A Robust Reversible Data Hiding Scheme with Large Embedding Capacity and High Visual Quality

Doyoddorj Munkbaatar[†], Youngho Park[††], Kyung-Hyune Rhee[†††]

ABSTRACT

Reversible data hiding scheme is a form of steganography in which the secret embedding data can be retrieved from a stego image for the purpose of identification, copyright protection and making a covert channel. The reversible data hiding should satisfy that not only are the distortions due to artifacts against the cover image invisible but also it has large embedding capacity as far as possible. In this paper, we propose a robust reversible data hiding scheme by exploiting the differences between a center pixel and its neighboring pixels in each sub-block of the image to embed secret data into extra space. Moreover, our scheme enhances the embedding capacity and can recover the embedded data from the stego image without causing any perceptible distortions to the cover image. Simulation results show that our proposed scheme has lower visible distortions in the stego image and provides robustness to geometrical image manipulations, such as rotation and cropping operations.

Key words: Reversible Data Hiding, Embedding Capacity, Visual Quality, Robustness

1. INTRODUCTION

Reversible data hiding has recently become a major security technology with increasing importance and widespread distribution of digital media through the Internet with mobile computing [1,2]. The data hiding is referred to as a process to hide secret data into cover media, which plays an important role in multimedia security. Reversibility means that not only secret data but also cover image can be precisely recovered in the decoding stage. Hence, it is applicable to some kind of scenarios such as military remote sensing imaging, diagnostic medical imaging, precious arts protection, and online content distribution systems, and so on.

On the other hand, if some data are embedded into an image, the pixel values of the marked image might be changed. Then, the changes of pixel values are subject to causing degradation to the image. Unless the altered pixels are completely recovered to their original state after the secret data has been extracted, a potential distortion can naturally be occurred. Therefore, reversible data hiding techniques, which can recover the hidden data without degrading the visual quality of original image as far as possible, are necessary.

For some applications, moreover, it is desired that the hidden data will be robust against unintentional changes applying to the stego image, such as geometrical distortion and unavoidable addition of random noise which is below a certain level and does not affect the content of an image. The geo-

※ Corresponding Author : Kyung-Hyune Rhee, Address : (608-737) Pukyong National Univ., 599-1 Dae-Yon 3-Dong Nam-Gu, Busan., TEL : +82-51-629-6247, FAX : +82-51-626-4887, E-mail : khrhee@pknu.ac.kr

Receipt date : Feb. 15, 2012, Revision date : Apr. 24, 2012
Approval date : May 11. 2012

[†] Department of Information Security, Graduate School, Pukyong National University
(E-mail: d_mbtr@pknu.ac.kr)

[††] Department of IT Convergence and Application Engineering, Pukyong National University
(E-mail: pyhoya@pknu.ac.kr)

[†††] Department of IT Convergence and Application Engineering, Pukyong National University

※ This research was supported by Basic Science Research Program through the National Research Foundation of Korea(NRF) funded by the Ministry of Education, Science and Technology(No. 2012-0001331)

metrical distortion is known as one of the most difficult attacks to resist since it can desynchronize the location of the hidden data and hence cause an incorrect secret data extraction. That is, the original cover image should be recovered without any distortion after hidden data extraction only if the stego image remains intact, and conversely, the hidden data should be extracted correctly even if the stego image goes through geometric operations to some extent. Techniques with this property are referred to as a robust data hiding.

However, reversible data hiding introduces certain technical challenges such that increasing the embedding capacity (payload), maintaining the reversible characteristic, decreasing the distortion of the cover image and providing robustness against falsification, simultaneously.

