Original Paper

Reversible Data Hiding for OpenEXR Images in Compressible Encrypted Domain

Noa KIKUCHI[a], Shoko IMAIZUMI[b]

Abstract: In this paper, we propose a novel method for reversible data hiding (RDH) for OpenEXR images in the compressible encrypted domain. There exist three main advantages in the proposed method. First, it introduces an encryption-then-compression system, which provides block-by-block encryption, to the encryption process. This leads to a high compression efficiency for marked encrypted images. In the encrypted domain, we embed a payload using an RDH method based on prediction error expansion. This makes the hiding capacity of our method high. The method further allows for flexible restoration patterns; we can decrypt a marked encrypted image without data extraction. In this case, a marked image will be obtained. In addition, both the encryption and data hiding processes are carried out while preserving the dynamic range of an original image. This also contributes both to compression efficiency and marked-image quality. In an experiment, we evaluate the hiding capacity, marked-image quality, and compression efficiency by using JPEG XT so as to clarify the effectiveness of our proposed method. Additionally, the security of the method is also discussed.

Key words: HDR image, reversible data hiding, encryption-then-compression system, JPEG XT

1. Introduction

Data hiding is one technique used for copyright protection. It embeds copyright information into the body data of an image without increasing the file size. Data hiding is classified into reversible and irreversible methods. Methods that can perfectly recover an original image by extracting a payload are called reversible data hiding (RDH) [1) – 3)], while methods that can never retrieve the original image even after data extraction are called irreversible data hiding. Additionally, in the context of privacy protection, RDH in encrypted images (RDH-EI) has been widely studied [4) - 7)]. In RDH-EI, an owner first encrypts an image and then sends the encrypted image to a third party, such as a service provider. The third party is then supposed to embed information such as management data or fingerprints into the encrypted image.

Motomura et al. proposed an RDH-EI method that achieved both a high hiding capacity and compression efficiency [7)]. This method introduced an encryption-then-compression (EtC) system [8)], which performs block-by-block encryption, for image encryption so that a marked encrypted image can be effectively compressed. They attained a high hiding capacity of up to 7.35 bpp in three color components. Additionally, their method offers flexibility in the recovery process; it can decrypt a marked encrypted image without data extraction. We focus on their method and call it the RDH-CEI method hereafter.

High dynamic range (HDR) images can represent a wider range of luminance compared with legacy standard dynamic range (SDR) images [9)]. HDR images have a large bit depth for capturing real-world luminance. Various file formats for HDR images have been developed, such as OpenEXR [10)], Radiance RGBE [11)], and LogLUV TIFF [12)]. In recent years, RDH-EI methods for HDR images have been increasingly explored [13) 14)]. Chia et al. proposed an RDH-EI method for OpenEXR images that achieved a high hiding capacity of up to 9.14 bpp [13)]. A marked encrypted image derived by this method, however, cannot be compressed efficiently. Further, a marked encrypted image can never be decrypted without extracting a payload. Thus, this method has a limitation that makes it not possible to provide users with flexible privileges for access control.

In this paper, we propose a novel RDH-EI method for OpenEXR images. Our method first encrypts a target image using the EtC system, which allows for image compression even after encryption. In the data hiding process, a prediction error expansion with a histogram shifting (PEE-HS) method [15)] is carried out in each block. Owing to the PEE-HS method, we can attain a high hiding capacity. This method also has flexibility in the restoration process; we can decrypt a marked encrypted image without data extraction. At this time, we can obtain a marked image in which the payload still remains. In addition, our method conducts each encryption and data hiding process while maintaining the dynamic range of the original image, thereby improving both the compression efficiency and marked-image quality. Additionally, we can further enhance both advantages by excluding negative-positive inversion in the encryp-

Received 5th February, 2024; Accepted 26th February, 2024 Published 7th March, 2024

[a] Graduate School of Science and Engineering, Chiba University
1-33 Yayoicho, Inage-ku, Chiba-shi, Chiba 263-8522, Japan
[b] Graduate School of Engineering, Chiba University
1-33 Yayoicho, Inage-ku, Chiba-shi, Chiba 263-8522, Japan

tion process. We evaluated the effectiveness of our method from four aspects: hiding capacity, marked-image quality, compression efficiency, and key space for security. Here, this paper is an extended article from our domestic conference paper [16] in terms of further discussions on both attack resistance and experimental analysis.

2. Preliminaries

In this section, we first explain the structure of OpenEXR image and then elaborate the RDH-CEI method [7] introduced into our method.

2.1 OpenEXR Image

OpenEXR is a format for HDR images that was developed by Industrial Light & Magic. In this format, each RGB component consists of 16-bit floating-point numbers. This allows images to have a broader dynamic range and color gamut compared with traditional SDR images, where each component is represented by 8 bits. Figure 1 shows the structure of 16-bit floating-point numbers. Each pixel value x is represented as

$$x = \begin{cases} (-1)^S \times (1 + \frac{M}{1024}) \times 2^{E-15}, & if\ E \neq 0 \\ (-1)^S \times (0 + \frac{M}{1024}) \times 2^{-14}, & if\ E = 0, \end{cases} \tag{1}$$

where S, E, and M are a sign, exponent, and mantissa, respectively.

