LETTER *Special Section on Enriched Multimedia —New Technology Trends in Creation, Utilization and Protection of Multimedia Information—*

An Encryption-then-Compression System for Lossless Image Compression Standards

Kenta KURIHARA[†], ***Nonmember***, **Shoko IMAIZUMI**[††], **Sayaka SHIOTA**[†], ***Members***,
and* Hitoshi KIYA**[†a)], ***Fellow

SUMMARY In many multimedia applications, image encryption has to be conducted prior to image compression. This letter proposes an Encryption-then-Compression system using JPEG XR/JPEG-LS friendly perceptual encryption method, which enables to be conducted prior to the JPEG XR/JPEG-LS standard used as an international standard lossless compression method. The proposed encryption scheme can provides approximately the same compression performance as that of the lossless compression without any encryption. It is also shown that the proposed system consists of four block-based encryption steps, and provides a reasonably high level of security. Existing conventional encryption methods have not been designed for international lossless compression standards, but for the first time this letter focuses on applying the standards.

key words: *Encryption-then-Compression system, lossless compression, international standard, JPEG XR, JPEG-LS*

1. Introduction

With the wide/rapid spread of distributed systems for information processing, such as cloud computing and social networks, not only transmission but also processing is done on the public Internet, and thus contents are transmitted over an insecure bandwidth-constrained communication channel [1], [2]. In the meantime, a lot of studies on secure, efficient and flexible communications have been reported. For securing multimedia data, full encryption with a state-of-the-art cipher (like RSA, AES, etc.) is the most secure option. However, many multimedia applications have been seeking a trade-off in security to enable other requirements, e.g., low processing demands, retaining bitstream compliance, and signal processing in the encrypted domain. Because of this situation, a lot of perceptual encryptions have been studied as one of schemes for achieving the trade-off.

In this letter, we focus on an Encryption-then-Compression (ETC) system [3] under the use of two international lossless coding standards, although the traditional way of securely transmit images is to use a Compression-then-Encryption system. Recently, the JPEG committee has started to standardize a new work item, referred to as JPEG Privacy [4], in which secure transmission between network servers in cloud computing and social networks is supposed as one of the technical requirements of the JPEG Privacy.

However, most of the conventional works for ETC systems have no compatibility with the international standards, e.g., JPEG, JPEG XR, JPEG 2000, etc. [3]. Also, a number of perceptual encryption schemes have been studied for the international standards, but they do not correspond to ETC systems, except for the articles regarding JPEG and JPEG 2000 [5]–[7]. However, the ETC systems [5]–[7], which have been studied for the international standard, do not correspond to lossless coding. In other words, they have not focused on lossless coding methods. It has not been clarified that whether they are useful for international standard lossless coding methods, and how the parameters should be chosen for the encryption. Because of such situations, in this letter, an ETC system is considered under the use of international standard lossless coding methods such as JPEG XR and JPEG-LS for the first time.

2. ETC System and International Standards

2.1 ETC System

In this letter, we focus on image compression systems in the encrypted domain, namely ETC systems as illustrated in Fig. 1, in which a content owner Alice wants to securely and efficiently transmit an image I to a recipient Bob, via an untrusted channel provider Charlie. In particular, the use of two international standards i.e. the lossless mode of JPEG XR and JPEG-LS, is supposed as a lossless compression method.

2.2 JPEG XR and JPEG-LS

JPEG XR [8] and JPEG-LS [9] are image coding standards from the JPEG committee. They allow lossless coding for still images.

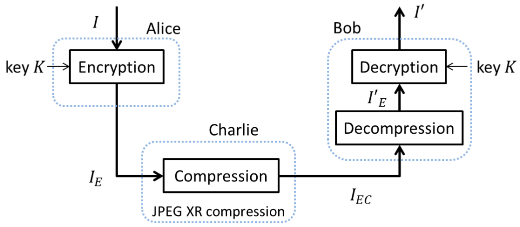


Fig. 1 Encryption-then-Compression system

Manuscript received March 28, 2016.
Manuscript revised July 29, 2016.
Manuscript publicized October 7, 2016.

[†] The authors are with the Graduate School of System Design, Tokyo Metropolitan University, Hino-shi, 191–0065 Japan.