In this paper, we propose a reversible data hiding scheme based on the concept of difference values of the image. Particularly, we divide a cover image into a series of non-overlapping sub-blocks consisting of 3×3 pixels. In each block, a number of pixel differences between a center pixel and its neighboring pixels are calculated while the center pixel remains intact. As a reference of the center pixel, it will be used to restore the neighboring pixels. The extracted difference values of pixel are applied to two important operations for concealing information, named as extra space extraction and random permutation of pixels in each sub-block. These operations allow the proposed scheme to embed not only large secret data but also unnoticeable way into each sub-block in a single-pass period. Moreover, a multi-pass embedding period can be used to increase the embedding capacity. As a result, the proposed scheme can embed much more data than other existing reversible data hiding schemes.

The rest of the paper is organized as follows: Section 2 gives an overview of related work in the area of reversible data hiding. The proposed scheme and its characteristics are described in Section 3. Experimental results and performance comparison are shown in Section 4. Finally, we conclude the paper in Section 5.

2. RELATED WORK

In recent years, a variety of reversible data hiding schemes have been proposed in the literatures. Most of them can be classified into transform domain, compressed domain and spatial domain schemes.

In the transform domain schemes [3], the host image is transformed into a set of coefficients first, and then these coefficients are modified according to secret bits. After that, the modified coefficients are inversely transformed into marked pixels. For example, Yang et al. [4] proposed a reversible data hiding scheme based on integer discrete cosine transform, while Xuan et al.'s scheme [5] is based on integer wavelet transform.

The compressed domain schemes are designed for images compressed by means of JPEG [6], vector quantization [7], block truncation coding [8], etc.

The spatial domain method is based on lossless data compression where the subsequent schemes can be further divided into two subcategories. One is based on difference expansion and the other is based on histogram shifting. The former was developed by Tian [9] with the idea of expanding the difference between a pair of adjacent pixels, and then embedding data in the expanded versions. For example, Chang et al. [10] proposed a just noticeable distortion (JND) based method exploiting pair-difference correlations among DCT domain sub-images. In [11], Alattar extended the difference expansion method via generalized integer transform for capacity enhancement. As a result, $n-1$ bits can be hidden in a set of n adjacent pixels. In addition, various improved difference expansion methods are proposed in [12,13]. In [14], the histogram shifting principle is firstly proposed which usually consists of two stages during data embedding. The first stage is to find peak and zero points of the host image histogram. Then the bits between the peak and zero points are shifted with one level, and hence the peak point is emptied. In the second stage, the secret bit is embedded by adjusting the new peak point and the emptied one. Recently, many reversible data hiding schemes based on histogram shifting are presented in [15,16].

3. PROPOSED REVERSIBLE DATA HIDING SCHEME

3.1 Design Objectives

A fundamental issue in data hiding is to achieve balance between the embedding capacity and the visual quality against image distortion, and to provide robustness against falsification. Even though there are lots of metrics used to evaluate the effectiveness of a data hiding scheme, we consider the followings as our design objectives in this paper:

- **Providing Large Embedding Capacity.** The well-designed data hiding algorithm should be provide a large embedding capacity. That is, the embedding capacity needs to be more than some threshold so that it can be easily recognized after the extraction of hidden data.
- **Preserving High Visual Quality.** A successful image hiding method can maintain a stego image quality that is visually identical to the cover image. The peak-signal-to-noise ratio (PSNR) is a common measure of stego image quality.
- **Assuring Robustness.** The scheme should be resistant to falsification from image processing and malicious attacks. That is, the embedded information cannot be removed beyond reliable detection by targeted attacks based on a full knowledge of the embedding algorithm.

3.2 Workflow Overview

In the proposed scheme, reversible data hiding and retrieval algorithms are designed by calculating a series of pixel differences between a center pixel and its neighboring pixels in each sub-block of the cover image and by utilizing the pixel differences. In this way, the secret data s is embedded into the value of pixel differences in the extracted extra space under secret key k. Table 1 gives the notations that will be used in this paper.

