Flexible Data Hiding and Extraction in EtC Images

Ryoichi HIRASAWA*, Shoko IMAIZUMI† and Hitoshi KIYA‡
* Graduate School of Science and Engineering, Chiba University, Chiba, Japan
E-mail: ryoichi0616@chiba-u.jp
† Graduate School of Engineering, Chiba University, Chiba, Japan
E-mail: imaizumi@chiba-u.jp
‡ Faculty of System Design, Tokyo Metropolitan University, Tokyo, Japan
E-mail: kiya@tmu.ac.jp

***Abstract*—This paper proposes a data hiding method for compressible encrypted images called the encryption-then-compression (EtC) images. We allocate bit planes of an original image to an encryption field and a data hiding field, where the upper 7 bits are encrypted by a compressible encryption (CE) method and LSBs are replaced with payload bits. By conducting the encryption and data hiding processes in the independent fields, the proposed method can embed and extract the payload flexibly without any complex condition and preprocessing. Experimental results and discussion show the effectiveness of the proposed method in terms of lossless compression efficiency by using JPEG-LS and JPEG 2000, marked-image quality, and robustness against attacks.**

I. Introduction

Compression efficiency of encrypted images is currently an important topic in image encryption. Encryption-then-compression (EtC) systems, where an image is first encrypted and then compressed, have been studied [1]–[8]. In EtC systems, a certain type of compressible encryption (CE) [6]–[8] can compress encrypted images by using international image compression standards, such as JPEG, JPEG-LS [9] and JPEG2000 [10]. Hereafter, EtC images denote such compressible encrypted images in this paper. The CE method first divides an original image into fixed-size blocks and conducts four processes: positional scrambling, block rotation/flip, negative-positive transformation, and color component shuffling. Thus the correlation among pixels within each block is preserved. Consequently, the EtC images obtained by the CE method can be highly compressed by international compression standards. Here, we define CE in this paper as such block-based encryption.

Image data hiding techniques to embed a payload into an image without visible artifacts have been actively studied [11]–[18]. In such researches, a payload is embedded into an original image without considering encryption. In contrast, data hiding for encrypted images is also receiving increased attention [19]–[23]. A third party, such as the system/channel administrator, may embed additional information into encrypted images. On another hand, an image user would desire to obtain a high-quality image after decryption; there still exists the payload in the image. For authentication, it is further helpful to extract the hidden data from the decryption-only image, i.e., the plain domain. Parah [22] proposed a data hiding method, where payload bits are replaced with LSBs [11] of an encrypted image. This method is a steganographic method that mainly protects the payload itself, and ensures double layer data security, which consists of pixel-based encryption and bit-plane substitution. However, the payload cannot be extracted after decryption in this method. When a user aims to extract the payload from the decrypted image, the user first needs to encrypt the decrypted image again.

When a payload is image notation or authentication data, data hiding techniques for encrypted images are useful for image retrieval in the encrypted domain. If the payload is extracted from the encrypted domain, the encrypted image can be searchable without decryption. A flexible reversible data hiding method has been proposed, which can embed different payloads into both the plain and encrypted domains [23]. In this method, the payloads can be extracted from either domain regardless of the process sequence. This method, however, requires complex conditions to define the data hiding order and the target blocks to be encrypted.

In this paper, we propose an efficient data hiding method for encrypted images. Our method first conducts bit-plane slicing and assigns the bit planes to the encryption and data hiding processes exclusively; thus these processes can be performed independently without interfering each other. We consequently do not need to define any complex condition such as in the previous work [23]. Additionally, the output image can be highly compressed by JPEG-LS and JPEG 2000 due to the use of CE. Meanwhile, in the proposed method, the data hiding capacity is 3 bpp by using LSB substitution for data hiding. Note that the LSB substitution is an irreversible data hiding algorithm, and thus the original image cannot be perfectly retrieved. We verify the output images derived by the proposed method in terms of lossless compression performance using JPEG-LS and JPEG 2000, and quality of decryption-only images, namely, marked images. We further discuss robustness against attacks.