2.2 RDH-CEI Method

We explain the RDH-CEI method [7] adopted for our proposed method. Figure 2 shows the flow of this method. First, a grayscale-based EtC system [8] is applied to an RGB image for encryption. After encryption, a payload is embedded using a PEE-HS method [15]. The RDH-CEI method uses the EtC system for encryption so that a marked encrypted image can be effectively compressed by international standards such as JPEG-LS [17]. Additionally, marked encrypted images can be decrypted without data extraction. This leads to deriving marked images that still contain a payload. This method achieved the highest hiding capacity among existing RDH-EI methods that can carry out a decryption process without data extraction. In the following, we will explain the encryption and data hiding procedures.

2.2.1 Image Encryption

In the encryption process, as shown in Fig. 3, the RGB components of an original image I are spatially combined to derive an 8-bit grayscale image I_G. Note that this figure shows horizontal combination, but an arbitrary direction can be chosen for combination. In this example, if the size of I is defined as $W \times H$ pixels, I_G will have $3W \times H$ pixels. I_G is then divided into blocks with $B \times B$ pixels and encrypted by using the EtC system so that an encrypted image I_{GE} can be obtained. The EtC system involves three processes: block-position scrambling, block rotation/inversion, and negative-positive inversion of pixel values within each block.

2.2.2 Data Hiding

After image encryption, arbitrary data called a payload is embedded into I_{GE}. In the RDH-CEI method, the PEE-HS method [15] is used for data hiding, but other methods are also available. It should be noted that the top-left pixel in each block is excluded from the following processes to ensure reversibility.

Step 1. Define a parameter for hiding-capacity control as L and modify the histogram so that the frequencies of 0 to L and 255-L to 255 turn out 0.

Step 2. Obtain prediction values $\hat{p}_{i,j}$ $(0 \leq i \leq B{-}1, 0 \leq j \leq \mathrm{B} - 1)$ for pixel values $p_{i,j}$ within each block:

$$\hat{p}_{i,j} = \begin{cases} min(p_{i-1,j}, p_{i,j-1}), & if\ p_{i-1,j-1} \geq \max(p_{i-1,j}, p_{i,j-1}) \\ \max(p_{i-1,j}, p_{i,j-1}), & if\ p_{i-1,j-1} \leq \min(p_{i-1,j}, p_{i,j-1}) \\ p_{i-1,j} + p_{i,j-1} - p_{i-1,j-1}, & otherwise. \end{cases} \tag{2}$$

Regarding $p_{0,j}$ and $p_{i,0}$, their prediction values $\hat{p}_{0,j}$ and $\hat{p}_{i,0}$ are

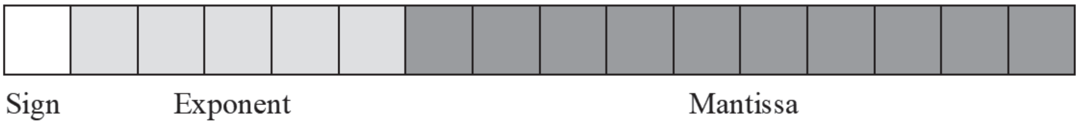


Fig. 1. Single channel structure of OpenEXR.

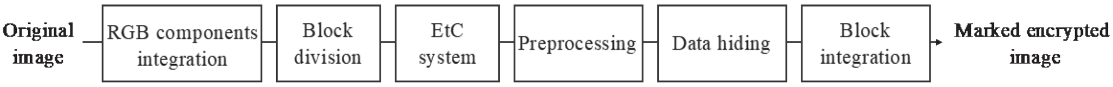


Fig. 2. Block diagram of RDH-CEI method [7].

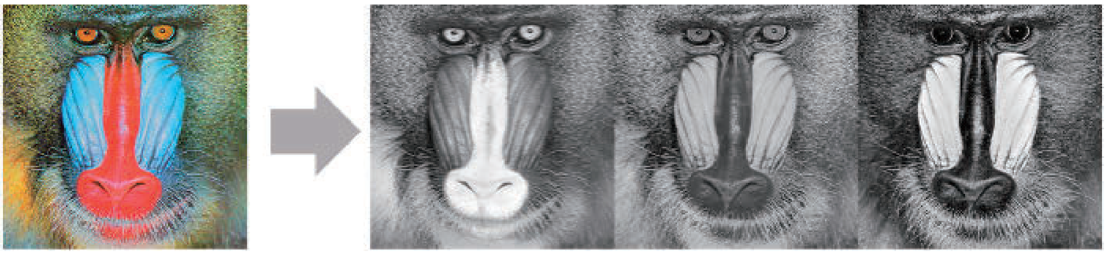


Fig. 3. Spatial combination of RGB component.

provided by

$$\hat{p}_{0,j} = p_{0,j-1}, \tag{3}$$

$$\hat{p}_{i,0} = p_{i-1,0}, \tag{4}$$

respectively.

Step 3. Derive prediction errors $e_{i,j}$:

$$e_{i,j} = \hat{p}_{i,j} - p_{i,j}. \tag{5}$$

Step 4. Explore a pair of adjacent bins E_s and E_l ($E_s < E_l$) with the highest sum of frequencies from a prediction-error histogram.

Step 5. As shown in Fig. 4, embed a payload ω into each $e_{i,j}$:

$$\tilde{e}_{i,j} = \begin{cases} e_{i,j} + 1, & if\ e_{i,j} > E_l \\ e_{i,j} + \omega, & if\ e_{i,j} = E_l \\ e_{i,j} - \omega, & if\ e_{i,j} = E_s \\ e_{i,j} - 1, & if\ e_{i,j} < E_s, \end{cases} \tag{6}$$

where $\tilde{e}_{i,j}$ is the marked prediction errors.