For example, let image be a grayscale cover image with $N \times N$ pixels partitioned into n number of non-overlapped $3 \times 3(m \times m)$ sized sub-blocks. Then $r(r = m \times m)$ pixels in sub-block, $p_1, p_2, \ldots, p_r$ are obtained. In each sub-block, we select a center pixel p_c, and create a series of pixel differences between the center pixel and its neighboring pixel values $d_i = p_c - p_i$, where $1 \le i \le r$ and $i \ne c$. The center pixel p_c of each sub-block remains unaltered in the data embedding and extraction phases to restore other pixel values. After these operations, we can embed bits of the secret data s into the extracted extra space.

In general, the perception of an image is developed by the strong correlation of the neighborhood pixels in the image. Therefore, in order to break such perception, most of the algorithm decorrelates those neighboring pixels, either by moving the pix-

Table 1. Notations.

Notations	Descriptions
s	Secret data (binary bits)
k, k_p	Secret and Permutation key (indexing)
N	Size of image
r	Pixels in sub-block
n	Number of sub-block
p_c	Center pixel in sub-block
p_i	i^{th} pixel in sub-block
d_i	Difference value of i^{th} pixel
d_i^*	Modified value of i^{th} pixel
$d_{k_{p(i)}}^*$	Permuted position of pixel under k_p
τ	Pre-defined threshold
$b_{i,j}$	Binary representation, j^{th} bit of i^{th} pixel

els to other position or changing the value of those pixels according to a certain rule. In our scheme, permutation breaks the neighboring pixels correlation by moving position of a pixel to other position that can produce the pseudorandom sequences with good randomness. A one-to-one pixel permutation is necessary for encryption so that the decryption process is possible [17,18]. The permutation is uses key k_p, generated by a pseudorandom index generator (PRIG), which is usually transmitted through a secured channel for intended users only. A strong level of security for numerous cryptosystems and data hiding schemes is directly dependent on the quality of PRIG. Such a generator can strongly improve the security in data hiding and cryptography. Due to the unpredictability of PRIG, the possibilities offered to attackers to achieve their goal are drastically reduced.

The permutation key is useful for reducing the intelligible information having the properties: (1) displacement of each element from its own location, (2) adjacent element's appearance in different order. We provide the detailed data embedding and extracting processes in the following section.

3.3 Data Hiding and Extraction Phases

Fig. 1 shows the reversible data hiding and extraction phases of our scheme, which consist of creating pixel differences, extra space extraction, random permutation, and embedding and extraction procedures with secret data s, secret key k and random permutation key k_p.

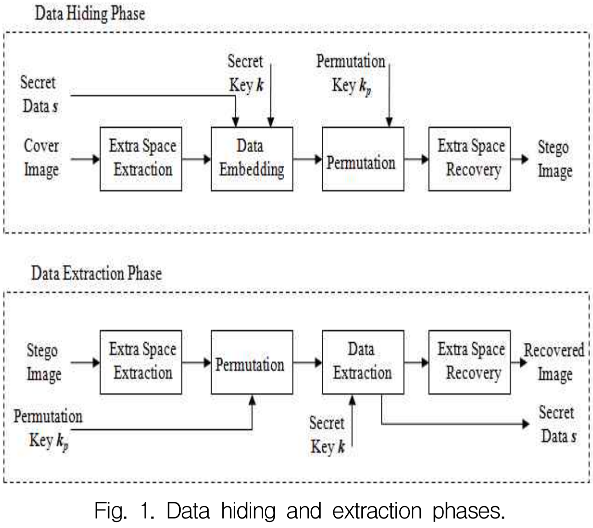


Fig. 1. Data hiding and extraction phases.

Data Hiding Phases. Before data embedding, we modify pixel differences to obtain extra space d_i^*. For this purpose, we set a pre-defined threshold value τ, which is non-negative integer. Then, a difference value d_i is modified to d_i^* as follows:

$$d_i^* = \begin{cases} d_i + 1, & \text{if } d_i > \tau; \\ d_i, & \text{if } (-\tau) \le d_i \le \tau; \\ d_i - 1, & \text{if } d_i < (-\tau). \end{cases} \tag{1}$$

where $1 \le i \le r$ and $i \ne c$, and τ is threshold value.