II. Compressible Encryption [6]

Figure 1 shows the CE procedure [6]. CE divides an original image into fixed-size blocks and performs four processes: positional scrambling, block rotation and block flip, negative-positive transformation, and color component shuffling. After the above processes, all the blocks are integrated into a

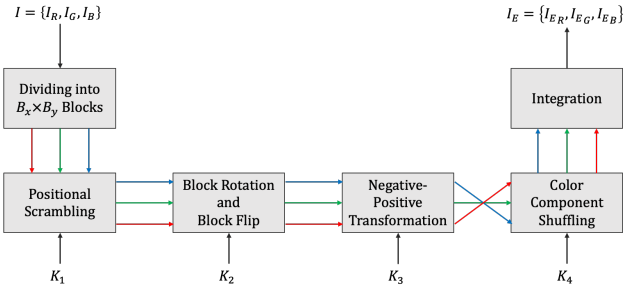


Fig. 1 Outline of CE [6].

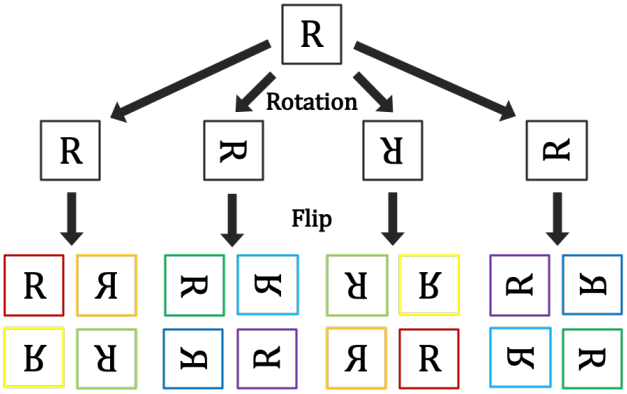


Fig. 2 Block rotation and block flip.

single encrypted image called the EtC image. The encryption procedure is described as follows.

Step1: Divide a 24-bit color image $I = \{I_R, I_G, I_B\}$ with $M \times N$ pixels into multiple blocks with $B_x \times B_y$ pixels.

Step2: Scramble the position of each block using a pseudo-random number sequence generated by key K_1.

Step3: Rotate and flip each block using a pseudo-random number sequence generated by key K_2.

Step4: Apply the negative-positive transformation to each block using a pseudo-random number sequence generated by key K_3.

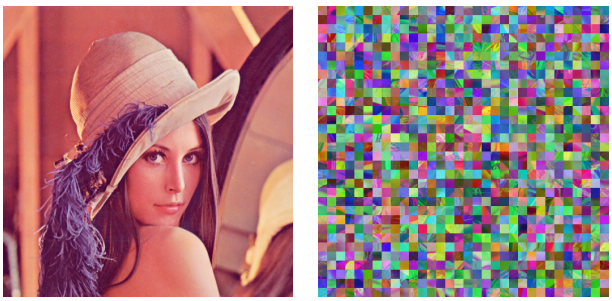
Step5: Shuffle the R, G, and B components in each block using a pseudo-random number sequence generated by key K_4.

Step6: Integrate all blocks and generate the encrypted image $I_E = \{I_{E_R}, I_{E_G}, I_{E_B}\}$.

We clarify the main four processes in more detail below.

A. Positional Scrambling

The positions of the divided blocks are shuffled with a pseudo-random number sequence generated from key K_1.



(a) Original image (b) EtC image

Fig. 3 Resulting image by CE method (block size: 16×16 pixels).

B. Block Rotation and Block Flip

Each block is rotated 0, 90, 180, or 270 degrees, and then flipped horizontally and/or vertically or is not flipped with a pseudo-random number sequence generated from key K_2. Figure 2 illustrates the procedure of rotation and flip. Each of rotation and flip has four operation types. There are 16 combinations when simply combining those two processes. Some combinations, however, correspond to other combinations. Accordingly, the number of total block patterns is eight.

C. Negative-Positive Transformation

The negative-positive transformation reverses the entire pixel values in each block with a pseudo-random number sequence generated from key K_3. In this operation, the transformed pixel value p' is given by

$$p' = \begin{cases} p & (r_{NPT}(i) = 0) \\ 255 - p & (r_{NPT}(i) = 1), \end{cases} \tag{1}$$

where p denotes the original pixel value and $r_{NPT}(i)$ is a pseudo-random number for the i-th block, which is generated from key K_3.