Step 6. Repeat Steps 4 and 5 for L-1 times.

Step 7. Obtain marked pixel values $\tilde{p}_{i,j}$ by

$$\tilde{p}_{i,j} = \hat{p}_{i,j} - \tilde{e}_{i,j}. \tag{7}$$

Step 8. Integrate all the blocks to derive a marked encrypted image $\tilde{I}_{GE}$.

The RDH-CEI method has attained a high hiding capacity of up to a total of 7.35 bpp for three color components and a high compression efficiency even after encryption. Additionally, since the encryption and data hiding processes are independent from each other, this method allows for flexible control of access privileges. Generally, users can decrypt a marked encrypted image only after extracting a payload. In this method, however, image decryption can be carried out without data extraction, allowing users to obtain a marked image where a payload remains.

3. Proposed Method

In this section, we propose a novel RDH-EI method that is extended for OpenEXR images. Figure 5 illustrates the flow of the method. The pixel values of an original image are treated as signed 15-bit values. First, we encrypt a target image using the EtC system [8] and then embed a payload with the PEE-HS method [15]. In the following, we describe the encryption, data hiding, restoration processes, and the features of the proposed method.

3.1 Image Encryption

In the encryption process, we encrypt a target image I using the EtC system so that an encrypted image I_E is derived. Here, we perform the following steps while preserving the original dynamic range.

Step 1-1. Divide I into blocks with B×B pixels.

Step 1-2. Scramble the positions of all blocks.

Step 1-3. Rotate and/or invert each block.

Step 1-4. Conduct negative-positive inversion on each block:

$$m'_k = \begin{cases} m_{max} + m_{min} - m_k, & if\ r(k) = 1 \\ m_k, & if\ r(k) = 0, \end{cases} \tag{8}$$

where m_k and m'_k are the original and encrypted pixels in the k-th block, and $r(k)$ is a random number generated by an encryption key. Note that m_{max} and m_{min} are the maximum and minimum pixel values, respectively.

Step 1-5. Shuffle the R, G, and B components within each block.

Our method can enhance the compression performance of encrypted images by preserving the dynamic range of the original images during negative-positive inversion. Additionally, when this inversion is excluded from the encryption process, both the compression performance and marked-image quality can be further improved. We discuss this in detail in 3.4 and evaluate the efficiency in Section 4.

3.2 Data Hiding

After image encryption, we embed a payload into I_E using the PEE-HS [15] method. This process is carried out while maintaining the dynamic range of I_E so that the compression efficiency and marked-image quality can be enhanced. As mentioned in 3.1, the top-left pixel of each divided block is excluded from the following steps for reversibility. Here, we define a parameter to control the

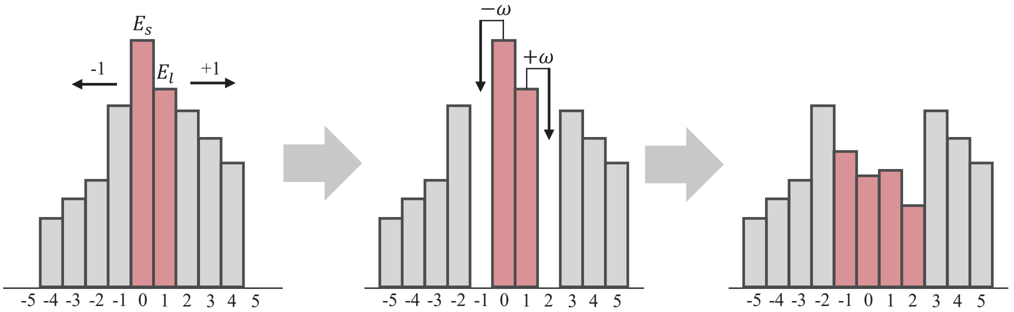


Fig. 4. Data hiding.

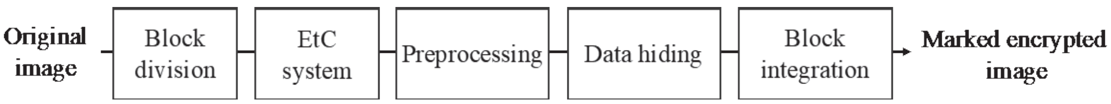


Fig. 5. Block diagram of proposed method.

hiding capacity as L.

Step 2-1. As shown in Fig. 6, modify the histogram so that the frequencies in the range of m_{min} to m_{min}-L and m_{max}-L to m_{max} turn out 0.

Step 2-2. By using Eq. (2), obtain prediction values $\hat{p}_{i,j}$ $(0 \le \mathrm{i} \le B–1, 0 \le j \le \mathrm{B} - 1)$ for pixel values $p_{i,j}$ within each block. In the case of $p_{0,j}$ and $p_{i,0}$, the prediction values $\hat{p}_{0,j}$ and $\hat{p}_{i,0}$ are given by Eqs. (3) and (4) , respectively.

Step 2-3. Derive prediction errors $e_{i,j}$ using Eq. (5).

Step 2-4. Search for the pair of adjacent bins E_s and E_l $(E_s < E_l)$ that has the highest sum of frequencies, from a prediction-error histogram.