Then, in the embedding process, each d_i^* can be represented by 8 bits, $b_{i,7}, b_{i,6}, \dots, b_{i,0}$,

$$b_{i,j} = \lfloor d_i^* / 2^j \rfloor \, (\bmod 2), \tag{2}$$

$$d_i^* = \sum_{i=o}^{7} \lfloor b_{i,j} * 2^j \rfloor, \quad j = 0,1,\dots,7. \tag{3}$$

The secret data s_i is embedded into $b_{i,0}$ bit of the modified value d_i^* under the key k_i by using a bitwise XOR operation given in equation (4),

$$b_{i,0} \leftarrow b_{i,0} \oplus s_i \oplus k_i \tag{4}$$

where $1 \le i \le r$, $i \ne c$ and s_i, $k_i \in \{0, 1\}$.

Then, the embedded values $\{d_1^*, d_2^*, \dots, d_r^*\}$ of the extra space in sub-block are applied to the random permutation. The pixels in the sub-block are calculated using the following permutation operation (5).

$$P_{k_p}[\, d_1^*, d_2^*, \dots, d_r^* \,]_n = [\, d^*_{k_{p(1)}}, d^*_{k_{p(2)}}, \dots, d^*_{k_{p(r)}} \,]_n \tag{5}$$

where n is the number of the sub-blocks and $k_p \in \{k_1, k_2, \dots, k_n\}$ is the permutation key.

Finally, we restore the pixels ps_i of stego image by using the difference values between a center pixel and neighboring permuted pixels $ps_i = p_c - d^*_{k_{p(i)}}$, where $1 \le i \le r$ and $i \ne c$. Once the secret data embedded into cover image, the components of the stego image can be changed. In general, the difference between the cover image and the stego image is unnoticeable by the human eyes. Therefore, we can check the histogram distribution of the stego image to identify alteration of processing as shown in Fig. 2.

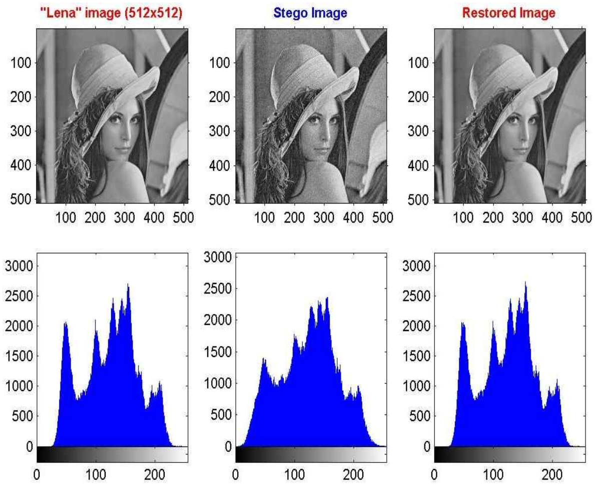


Fig. 2. Histogram distributions of Lena images (512×512).

Data Extraction Phase. We extract the secret data s and reverse the embedded stego image to the cover image. We calculate the difference values $d^*_{k_{p(i)}} = p_c - ps_i$. The permutation is applied to restore a pixel's positions of the each cover sub-blocks by using the permutation key k_p as follows:

$$P_{k_p}[\, d^*_{k_{p(1)}}, d^*_{k_{p(2)}}, \dots, d^*_{k_{p(r)}} \,]_n = [\, d^*_1, d^*_2, \dots, d^*_r \,]_n \qquad (6)$$

In order to reconstruct the secret data s, the difference values d^*_i are represented by binary form $b_{i,j}$. Then, we perform a reverse embedding by using the bitwise XOR operation between the $b_{i,0}$ bit of the modified value d^*_i and secret key k_i given in equation (7),

$$s_i = b_{i,0} \oplus k_i \qquad (7)$$

where $1 \le i \le r$, $i \ne c$ and $s_i, k_i \in \{0,1\}$.