D. Color Component Shuffling

The color component shuffling permutes the R, G, and B components in each block with a pseudo-random number sequence generated from key K_4. In this operation, a set of the transformed color component c' is obtained by

$$c' = \begin{cases} \{c_R, c_G, c_B\} & (r_{CCS}(i) = 0) \\ \{c_R, c_B, c_G\} & (r_{CCS}(i) = 1) \\ \{c_G, c_R, c_B\} & (r_{CCS}(i) = 2) \\ \{c_G, c_B, c_R\} & (r_{CCS}(i) = 3) \\ \{c_B, c_R, c_G\} & (r_{CCS}(i) = 4) \\ \{c_B, c_G, c_R\} & (r_{CCS}(i) = 5), \end{cases} \tag{2}$$

where c_R, c_G, and c_B denote the R, G, and B components of the original image, respectively, and $r_{CCS}(i)$ is a pseudo-random number for the i-th block, which is generated from key K_4.

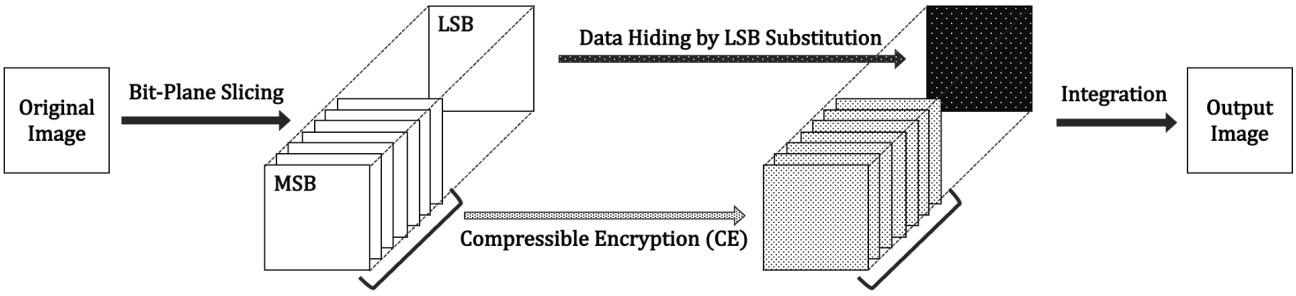


Fig. 4 Framework of proposed method procedure.

Figure 3 shows an EtC image, which is obtained according to the above procedure. It is noted that keys K_1, K_2, and K_3 are commonly used among the three color components in the fundamental CE [6]. On another hand, those keys are independently used among the three color components in the extended CE [7]. In the latter case, three different keys should be prepared for each color component, such as $K_{1,R}$, $K_{1,G}$, and $K_{1,B}$. Each color component is consequently proceeded by the different key. The proposed method can adopt either case.

III. Proposed Method

We propose a new framework that can conduct an encryption process and a data hiding process independently. In our method, we first perform bit-plane slicing and determine the fields for encryption and data hiding. Thus, the proposed method does not require any complex condition for either process. We applied the CE method [7] to encryption and the LSB substitution method [11] to data hiding. The CE method allows us to compress the output images efficiently. Further, the LSB substitution method suppresses degradation of the output images. The data hiding capacity is 3 bpp. We elaborate the encryption and data hiding procedures, and the user authority in the proposed method.

A. Procedure of Proposed Method

Figure 4 illustrates the encryption and data hiding procedures. We describe the detailed steps as follows.

- **Step 1 :** Slice the R, G, and B components of an original image *I* into bit planes, respectively.
- **Step 2-1:** Encrypt the upper 7 bits of the color components by CE.
- **Step 2-2:** Substitute the LSBs with a payload. Note that the payload should be encrypted beforehand.
- **Step 3 :** Integrate the bit planes and generate the output image I_{OUT}.

Bit-plane slicing allows us to entirely separate a data hiding field from an encryption field. Thus these processes can be independently conducted without interferences from one another. This means that we can flexibly embed/extract data and encrypt/decrypt an image regardless of the process sequence.