[††] The author is with the Graduate School of Advanced Integration Science, Chiba University, Chiba-shi, 263–8522 Japan.

a) E-mail: kiya@tmu.ac.jp

DOI: 10.1587/transinf.2016MUL0002

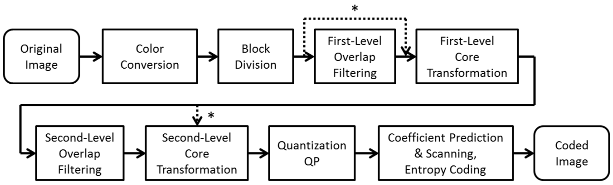

Original Image
Color Conversion
Block Division
First-Level Overlap Filtering
*
First-Level Core Transformation
Second-Level Overlap Filtering
Second-Level Core Transformation
*
Quantization QP
Coefficient Prediction & Scanning, Entropy Coding
Coded Image

Fig. 2 JPEG XR coding (*: There are three modes)

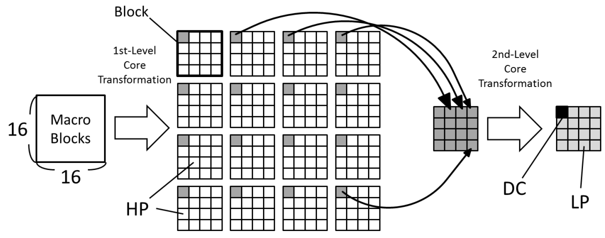


Fig. 3 Lapped biorthogonal transform

The block diagram of JPEG XR encoding is illustrated in Fig. 2. The encoding consists of the following basic steps:

1) Performing a color conversion to YUV space and subsampling chroma components.
2) Dividing an image into non-overlapped consecutive 16×16 blocks, called *macro block*, and then each macro block into consecutive 4 × 4 blocks, called *block* (see Fig. 3).
3) Applying two basic operators (referred to as a lapped biorthogonal transform) i.e. optional overlap filtering to the blocks and core transform, where the operators are hierarchically executed twice shown in Fig. 3.
4) Applying a coefficient quantization approach controlled by quantization parameters (QPs).
5) Executing adaptive coefficient scanning to convert the two-dimensional array transform coefficients within a block into a one-dimensional vector to be encoded. Finally, the coefficients are entropy encoded.

In step 3), one temporal DC (Direct Current) coefficient and 15 HP (High Pass) coefficients are obtained for each block by the 1st-level core transform, and 16 temporal DC coefficients are gathered from each macro block as shown in Fig. 3. The 2nd-level core transform is then applied to them. As a result, one DC coefficient, 15 LP (Low Pass) coefficients and 16 HP coefficients are calculated for each macro block.

Thus, it can be said that the JPEG XR standard is a block based coding method. In the lossless coding, subsampling is not applied to chroma components U and V, and QPs are equal to one.

On the other hand, the JPEG-LS standard does not have block-based processing. It encodes an image using the correlation among adjacent pixels.

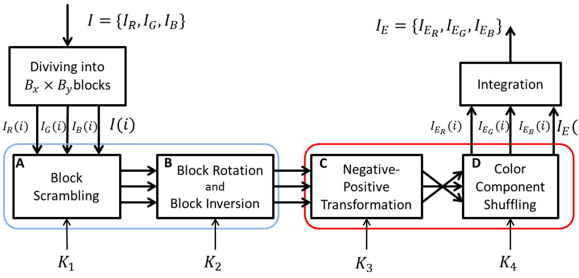
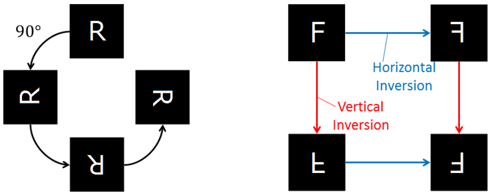

$I = \{I_R, I_G, I_B\}$
Diviving into $B_x \times B_y$ blocks
$I_R(i)$
$I_G(i)$
$I_B(i)$
$I(i)$
A
Block Scrambling
B
Block Rotation and Block Inversion
C
Negative-Positive Transformation
D
Color Component Shuffling
K_1
K_2
K_3
K_4
$I_{E_R}(i)$
$I_{E_G}(i)$
$I_{E_B}(i)$
$I_E(i)$
Integration
$I_E = \{I_{E_R}, I_{E_G}, I_{E_B}\}$

Fig. 4 Four block-based steps for encryption



(a) Block rotation

(b) Block inversion

Fig. 5 Block rotation and inversion

3. Proposed Scheme

3.1 Block-Based Encryption Scheme

A block-based permutation method was applied on the transformed domain such as the wavelet transform [5] as another method. However, it is not suitable for lossless coding methods. Therefore, we focus on the permutation methods on the spatial domain.