Step 2-5. Embed a payload ω into each $e_{i,j}$ belonging to E_s or E_l. The modified prediction errors $\tilde{e}_{i,j}$ are given by Eq. (6).

Step 2-6. Obtain marked pixel values $\tilde{p}_{i,j}$ using Eq. (7).

Step 2-7. Repeat Steps 2-2 to 2-6 for L-1 times.

Step 2-8. Integrate all the blocks to derive a marked encrypted image $\tilde{I}_E$.

For ensuring reversibility, it is necessary to store the value of L and the value of E_s in the final iteration process. These values are stored by replacing the least significant bits (LSBs) of the top-left pixels of the divided blocks, which were excluded from the above steps. The other information required for reversibility consists of a location map for preprocessing, E_s in each iteration process, and the original bits of the above LSBs. The information should be embedded into the image together with the payload.

3.3 Restoration Process

Our method provides flexible restoration patterns owing to the independent implementation of the encryption and data hiding processes. Figure 7 depicts three possible restoration patterns with the proposed method.

First, Fig. 7 (a) shows a pattern where a payload is extracted, and an original image I is fully recovered. In this process, a marked encrypted image $\tilde{I}_E$ is divided into blocks with $B \times B$ pixels. The values of L and the value of E_s in the final iteration process are obtained from the LSBs of the top-left pixels of the divided blocks. The PEE-HS method [15] is then carried out in reverse order to extract a payload and the essential information for reversibility. By using the information, each pixel value is corrected. Finally, we decrypt the image and integrate all the divided blocks so that the original image I can be completely recovered.

Figure 7 (b) illustrates another pattern where $\tilde{I}_E$ is decrypted without data extraction. In this case, a user can obtain a marked image $\tilde{I}$ where a payload remains. This pattern assumes a scenario

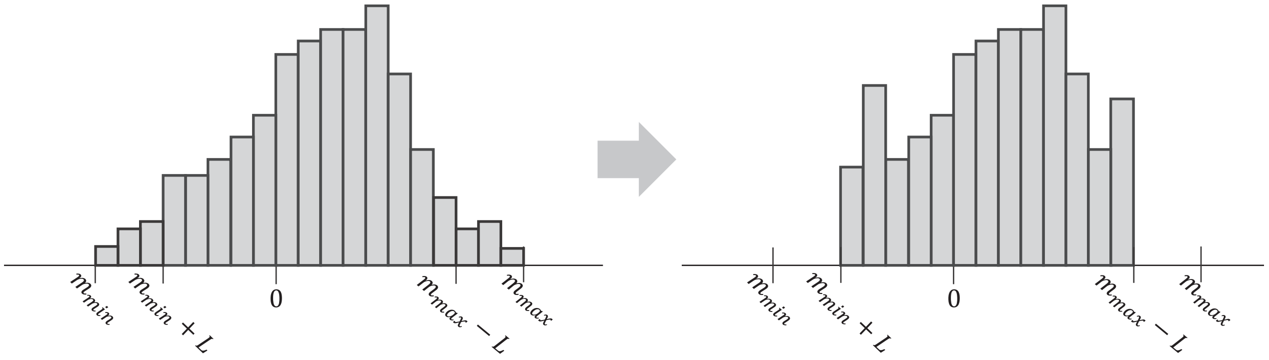
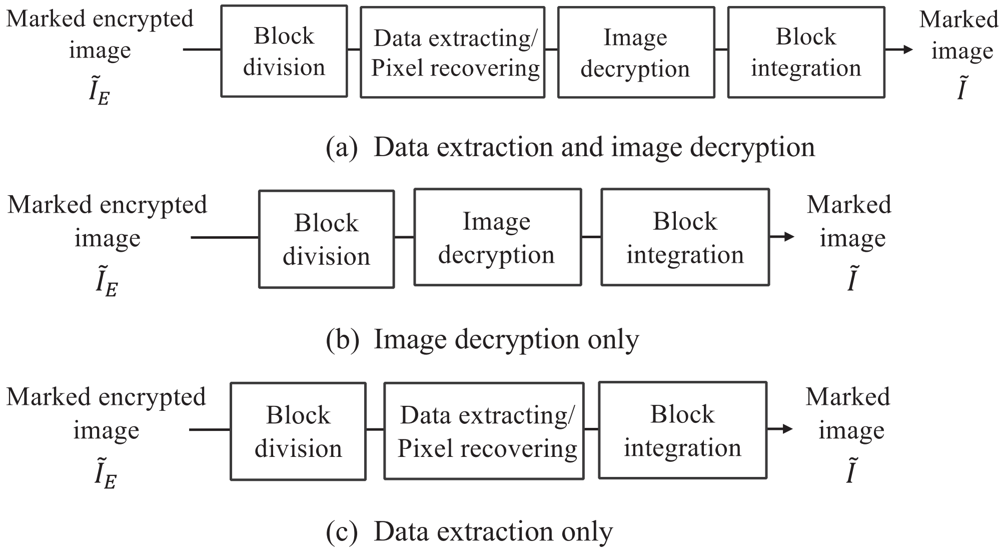


Fig. 6. Preprocessing.



(a) Data extraction and image decryption

(b) Image decryption only

(c) Data extraction only

Fig. 7. Three patterns of restoration process.

where the user has the privilege to access the image content only.