As a result of this operation, we obtain the hidden secret data s and then the $b_{i,0}$ bit value is restored without any degradation by using equation (4). Afterward, the represented value d^*_i from binary form (3) is modified to the difference value d_i, according to the following equation (8),

$$d_i = \begin{cases} d^*_i + 1, & \text{if } d^*_i > \tau + 1; \\ d^*_i, & \text{if } (-\tau) + 1 \le d^*_i \le \tau + 1; \\ d^*_i - 1, & \text{if } d^*_i < (-\tau) + 1. \end{cases} \qquad (8)$$

where $1 \le i \le r$, $i \ne c$ and τ is threshold value.

The pixel of restored image are obtained by calculating a difference between the center pixel and its neighbor pixels as $p_i = p_c - d_i$, where $1 \le i \le r$ and $i \ne c$.

4. EVALUATIONS

4.1 Performance

In this section, we describe our experiments and discuss the results. We simulated our experiments under a PC with 1.8G Hz Dual CPU, 6G RAM, and Windows Vista platform. The simulation was carried out using Matlab version R2008a. In order to evaluate the performance of our proposed scheme, we considered eight commonly used grayscale images with the size of 512×512 as shown in Fig. 3.

The embedding capacity is an important factor for reversible data hiding since one can hide more data with less computation and with a reasonably good perceptual quality. In fact, the embedding capacity is directly related to the quality of the embedded image, which is measured by Peak-Signal-to-Noise-Ratio ($PSNR$) as follows:

$$PSNR = 10\log\frac{255^2}{MSE};\ MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}[I_{i,j} - I'_{i,j}]^2 \qquad (9)$$

where MSE is the mean square error between the cover image I and the corresponding stego image I'.

Typically, it is acceptable if the $PSNR$ in both lossy image and video compression is between 30 dB and 50 dB, where higher is better performance. The permissible quantity of concealing (PQC) is the proportion of the maximum quantity of concealment to the size of image. The PQC can be found by the formula $PQC = Q/(N \times N)$, where Q is the total quantity of payload.

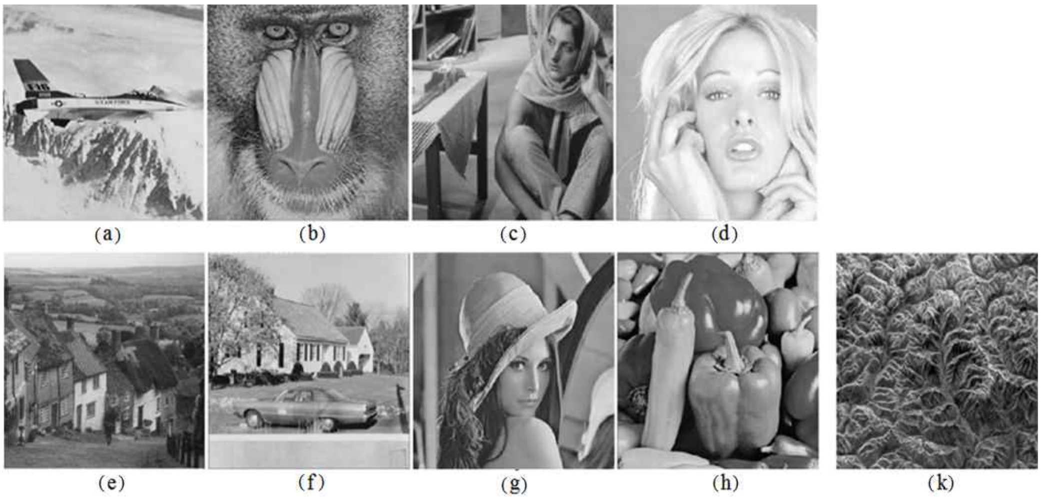


Fig. 3. Test images (512×512). (a) Airplane, (b) Baboon, (c) Barbara, (d) Tiffany, (e) Goldhill, (f) House, (g) Lena, (h) Pepper and (k) Secret data s (binary image 231,200bits).