We investigate a block-based encryption scheme, in which an image with $M \times N$ pixels is divided into non-overlapped consecutive blocks with $B_x \times B_y$ pixels. The proposed system consists of four block-based steps as illustrated in Fig. 4. The procedure of performing the proposed image encryption is given as follows:

Step1: Divide each color component of a color image $I = \{I_R, I_G, I_B\}$ into $B_x \times B_y$ blocks respectively. The i-th block image is defined as $I(i) = \{I_R(i), I_G(i), I_B(i)\}$ where $i = 1, 2, \cdots$ is a block number.

Step2: Permute randomly the divided blocks using a random integer generated by a secret key K_1, where K_1 is commonly used for all color components.

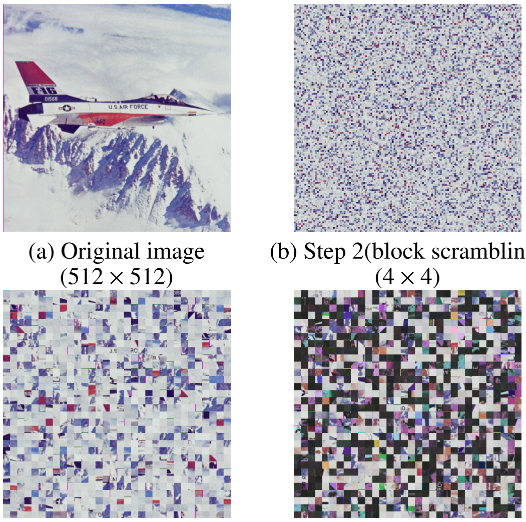
Step3: Rotate and invert randomly each $B_x \times B_y$ block (see Fig. 5) using a random integer generated by a key K_2, where K_2 is commonly used for all color components as well.

Step4: Apply the negative-positive transformation to each $B_x \times B_y$ block using a random binary integer generated by a key K_3, where K_3 is commonly used for all color components. In this step, a transformed pixel value in the i-th block, p' is computed by

$$p' = \begin{cases} p & (r(i) = 0) \\ 255 - p & (r(i) = 1) \end{cases} \quad (1)$$

Table 1 Permutation of color components with random integer

Random Integer	R	G	B
0	R	B	G
1	G	R	B
2	G	B	R
3	B	R	G
4	B	G	R



(a) Original image (512 × 512)

(b) Step 2(block scrambling) (4 × 4)

(c) Step 2 (block scrambling) (16 × 16)

(d) Step 6 (proposed) (16 × 16)

Fig. 6 Encrypted images ($B_x \times B_y$)

where p is the pixel value of an original image with 8 bpp, $r(i)$ is a random binary integer generated by K_3.

Step5: Shuffle three color components in each $B_x \times B_y$ block (the color component shuffling) using a random quinary integer generated by a key K_4. Table 1 shows the permutation of color components corresponding to the random quinary integer.

Step6: Generate the encrypted image by integrating the transformed block images.

Figures 6 (b) and 6 (c) illustrate permuted images of the image in Fig. 6 (a) with the block sizes 4 × 4 and 16 × 16 respectively. Figure 6 (d) also shows an encrypted image by using the above four steps.

3.2 Consideration on Block Sizes

Since the proposed scheme is based on a block-based operation, encrypted images still have the correlation among pixels in each block. Therefore, the proposed scheme is expected to be available for a number of compression standards. Especially, in JPEG XR, which is a block-based coding method, the proposed scheme is expected to have high compression performance by selecting an appropriate block size.

JPEG XR independently operates each macro block with the size of 16 × 16 in an image, as shown in Fig. 3. Therefore, when the block size $B_x \times B_y$ is $16N \times 16N$ where N is a natural number, the proposed encryption scheme is expected to provide the almost same compression performance as that of images without any encryption.

On the other hand, JPEG-LS is not a block-based coding, and it encodes an image using the correlation among adjacent pixels. Although the proposed encryption scheme is a block based one, it is expected that the correlation among adjacent pixels in each block can be preserved even when images are encrypted. In Sect. 5, it will be shown that the proposed scheme has a slightly lower compression performance than that of JPEG XR. We will also illustrate that the compression performance can be controlled by selecting a block size. In addition, it will be confirmed that the block size is not limited to a multiple of 16 for compression by JPEG LS.