The third pattern is shown in Fig. 7 (c) , where only data extraction is available. A user can extract a payload from $\tilde{I}_E$, but the user is not allowed to access the image content.

Most previous RDH-EI methods have their restoration process in the patterns of Fig. 7 (a) and (c), but they cannot decrypt a marked encrypted image without data extraction as shown in Fig. 7 (b). In contrast, the proposed method can carry out image decryption with the payload remaining; in this case, we provide a marked image to each user.

3.4 Advantages of Proposed Methods

Here, we elaborate on the advantages of our method in comparison with two previous RDH-EI methods for HDR images [13) 14)]. Table 1 summarizes a feature comparison in terms of the hiding capacity, flexibility of restoration patterns, and compression performance for marked encrypted images.

Regarding the hiding capacity, although the image formats and test datasets are not uniform among the three methods, our method attained the highest capacity. The two previous methods use only several bits of the pixel value for data hiding, while the proposed method uses all bits of the pixel value. In addition, since the proposed method counts the compression performance of marked encrypted images, an RDH-CEI method [7)], which is highly compatible with the EtC system [8)], is adopted for the data hiding process. These efforts lead to an increase in embedding capacity.

Our method offers high flexibility in restoration patterns, particularly allowing for the decryption of marked encrypted images without data extraction. Chia et al.'s method [13)] carries out the encryption and data hiding processes interdependently. Thus, we can only decrypt an encrypted image for which a payload has already been extracted. Tsai et al.'s method [14)] first encrypts a target image and then replaces the pixel values of the exponent channel with a payload. In addition, this method embeds another payload into the other channels. In their method, image decryption requires extracting both payloads in advance. In contrast, as mentioned in 3.3, our method can decrypt a marked encrypted image without extracting the payload, allowing for more flexible access privileges based on user requirements.

Further, the marked encrypted images derived by our method are effectively compressed with international standards such as JPEG XT [18)]. This is because our encryption method, which is based on the EtC system, retains the correlation among pixels within a block even after encryption. In contrast, related studies encrypt an image on a pixel basis. This causes loss of correlation among adjacent pixels, leading to an increment in the overall entropy in an image. Consequently, it is difficult for the related studies to compress encrypted images.

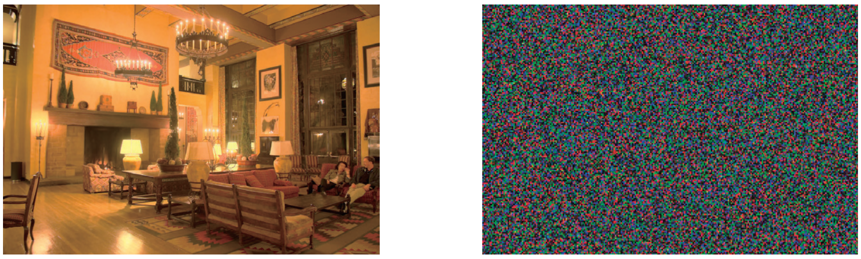
The proposed method preserves the dynamic range of original images during the encryption and data hiding processes, considering the compression efficiency and marked-image quality. Particularly, in the encryption process, if negative-positive inversion is carried out without maintaining the original dynamic range, the entropy of an encrypted image will increase. This significantly reduces compression efficiency. Moreover, if either the encryption or data hiding process affects the original dynamic range, there is a higher risk of introducing pixel values in the marked image that were not present in the original image, leading to color distortion. Our method thus carefully conducts the encryption and data hiding processes while maintaining the dynamic range of original images. Additionally, we can enhance both the compression efficiency and marked-image quality by excluding negative-positive inversion.

4. Experimental Results

In this section, we evaluate the proposed method in terms of marked-image quality, compression efficiency of marked encrypted images, and key space for security analysis. We assessed these perspectives both with and without negative-positive inversion to explore the impacts of this process. For the experiment, we used 20

Table 1. Feature comparison among RDH-EI methods for HDR images.

	Image format	Hiding capacity [bpp]	Restoration pattern	Compression of marked encrypted image
Prop.	OpenEXR	11.14	3	Effective
Chia et al. [13)]	OpenEXR	9.14	1	Ineffective
Tsai et al. [14)]	OpenEXR	7.03	2	Ineffective