We chose the Fig. 3(k) as the secret data, which is divided into several payload parts: 30 percent of whole payload is 69,360 bits, 60 percent of whole payload is 138,720 bits, 100 percent of whole payload is 231,200 bits, and so on.

We simulated the embedding quality of Lena image with a different payload sizes as shown in Fig. 4. We can embed 231,200 bits (0.889bpp) payload (≈28K bytes), for an image of 512×512 size, and presented the results of *PSNR* for all test images in Table 2. In this experiment, we considered two parameters related to improve the performance of the scheme: pre-defined threshold τ which is a condition for generating extra space, and R is repetition of the embedding through multi-pass period.

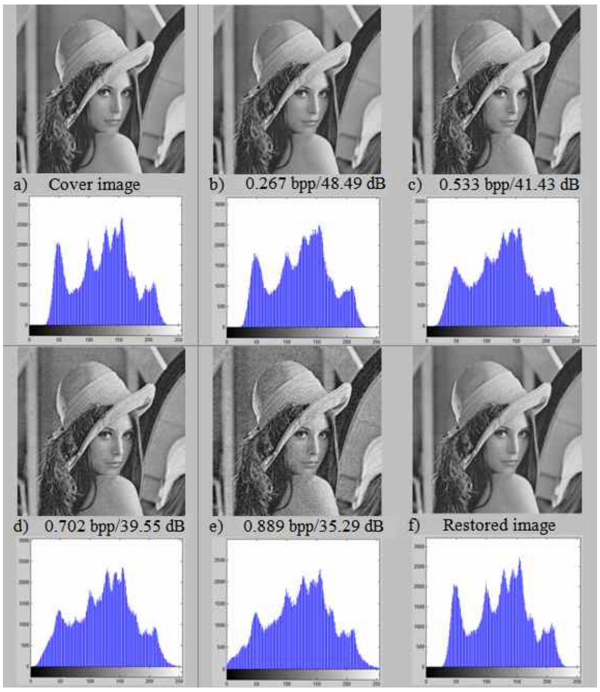


Fig. 4. Histogram distributions of Lena image according to different payload sizes.

As shown in Table 2, the pre-defined threshold τ value is not effective to the increase of *PSNR*, but repetition R of sub-block is directly affected to the decrease of *PSNR*. However, the number of repetition can generate secure data hiding system because the pixels of each sub-block will permute R times.

Table 3 shows the visual quality and the per-

Table 2. The visual quality of test images depend on threshold value and multi-pass period of embedding

Test Images	PSNR(dB)					
	τ (threshold)			R (times)		
	0	2	5	1	3	5
Airplane	47.36	47.34	47.29	47.43	43.26	40.12
Baboon	47.12	47.17	47.09	47.09	42.93	39.56
Barbara	47.66	47.59	47.61	47.68	43.61	41.25
Tiffany	48.32	48.38	48.32	48.35	43.89	40.97
Goldhill	48.26	48.25	48.33	48.35	42.61	41.36
House	47.46	47.61	47.54	47.52	41.87	40.12
Lena	48.34	48.30	48.36	48.30	44.65	41.42
Pepper	48.05	48.17	48.19	48.26	42.76	41.43

Table 3. Visual quality and permissible quantity of test images depend on payload sizes

Test Images	PSNR dB	payload (30%)	PSNR dB	payload (60%)	PSNR dB	payload (100%)
		PQC		PQC		PQC
Airplane	47.27		39.53		33.41	
Baboon	47.11		38.87		34.59	
Barbara	47.59	69,360bits	40.15	138,720bits	34.73	231,200bits
Tiffany	48.24		42.65		35.34	
Goldhill	48.31	0.267bpp	41.69	0.533bpp	36.19	0.889bpp
House	46.54		39.13		33.65	
Lena	48.49		41.43		35.29	
Pepper	48.23		40.68		35.63	

missible quantity of all test images depend on payload sizes. From these comparisons, we can observe that our scheme has achieved the higher *PSNR* with a quite large data embedding capacity. The results of the *PSNR* values of stego images are larger than 46 dB with a payload 0.267 bpp, so that our scheme can be observed high embedding quality and low image degradation.