B. User authority

Here, we consider user authorities of output images. The types of user authorities are divided into three main classes: decryption only, data extraction only, and both decryption and data extraction. If a user has an authority to decrypt an output image without data extraction, the user can obtain the decryption-only image, namely, the marked image. In contrast, in case that a user has another authority to extract the payload from an output image without decryption, the hidden data and the data-extraction only image, i.e., the EtC image are attained. Finally, when the user can decrypt an output image and also extract the payload from the image, the user gains both the hidden data and the marked image. It is noted that the LSB substitution method is an irreversible data hiding algorithm, and thus the original image cannot be perfectly retrieved under either authority.

IV. Experimental Results and Discussion

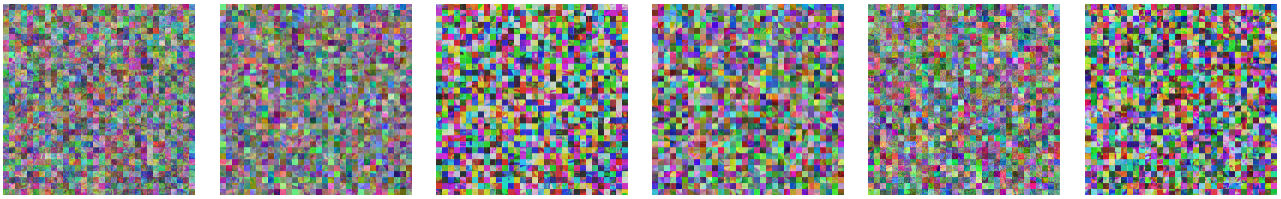
We evaluate the output images from the aspect of lossless compression performance using JPEG-LS [9] and JPEG 2000 [10], and assess the quality of the decryption-only images, i.e., the marked images. The six test images with 512×512 pixels, which are shown in Fig.5, from the signal and image processing institute (SIPI) database [24] were used in our experiments. Figure 6 exhibits the output images generated by the proposed method. In this experiment, the block size is 16×16 pixels and the color components are independently scrambled [7]. It is noted that we can adopt an arbitrary block size for encryption, but there is a tradeoff between the block size and robustness against attacks. We further discuss the robustness against some attacks.

A. Compression Performance

We confirm the lossless compression performance by JPEG-LS and JPEG 2000. Table I represents the compression ratio of the original and output images. It is clear that the output images can be suitably compressed even though the compression ratio of the output images is slightly deteriorated relative to that of the original images. Additionally, according to table

(a) Woodland Hills (b) Earth (c) Airplane (d) Lena (e) Mandrill (f) Sailboat

Fig. 5 Original images.



(a) Woodland Hills (b) Earth (c) Airplane (d) Lena (e) Mandrill (f) Sailboat

Fig. 6 Output images (block size: 16×16 pixels).

(a) Original image (b) Marked image, i.e., decryption-only image

Fig. 7 Resulting image by decryption only (Lena).

I, the compression ratio of JPEG 2000 is smaller than that of JPEG-LS. This is attributed to a discrete wavelet transform in the JPEG 2000 coding system, where the correlation is calculated from the wider spatial range of images than JPEG-LS.

Despite the high compression performance of our scheme, the output images obtained by the pixel-based encryption method [22] cannot be compressed by using international compression standards, such as JPEG-LS and JPEG 2000. On the contrary, their data amount may be increased by compression using those standards.

B. Image Quality

In the proposed method, the LSB substitution method is employed for data hiding; thus, the data hiding capacity is 3 bpp. Figure 7 illustrates the original and marked images. Note that marked images are equal to decryption-only images in our method. We evaluate the marked-image quality by using PSNR and MSSIM [25], as shown in Table II. The values of

TABLE I Lossless compression performance using JPEG-LS and JPEG 2000

	Compression ratio [%]			
	JPEG-LS		JPEG 2000	
	Original	Output	Original	Output
Woodland Hills	27.72	25.99	32.29	19.85
Earth	39.19	37.22	37.91	30.90
Airplane	50.67	46.62	51.91	34.46
Lena	43.31	40.40	43.38	31.48
Mandrill	22.86	20.64	24.65	13.18
Sailboat	34.60	31.83	33.27	22.41