(a) Original image

(b) Marked encrypted image

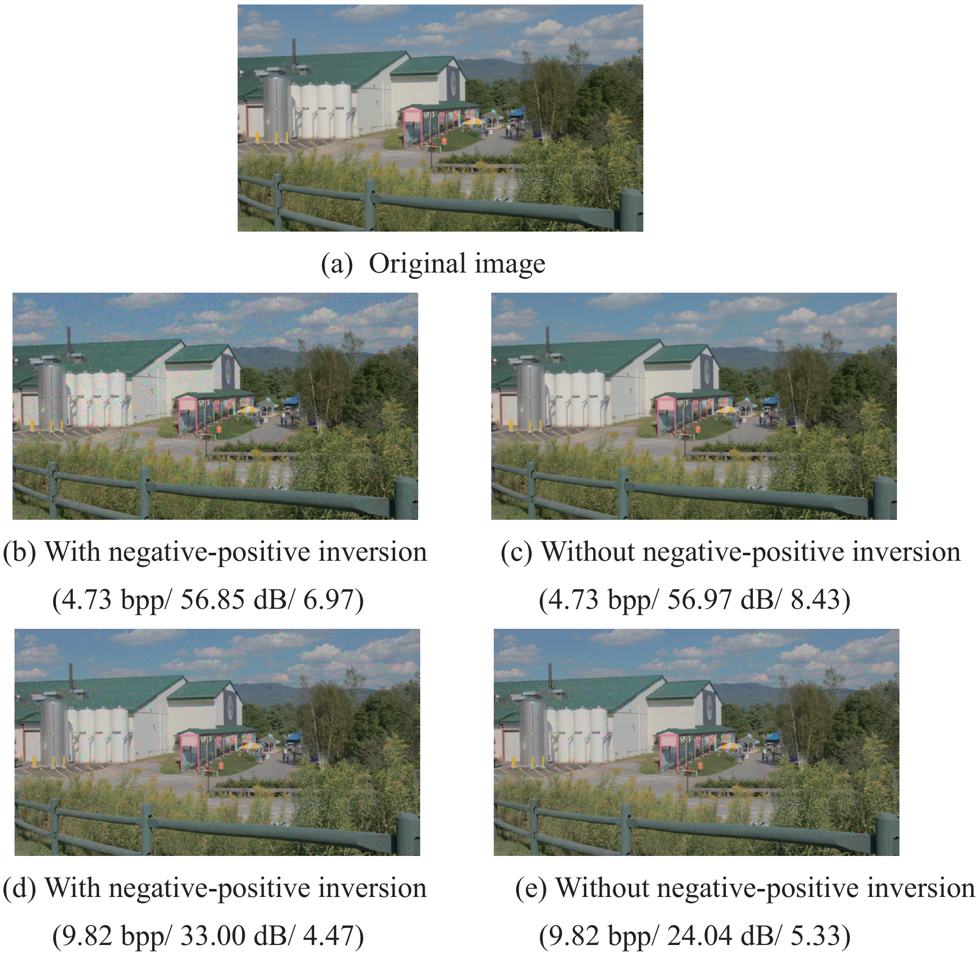
Fig. 8. Example of marked encrypted image.

images in the OpenEXR format from The HDR Photographic Survey [19]. In this paper, we define B=16 in the EtC system [8]. Figure 8 illustrates an example of a marked encrypted image derived by our method.

4.1 Hiding Capacity and Marked-Image Quality

Our method allows for decryption without data extraction, generating a marked image where a payload remains within the image. Here, we evaluate the marked-image quality, which is related to the payload amount. Figure 9 shows examples of marked images obtained by our method. Figures 9 (b) and (c) are the results when the payload was 4.73 bpp, and Figs. 9 (d) and (e) are those for 9.82 bpp. As shown in these figures, the quality of the marked images was visually maintained even at a high hiding capacity. When negative-positive inversion was applied, and a large amount of information was embedded, however, some block-wise distortions could be observed. We can see these distortions clearly in Fig. 10 (a) , which is an enlarged view of the background regions in Fig. 9 (d). However, as can be seen from Fig. 10 (b) , such distortions can be suppressed by excluding negative-positive inversion.

For image quality evaluation, we introduce LogPSNR [20] and HDR-VDP-3 [21]. LogPSNR is an extension of PSNR for HDR images, which is given by



(a) Original image

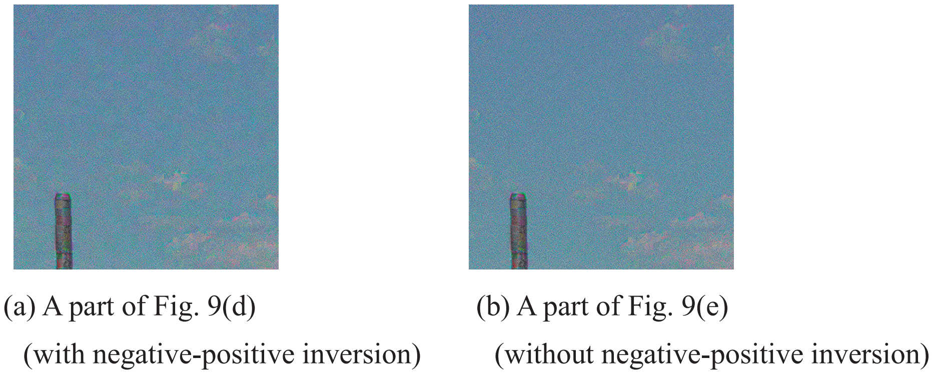
(b) With negative-positive inversion
(4.73 bpp/ 56.85 dB/ 6.97)

(c) Without negative-positive inversion
(4.73 bpp/ 56.97 dB/ 8.43)

(d) With negative-positive inversion
(9.82 bpp/ 33.00 dB/ 4.47)

(e) Without negative-positive inversion
(9.82 bpp/ 24.04 dB/ 5.33)

Fig. 9. Examples of marked images (hiding capacity [bpp]/ LogPSNR [dB]/ HDR-VDP (Q)) .