The computational complexity of the proposed scheme is less complex since it does not need to apply any transform such as discrete cosine transform (DCT) and Fourier transform (FFT). The required processing mainly lies on generating histogram, scanning pixels, and adding or subtracting grayscale values in spatial domain. Hence, the execution time is rather short.

Fig. 5 shows the results of *PSNR* where a single pass of embedding is applied to each test images, respectively. To measure the *PSNR* depending on payload size, we varied the payload size from 0.267 bpp to 0.889 bpp which is 30% to 100% of whole payload. According to the decrease of the payload values, the qualities of the stego images are up to 46.49 dB, which means the image degradations are lower with the payload 0.267 bpp. In fact, the average *PSNR* of the stego images of our scheme is 47.72 dB, when the payload stays around 0.267 bpp on all test images.

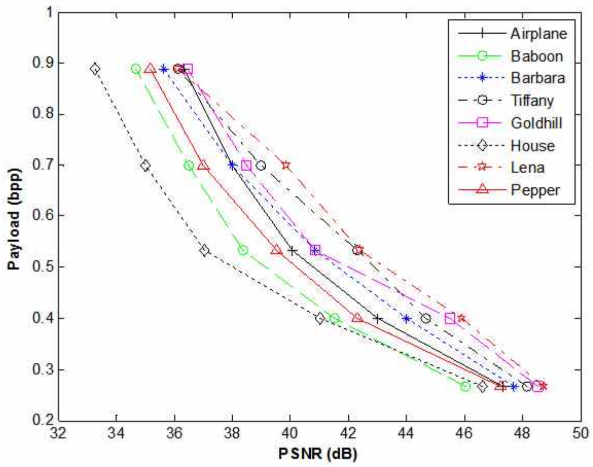


Fig. 5. The performance of test images depend on payload sizes.

The next experiment was designed to compare the performance of our scheme with those of the other reversible hiding schemes [6,9,14]. Table 4 shows the results of this experiment indicated. In Chang et al.'s method [6], secret data is hidden in

Table 4. The performance comparison with previous data hiding schemes.

Scheme	PSNR (dB)	Capacity (bits)	PQC (bpp)
Lena			
Chang et al.‘s	40.49	4,096	0.015
Ni et al.‘s	48.20	5,460	0.021
Tian's	43.52	52,445	0.200
Proposed	48.49	69,360	0.301
Airplane			
Chang et al.‘s	40.26	4,096	0.015
Ni et al.‘s	48.30	16,171	0.062
Tian's	44.49	65,920	0.251
Proposed	47.27	69,360	0.301
Baboon			
Chang et al.‘s	35.95	4,096	0.015
Ni et al.‘s	48.20	5,421	0.021
Tian's	36.39	27,948	0.106
Proposed	47.11	69,360	0.301

each block of quantized discrete cosine transformation (DCT) coefficients on a JPEG image. Two successive zero coefficients of the medium-frequency components in each block are used for secret data to be embedded. Let L be the number of bits carried in one block. Overall, our scheme is superior to Chang et al.'s method.

Our scheme can offer a large embedding capacity than Tian's scheme [9]. Since there is large variation in the gray-levels of most adjacent pixels in "Baboon", the amount of extra data is so large required to be recorded by Tian's scheme [9] that there is not enough space to hide the secret data. Although Ni et al.'s scheme [14] provides a better stego image quality, hiding capacity of Ni et al.'s is much smaller than that of ~~the~~ our scheme.