TABLE II Marked-image quality

	PSNR [dB]	MSSIM
Woodland Hills	52.98	0.9990
Earth	53.00	0.9980
Airplane	52.98	0.9969
Lena	52.98	0.9971
Mandrill	52.96	0.9990
Sailboat	52.98	0.9979

PSNR and MSSIM were calculated for the luminance value of each marked image. We confirmed that the marked images, i.e., the decryption-only images containing the payload, retain high image quality.

C. Robustness against Attacks

Ciphertext-only attacks (COAs) are a type of attack models where an attacker is assumed to have access only to ciphertexts. Brute-force (BF) attacks and jigsaw puzzle solver (JPS) attacks are mentioned as COAs for CE. The former is a cryptanalytic attack where an attacker tries all possible combinations to decrypt a ciphertext correctly without an authority.

The robustness against BF attacks can be discussed by using key space. In our previous research, it has been demonstrated that the key space of the CE method is adequately large [6]–[8]. In contrast, the latter is another attack for image information that attempt to assemble an original image from multiple pieces by using the correlation among them. The EtC images consist of multiple blocks, and thus JPS attacks should be evaluated. We have already confirmed that the EtC images have sufficient robustness against JPS attacks [26]–[28]. The CE method is applied to upper 7 bits of an original image and LSBs are replaced with payload bits in the proposed method, and the proposed method retains the robustness against those two attacks compared to the original CE method.

Additionally, robustness against know-plaintext attack (KPA) and chosen-plaintext attack (CPA) is also discussed. CE is based on the premise that different encryption keys are prepared for each image and user; thus it is robust against KPA. Since CE is not a public key encryption method, encryption keys are not commonly disclosed. CE is consequently robust against CPA.

V. Conclusion

We propose a flexible data hiding method for encrypted images derived from the CE algorithm. The proposed method allows us to conduct an encryption process and a data hiding process independently by allocating bit planes of an original image to two fields. Thereby, any complex condition is not required in our method. We confirmed that the output images can be highly compressed by the international lossless image compression standards and the marked-image quality is significantly high. Additionally, the robustness against COAs was discussed for security analysis. Our future work involves achieving reversibility for data hiding to retrieve the original image after decryption and data extraction.

Acknowledgement

This work was partially supported by Grant-in-Aid for Research Activity start-up, No.19K23070, from the Japan Society for the Promotion Science.

References

[1] W. Liu, W. Zeng L. Dong, and Q. Yao, "Efficient compression of encrypted grayscale images," IEEE Trans. Image Process., vol. 19, no. 4, pp. 1097-1102, 2010.

[2] X. Zhang, "Lossy compression and iterative reconstruction for encrypted image," IEEE Trans. Inf. Forensics Security, vol. 6, no. 1, pp. 53-58, 2011.

[3] J. Zhou, X. Liu, O. C. Au, and Y. Y. Tang, "Designing an efficient image encryption-then-compression system via prediction error clustering and random permutation," IEEE Trans. Inf. Forensics Security, vol. 9, no.1, pp. 39-50, 2014.

[4] M. Kumar and A. Vaish, "An efficient encryption-then-compression technique for encrypted image using SVD," Digital Signal Processing, vol. 60, pp. 81-89, 2017.

[5] O. Watanabe, A. Uchida, T. Fukuhara, and H. Kiya, "An encryption-then-compression system for JPEG 2000 standard," in Proc. IEEE ICASSP, no. IVMSP-L4.1, pp. 1226-1230, 2015.

[6] K. Kurihara, M. Kikuchi, S. Imaizumi, S. Shiota, and H. Kiya, "An encryption-then-compression system for JPEG / Motion JPEG standard," IEICE Trans. Fundamentals, vol. E98-A, no. 11, pp. 2238-2245, 2015.

[7] S. Imaizumi and H. Kiya, "A block-permutation-based encryption scheme with independent processing of RGB components," IEICE Trans. Inf. & Syst., vol. E101-D, no. 12, pp. 3150-3157, 2018.

