fig.6 shows an example of a marked EtC-image obtained by the proposed method. It is difficult for us to visually infer the original image from the encrypted image. For comparison with the proposed method, we conducted similar experiments with the RDH-BPEI [7] and the RDH-EtCI [8] methods. Note that these methods use the EtC system, consisting of only main blocks with 16×16 pixels.

Prior to the evaluation of the above three perspectives, we first confirmed the reversibility of the proposed method. The PSNR and SSIM showed ∞ and 1.0 for the plain images recovered by the access privileges in Figs.1(c) and (d), when comparing them with their original images. In particular, with the privilege in Fig.1(d), the payload could be accurately extracted from the marked images, even after decryption.

(a) Original image (b) Marked EtC-image

Fig. 6: Example of marked EtC-image (kodim04).

TABLE I: Performance comparison of proposed method and conventional methods [7], [8].

	Average of hiding capacity [bpp]	Number of available access privileges	Block distortion	Format
Proposed	3.13	4	None	VRAE
RDH-BPEI [7]	2.51	4	None	RRBE
RDH-EtCI [8]	3.17	3	Present	VRAE

A. Hiding Capacity

In Table I, we summarized the average hiding capacity and features of the proposed, RDH-BPEI [7], and RDH-EtCI [8] methods. Here, for the RDH-BPEI method, we set the number of bitplanes α used for the embeddable area to three. From the table, our method has a hiding capacity of 3.13 bpp, which is much higher than that of the RDH-BPEI method. However, our hiding capacity is slightly less than that of the RDH-EtCI method. This is because the storage of encryption data in the proposed method causes an increase in the total amount of restoration data. This storage, however, enables us to extract a payload from a marked image and to restore the original image; this was difficult for the RDH-EtCI method.

B. Marked-Image Quality

We evaluated the quality of marked images obtained by the proposed and two previous conventional methods [7], [8]. Fig.7 shows marked images, which include a pure payload of 1.0 bpp, by the three methods. From Fig.7(c), the image obtained by the RDH-EtCI method has block distortion. This is because the pre-processing was conducted after encryption. In comparison, as shown in Figs.7(a) and (b), the proposed and RDH-BPEI methods did not cause any block distortion; they carry out the pre-processing prior to encryption.

(a) Proposed (b) RDH-BPEI [7] (c) RDH-EtCI [8]

Fig. 7: Marked images with payload of 1.0 bpp (kodim04).

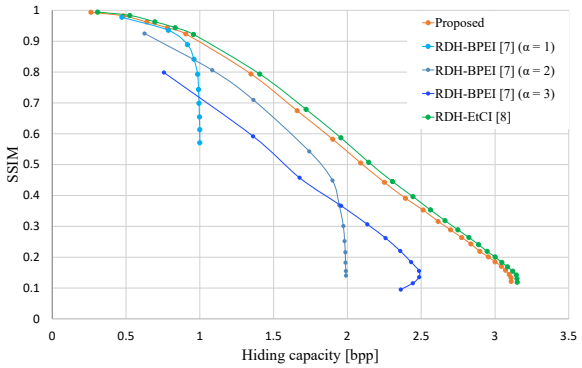


Fig. 8: Transition in marked-image quality for SSIM.

Fig.8 illustrates the transition in SSIM against the hiding capacity. In this figure, the results of the RDH-BPEI method are shown for α = 1, 2, and 3. It is clear that the proposed method outperformed the RDH-BPEI method for all three values of α. In contrast, the RDH-EtCI method had constantly higher quality than our method. The RDH-EtCI method does not prepare encryption data, so the change in each pixel value across the entire image is smaller than the proposed method. However, the RDH-EtCI method does not use encryption data and thus cannot extract payloads from marked images.

C. Security Analysis

The security of the EtC system used in the proposed method is first discussed in terms of the key space. A key space is the total number of keys that can be generated in an encryption method; the larger the key space is, the higher the resistance against brute force attacks becomes. In the experiments, the RDH-BPEI and RDH-EtCI methods performed main-block scrambling, rotation/flip, and NP inversion on all 1,536 blocks, resulting in a key space of approximately 10^{6078}. On the other hand, the proposed method uses an EtC system that introduces not only main blocks but also sub blocks to reduce the amount of the encryption data. In this system, the key space S is given by

$$S = (8^{N_s} \times 2^{N_s} \times N_s!)^{N_m}. \quad (3)$$

Since $N_m = 384$ and $N_s = 4$ in this experiment, S is reduced to approximately 10^{3206}. This value is, however, sufficiently larger compared with the key lengths used in general encryption methods.

Moreover, previous research [10] confirmed that another EtC system with main and sub blocks is sufficiently resistant to the extended jigsaw puzzle solver (EJPS) attack [11]. The EJPS attack is one of the ciphertext-only attacks and effectively recomposes visual content from encrypted images. In particular, when a compressible image encryption method such as the EtC system is adopted, the EJPS attack poses a great threat. The EtC system in [10] is applied to color images and conducts color channel shuffling instead of main block scrambling. The EtC system used in the proposed method is slightly different from the above system but is equally tolerant.

V. Conclusion

We proposed an effective RDH-EI method that enables all four access privileges in this paper. The proposed method stores encryption data into the four corner pixels of each sub block so that a payload can be extracted from a marked image. We further allocate the pre-processing for data hiding before encryption; this inhibits the block distortion caused in marked images. Note that the proposed method is the VRAE format. Through our experiments, we verified that the method has both a high hiding capacity of 3.13 bpp and high quality of marked images. Additionally, we confirmed that our method can correctly reconstruct the original form of an image and payload regardless of the restoration order.

References

[1] Y.-Q. Shi et al., "Reversible data hiding: advances in the past two decades," IEEE Access., vol.4, pp.3210–3237, 2016.
[2] H.-T. Wu et al., "Reversible image data hiding with contrast enhancement," IEEE Signal Processing Letters, vol.22, no.1, pp.81-85, 2014.
[3] L. Sui et al., "Reversible data hiding in encrypted images based on hybrid prediction and Huffman coding," Symmetry, vol.15, no.6, 1222, 2023.
[4] X. Zhang et al., "Reversible data hiding in encrypted images with asymmetric coding and bit-plane block compression," IEEE Trans. Multimedia, vol.26, pp.10174-10188, 2024.
[5] S. Chen et al., "Reversible data hiding in encrypted images using block-based adaptive MSBs prediction," Journal of Information Security and Applications, vol.69, 103297, 2022.
[6] F. Xiao et al., "High-capacity reversible data hiding in encrypted images based on adaptive block coding selection," Journal of Visual Communication and Image Representation, vol.104, 104291, 2024.
[7] E. Arai et al., "High-capacity reversible data hiding in encrypted images with flexible restoration," Journal of Imaging, vol.8, no.7, 176, 2022.
[8] R. Motomura et al., "Reversible Data Hiding in Compressible Encrypted Images with Capacity Enhancement," APSIPA Transactions on Signal and Information Processing, vol.12, no.1, 2023.
[9] D. M. Thodi et al., "Expansion embedding techniques for reversible watermarking," IEEE Trans. Image Process., vol.16, no.3, pp.721–730, 2007.
[10] H. Lin et al., "Privacy-Preserving ConvMixer Without Any Accuracy Degradation Using Compressible Encrypted Images," Information, vol.15, no.11, 723, 2024.
[11] T. Chuman et al., "A Jigsaw Puzzle Solver-Based Attack on Image Encryption Using Vision Transformer for Privacy-Preserving DNNs," Information, vol.14, no.6, 311, 2023.
[12] [Online]Available https://www.r0k.us/graphics/kodak/