4. Security Analysis

There are several kinds of attack on encryption, such as the brute-force attack, the differential attack, the statistical attack, and so on. In this letter, we evaluate the safety of the proposed system with its key space, assuming that an attacker performs the brute-force attack. In the proposed scheme, the key space is determined by the number of divided blocks. If an original image with $M \times N$ pixels is divided into blocks with $B_x \times B_y$ pixels, the number of the blocks L is computed by

$$L = \lfloor \frac{M}{B_x} \rfloor \times \lfloor \frac{N}{B_y} \rfloor. \tag{2}$$

In the block scrambling, the key space N_B, which is the number of permutation of L blocks, is given by

$$N_B = {}_L P_L = L!. \tag{3}$$

Similarly, when the key spaces of other encryption steps are given as

$$N_R = 8^L, \ N_N = 2^L, \ N_C = ({}_3P_3 - 1)^L = 5^L \tag{4}$$

where N_R, N_N and N_C are the key spaces of the encryption combining the block rotation and the block inversion, the negative-positive transformation and the color component shuffling respectively. Even N_N, which is the smallest key space in the above key spaces, is larger than 2^{256} when $L > 256$, i.e. the key space of the proposed scheme is larger than that of the 256-bit key, when the divided image has more than 256 blocks at least.

Consequently, the key space of encrypted images by using all the proposed encryption steps, N_A, is represented by

$$\begin{aligned} N_A &= N_B \cdot N_R \cdot N_N \cdot N_C \\ &= L! \cdot 8^L \cdot 2^L \cdot 5^L. \end{aligned} \tag{5}$$

Because the proposed encryption scheme permutes block images to generate a scrambled image, an attacker may restore the encrypted image to the original image using a tool such as the jigsaw puzzle solver [10], [11] or the

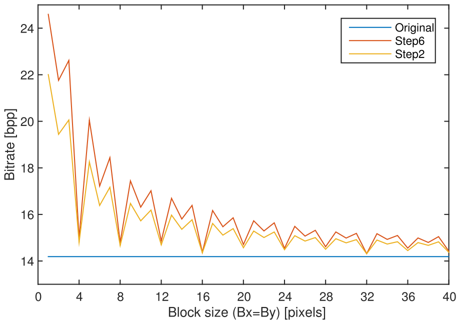


Fig. 7 Bit rate-Block bize curves (Lena, JPEG XR)

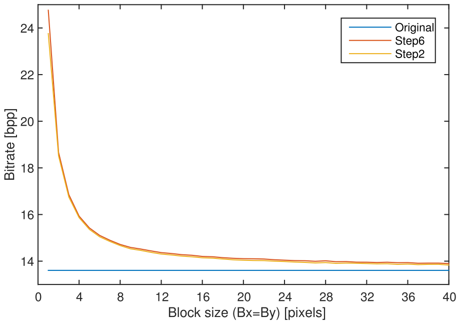


Fig. 8 Bit rate-Block bize curves (Lena, JPEG-LS)

deshredder [12]. The tools try to restore an original image from a scrambled image by using its edge information, color, and mean of pixel values. On the other hand, the proposed scheme can changed mean values, color values and edge directions in each block by carrying out four encryption steps. Therefore, the attacker will fail or take a huge amount of time to restore the encrypted image.

5. Simulation

We evaluate the effectiveness of the proposed scheme by a number of simulations. Five images were reversibly compressed by the JPEG XR standard or the JPEG-LS standard. Airplane, Lena, Mandrill, Milkdrop and Pepper (512 × 512, RGB24bpp) were used as test images.

Figures 7 and 8 show the relationship between bitrates and block sizes for the reference image, Lena, compressed by JPEG XR and JPEG-LS, respectively. In Fig. 7, 16 × 16 or 32 × 32 is chosen as a block size, which is the proposed size, the compression performance of encrypted images is almost same as that of the original image without any encryption. Also when 4 × 4 or its multiple integer is chosen, the encrypted image has a good compression performance due to two core transform operations shown in Fig. 3. On the other hand, in Fig. 8, a larger block size offers a better performance. In addition, it is also confirmed that the size 16 × 16, which was proposed for the JPEG XR standard, provides a reasonable performance.

Tables 2 and 3 show the compression results for reference images and ones encrypted by the proposed scheme, where the block sizes are 16 × 16 in Table 2 and 4 × 4 in Table 3, respectively. When the block size is equal to 16 × 16, which is the same size as macro blocks of the JPEG XR, the performance degradation is just one to two percent under the criteria of the number of bits per pixel (bpp), compared to the case without any encryption. The JPEG-LS also provides a similar performance trend to the JPEG XR. On the other hand, when the block size is equal to 4×4, which is the same size as blocks, the compression performance becomes a little lower than the case with the size 16 × 16, but the key space increases due to a larger number of blocks. There is the trade-off relation between the compression performance and the security level, and thus a user should choose a suitable block size depending on the application scenario.