(a) A part of Fig. 9(d)
(with negative-positive inversion)

(b) A part of Fig. 9(e)
(without negative-positive inversion)

Fig. 10. Enlarged views of background regions in Figs. 9 (d) and (e).

$$LogPSNR = 20\log_{10}\left(\frac{\log_{10} MAX}{\sqrt{\Delta(A_1,A_2)}}\right), \tag{9}$$

$$\Delta(A_1,A_2) = \frac{1}{M\times N}\sum_{pixels}(\log_{10}A_1 - \log_{10}A_2)^2, \tag{10}$$

where M and N are image dimensions, and A_1 and A_2 are pixel values of an original and marked images, respectively. In our experiment, we set MAX=2^{16}. Additionally, we use HDR-VDP (Q) from HDR-VDP-3 for quality assessment, where the maximum value of 10 indicates that the marked image is identical to the original image, and a lower value indicates a lower image quality. Figure 11 shows the mean values of LogPSNR and HDR-VDP (Q) for the marked images with and without negative-positive inversion. As shown in this figure, the quality of the marked images with our method was high. This is attributed to the encryption and data hiding processes which preserve the dynamic range of original images, as discussed in 3.4. From Fig. 11 (a) , the differences in LogPSNR with and without negative-positive inversion are negligible. This metric is based on the mean squared error of pixels, so the values of LogPSNR for both cases are analogous to each other at certain amounts of hiding capacity. In contrast, HDR-VDP (Q) considers human visual perception; Fig. 11 (b) indicates significant differences between the two cases. When negative-positive inversion is excluded, we can visually suppress block-wise distortions, obtaining a higher HDR-VDP (Q) score.

4.2 Hiding Capacity and Compression Efficiency

We evaluate the compression efficiency of marked encrypted images using lossless compression with JPEG XT [18] Profile C. The compression efficiency is defined as the rate of reduction of the file size by compression. Here, the original file size of each image is 48 bpp, and the compression efficiency is calculated by

$$Compression\ efficiency = \left(1 - \frac{Compressed\ file\ size\ [bpp]}{48\ bpp}\right)\times 100\ [\%]. \tag{11}$$

Figure 12 depicts the compression efficiency relative to the amount of a payload. From this figure, compression using JPEG XT is effective for marked encrypted images generated by our method. This is because we perform image encryption with the EtC system with the aim of preserving the dynamic range of original images. In the case without negative-positive inversion, a higher compression efficiency can be attained than the case with negative-positive inversion. Negative-positive inversion increases the entropy of encrypted images, leading to an expansion in the data volume of the residual information in JPEG XT. Here, the residual information is necessary for reconstructing an HDR image. Consequently, the inversion decreases the compression efficiency of marked encrypted images.

4.3 Security Analysis

We evaluate the key spaces of our method against brute-force attacks. The key space is defined as the total number of possible pat-

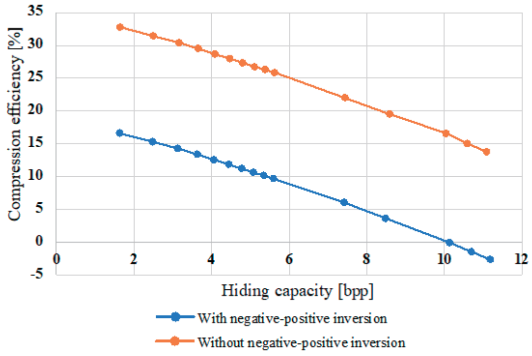
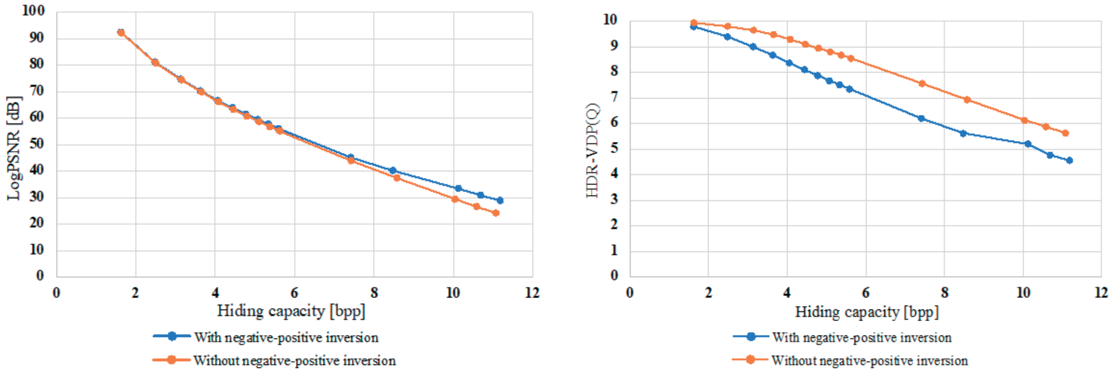


Fig. 12. Compression efficiency of marked encrypted images.



(a) LogPSNR (b) HDR-VDP(Q)

Fig. 11. Marked-image quality.

Table 2. Key spaces against brute-force attacks.

	Position scrambling	Rotation and inversion	Negative-positive inversion	RGB shuffling	Total
With negative-positive inversion	$(K!)^3$	8^{3K}	2^{3K}	6^K	$(K!)^3\times(24{,}576)^K$
Without negative-positive inversion	$(K!)^3$	8^{3K}	-	6^K	$(K!)^3\times(3{,}072)^K$

terns for the encryption key. Table 2 shows key spaces in the case with and without negative-positive inversion. In this table, the number of divided blocks in the encryption process is K. Even though we exclude negative-positive inversion, the key space is approximately given as $5.47\times10^{773,776}$ for an image with 4,288×2,848 pixels. This number indicates that the proposed method offers high security against brute-force attacks in terms of computational costs. However, the exclusion of negative-positive inversion has been reported to reduce the resistance against attacks using the jigsaw puzzle solver [22]. In our proposed method, we can thus take into account the exclusion of negative-positive inversion only when we prioritize marked-image quality and compression efficiency over security.