[8] K. Kurihara, S. Imaizumi, S. Shiota, and H. Kiya, "An encryption-then-compression system for lossless image compression standard," IEICE Trans. Inf. & Syst., vol. E100-D, no. 1, pp. 52-56, 2017.

[9] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: principles and standardization into JPEG-LS," IEEE Trans. Image Process., vol. 9, no. 8, pp. 1309-1324, 2000.

[10] "Information technology - JPEG 2000 image coding system - Part1: Core coding system". ISO/IEC IS-15444-1, 2004.

[11] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Techniques for data hiding," IBM Systems Journal, vol. 35, Nos. 3 & 4, pp. 313-336, 1996.

[12] V. M. Potdar, S. Han, and E. Chang, "A survey of digital image watermarking techniques," in Proc. IEEE Int. Conf. INDIN, pp. 709-716, 2005.

[13] C. C. Lai and C. C. Tsai, "Digital image watermarking using discrete wavelet transform and singular value decomposition," IEEE Trans. Instrum. Meas., vol. 59, no. 11, pp. 3060-3063, 2010.

[14] M. Ali, C. W. Ahn, and M. Pant, "A robust image watermarking technique using SVD and differential evolution in DCT domain," Opt. Int. J. Light and Electron Opt., vol. 125, no. 1, pp. 428-434, 2014.

[15] J. Tian, "Reversible data embedding using a difference expansion," IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 8, pp. 890-896, 2003.

[16] D. M. Thodi and J. J. Rodriguez, "Prediction-error based reversible watermarking," in Proc. IEEE Int. Conf. Image Process., vol. 3 pp. 1549-1552, 2004.

[17] Z. Ni, Y. Q. Shi, N. Ansari, and W. Su, "Reversible data hiding," IEEE Trans. Circuits Syst. Video Technol., vol. 16, no. 3, pp. 354-362, 2006.

[18] Y. Q. Shi, X. Li, X. Zhang, H. T. Wu, and B. Ma, "Reversible data hiding: Advances in the past two decades," IEEE Access., vol. 4, pp. 3210-3237, 2016.

[19] X. Zhang, "Reversible data hiding in encrypted image," IEEE Signal Process. Lett., vol. 18, no. 4, pp. 255-258, 2011.

[20] K. Ma, W. Zhang, X. Zhao, N. Yu, and F. Li, "Reversible data hiding in encrypted images by reserving room before encryption," IEEE Trans. Inf. Forensics Security, vol. 8, no. 3, pp. 553-562, 2013.

[21] X. Zhang, "Commutative reversible data hiding and encryption," Secur. Commun. Netw., vol. 6, no. 11, pp. 1396-1403, 2013.

[22] S. A. Parah, J. A. Sheikh, A. M. Hafiz, and G. M. Bhat, "Data hiding in scrambled images: A new double layer security data hiding technique," Computers and Electrical Engineering, vol. 40, no. 1, pp. 70-82, 2014.

[23] S. Imaizumi, Y. Izawa, R. Hirasawa, and H. Kiya, "A reversible data hiding method in compressible encrypted images," IEICE Trans. Fundamentals, vol. E103-A, no. 12, in press.

[24] [Online] Available: http://sipi.usc.edu/database/database.php

[25] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," IEEE Trans. Image Process., vol. 13, no. 4, pp. 600-612, 2014.

[26] T. Chuman, K. Kurihara, and H. Kiya, "On the security of block scrambling-based EtC systems against extended jigsaw puzzle solver attacks, "IEICE Trans. Inf. & Sys., vol. E101-D, no.1, pp. 37-44, 2018.

[27] W. Sirichotedumrong, T. Chuman, S. Imaizumi, H. Kiya, "Grayscale-based block scrambling image encryption for social network services, " in IEEE Int. Conf. on Multimedia and Expo, pp. 1-6, 2018.

[28] W. Sirichotedumrong and H. Kiya, "Grayscale-based block scrambling image encryption using YCbCr color space for encryption-then-compression systems," APSIPA Trans. Signal Inform. Process., e7, vol. 8, 2019.