Table 2 Compression results [bpp] ($B_x \times B_y = 16 \times 16$)

	JPEG XR		JPEG-LS	
	Original	Encrypted	Original	Encrypted
Mandrill	18.45	18.78	18.52	19.02
Lena	14.18	14.37	13.60	14.20
Airplane	12.36	12.71	11.84	12.55
Pepper	15.39	15.46	14.27	14.82
Milkdrop	12.18	12.28	10.70	11.32
Average	14.51	14.72	13.79	14.38
Ratio	1	1.015	1	1.043

Table 3 Compression results [bpp] ($B_x \times B_y = 4 \times 4$)

	JPEG XR		JPEG-LS	
	Original	Encrypted	Original	Encrypted
Mandrill	18.45	19.09	18.52	20.19
Lena	14.18	14.98	13.60	15.94
Airplane	12.36	13.16	11.84	14.89
Pepper	15.39	16.02	14.27	16.52
Milkdrop	12.18	13.16	10.70	13.63
Average	14.51	15.28	13.79	16.23
Ratio	1	1.05	1	1.177

6. Conclusion

This letter proposed an efficient ETC system for the international standard lossless compression methods. The proposed scheme consists of the four block-based encryption steps considering the block-based coding method (JPEG XR) and the pixel correlation-based coding method (JPEG-LS). The block-based encryption processing can preserve correlation among adjacent pixels within a block, so that the compression performance of an encrypted image is almost same as that of the original image, choosing an appropriate block size. The experimental results demonstrated that the proposed system achieves a reasonable high security level for secure image communication while maintaining the compatibility with the international standard lossless compression methods.

References

[1] C.-T. Huang, L. Huang, Z. Qin, H. Yuan, L. Zhou, V. Varadharajan, and C.-C.J. Kuo, "Survey on securing data storage in the cloud," APSIPA Transactions on Signal and Information Processing, vol.3, e7, June 2014.

[2] R.L. Lagendijk and M. Barni, "Encrypted signal processing for privacy protection: Conveying the utility of homomorphic encryption and multiparty computation," IEEE Signal Processing Mag., vol.30,

no.1, pp.82–105, Jan. 2013.

[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 Transactions on Information and Forensics and Security, vol.9, no.1, pp.39–50, Jan. 2014.

[4] ISO/IEC JTC 1/SC 29/WG 1 N6402, "Use cases and requirements for JPEG Privacy," July 2013.

[5] O. Watanabe, A. Uchida, T. Fukuhara, and H. Kiya, "An encryption-then-compression system for JPEG 2000 standard," In Proceeding of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp.1226–1230, April 2015.

[6] K. Kurihara, S. Shiota, and H. Kiya, "An Encryption-Then-Compression System for JPEG Standard," In Proceedings of Picture Coding Symposium 2015, pp.119–123, June 2015.

[7] K. Kurihara, M. Kikuchi, S. Imaizumi, S. Shiota, and H. Kiya, "An Encryption-Then-Compression System for JPEG Standard for JPEG/motion JPEG standard," IEICE Transactions on Fundamentals, vol.E98-A, no.11, pp.2238–2245, Nov. 2015.

[8] F. Dufaux, G. Sullivan, and T. Ebrahimi, "The JPEG XR image coding standard [standards in a NUTSHELL]," IEEE Signal Process. Mag., vol.26, no.6, pp.195–199, 204, Oct. 2009.

[9] ISO/ICE 14495-1, "Information technology -Lossless and near-lossless compression of continuous-tone still images Baseline," 1999.

[10] M.G. Chung, M.M. Fleck, and D.A. Forsyth, "Jigsaw puzzle solver using shape and color," IEEE International Conference on Signal Processing (ICSP), vol.2, pp.877–880, Oct. 1998.

[11] D. Sholomon, O. David, and N.S. Netanyahu, "A genetic algorithm-based solver for very large jigsaw puzzles," In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp.1767–1774, June 2013.

[12] P. Butler, P. Chakraborty, and N. Ramakrishan, "The deshredder: A visual analytic approach to reconstructing shredded documents," In Proceedings of IEEE Conference on Visual Analytics Science and Technology (VAST), pp.113–122, Oct. 2012.