5. Conclusion

In this paper, we proposed a novel RDH-EI method for OpenEXR images with three main advantages. First, marked encrypted images can be effectively compressed by JPEG XT owing to the introduction of an EtC system for image encryption. Since we carefully reviewed the data hiding process, we also achieved a high hiding capacity of up to 11.14 bpp. Our method further offers high flexibility in restoration patterns, allowing for image decryption without data extraction. The proposed method additionally improved both compression efficiency and marked-image quality by preserving the dynamic range of original images during the whole process; this is effective for both compression efficiency and marked-image quality. Compared with other RDH-EI methods for HDR images, our method surpassed them in terms of hiding capacity, compression performance, and flexibility of access privileges. The key space was also discussed for security. We confirmed that the key space was large enough and that the proposed method is secure against brute-force attacks.

References

1) Y.-Q. Shi, X. Li, X. Zhang, H.-T. Wu, and B. Ma, Reversible data hiding: Advances in the past two decades, IEEE Access, **4** (2016).
2) Z. Ni, Y.-Q. Shi, N. Ansari, and W. Su, Reversible data hiding, IEEE Trans. Circuits Syst. Video Technol., **16**, 3 (2006).
3) W. He, G. Xiong, and Y. Wang, Reversible data hiding based on adaptive multiple histogram modification, IEEE. Trans. Inf. Forensics Secur., **16** (2021).
4) P. Puteaux, S. Ong, K. Wong, and W. Puech, A survey of reversible data hiding in encrypted images-The first 12 years, J. Vis. Commun. Image Represent., **77** (2021).
5) X. Zhang, Reversible data hiding in encrypted image, IEEE Signal Process. Lett., **18**, 4 (2011).
6) E. Arai and S. Imaizumi, High-Capacity Reversible Data Hiding in Encrypted Images with Flexible Restoration, J. Imaging, **8**, 176 (2022).
7) R. Motomura, S. Imaizumi, and H. Kiya, Reversible Data Hiding in Compressible Encrypted Images with Capacity Enhancement, APSIPA Trans. Signal Inf. Process., **12**, 1 (2023).
8) T. Chuman, W. Sirichotedumrong, and H. Kiya, Encryption-then-compression systems using grayscale-based image encryption for JPEG images, IEEE Trans. Inf. Forensics Secur., **14**, 6 (2019).
9) E. Reinhard, G. Ward, S. Pattanaik, P. Debevec, W. Heidrich, and K. Myszkowski, "High Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting", 2nd ed., Morgan Kaufmann, CA, USA, 2010.
10) Technical Introduction to OpenEXR. Available online: https://openexr.com/en/latest/TechnicalIntroduction.html (accessed on 30 November 2023).
11) G. Ward, "Real Pixels", Graphics Gems II, by J. Arvo, Academic Press, CA, USA, 1991, p.80-83.
12) G. W. Larson, LogLuv Encoding for Full-Gamut, High-Dynamic Range Images, Journal of Graphics Tools, **3**, 1 (2012).
13) K. Chia, K. Wong, and J.-L. Dugelay, Data Hiding in Perceptually Masked OpenEXR Image, Proc. IEEE MMSP, 1-6 (2019).
14) Y.-Y. Tsai, H.-L. Liu, P.-L. Kuo, and C.-S. Chan, Extending Multi-MSB Prediction and Huffman Coding for Reversible Data Hiding in Encrypted HDR Images, IEEE Access, **10** (2022).
15) T. Luo, G. Jiang, M. Yu, F. Shao, and Z. Peng, Disparity based stereo image reversible data hiding, Proc. IEEE ICIP, 5492-5496 (2014).
16) 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., **9**, 8 (2000).
17) T. Richter, A. Artusi, and T. Ebrahimi, JPEG XT: A New Family of JPEG Backward Compatible Standards, IEEE MultiMedia, **23**, 3 (2016).
18) The HDR Photographic Survey. Available online: http://markfairchild.org/HDRPS/HDRthumbs.html (accessed on 30 November 2023).
19) R. Mukherjee, K. Debattista, T. Bashford-Rogers, P. Vangorp, R. Mantiuk, M. Bessa, B. Waterfield, and A. Chalmers, Objective and subjective evaluation of High Dynamic Range video compression, Signal Process.: Image Commun., **47** (2016).
20) R. K. Mantiuk, D. Hammou, and P. Hanji, HDR-VDP3: A multi-metric for predicting image differences, quality and contrast distortions in high dynamic range and regular content, arXiv preprint arXiv:2304.13625 (2023).
21) T. Chuman, K. Kurihara, and H. Kiya, On the Security of Block Scrambling-Based EtC Systems against Jigsaw Puzzle Solver Attacks, IEICE Trans. Inf. & Syst., **E101-D**, 1 (2018).
22) N. Kikuchi and S. Imaizumi, A Reversible Data Hiding Method in Compressible Encrypted Domain for OpenEXR Images, IEICE Tech. Rep., **123**, 332 (2024) , in Japanese.