We considered the confidentiality of our proposed scheme according to random permutation. To protect the embedded data securely, a common way is to use a cryptographic method additionally with the data hiding. It is desired to add security protection directly into the data hiding process, aiming to increase the difficulty for unauthorized user to extract, alter, and forge hidden data by manipulating the stego image. Hence, the stego image should be possess certain random properties. We have performed statistical analysis by calculating the histogram of the correlation of two adjacent pixels in the stego images. Fig. 6 shows the results of the permutation techniques. Here, we simulated two different techniques such as bit and pixel permutations.

As a result, the bit permutation is better approach for image encryption and hashing according to the perception of hiding. Generally, after the bit permutation, the encrypted image will be appeared as a noisy image. However, reversible data hiding means that not only secret data but also cover image must be precisely recovered in decoding.

In the pixel permutation, eight pixels in a sub-block are taken as a group and permuted with the same sized key. Our obtained the stego image is nearly similar to the cover image due to high correlation between the adjacent pixels in 3×3 resized sub-block as shown in Fig. 6. Thus, the pixel permutation with small sized sub-block is suitable for reversible data hiding. But in larger 16×16 resized sub-block, the edges are slightly distorted in the encrypted image. However, the histogram of the stego image is same as the histogram of the cover image while the pixel values are same after encryption but their position will be changed.

4.2 Robustness

In this section, we evaluated the robustness of our scheme against several image processing operations. The robustness in data hiding is the

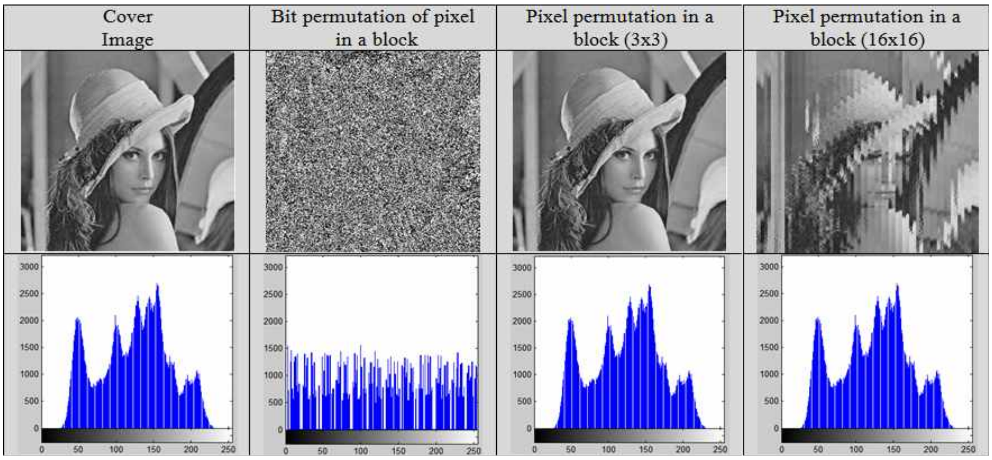


Fig. 6. The results of bit and pixel based permutations.

process of extraction the correct data after compression or any other alteration applied on the embedded image.

Hence, for fair benchmarking and performance evaluation, the robustness due to the embedding is an important issue. Since there is no universal metric, we review in this section the most popular pixel-based distortion criteria and introduce one metric which makes use of effect in the human visual system (HVS) [19]. Most distortion measures used in visual information processing belong to the group of difference distortion measures, such as similarity measure.

To confirm the efficiency and robustness, well-known attacks such as rotation, cropping, JPEG compression and Gaussian noise are applied to our scheme in Fig. 7. For each attack, we computed a similarity measure between an original embedded data and attacked embedded data according to the percentage. This percentage is number of equal bits between original and extracted embedded data. A result less than or equal to 50% implies that the cover image has probably not been hidden. Table 5 shows the results of the robustness test against various image processing operations.

- **Rotation.** Let r_α be the rotation of angle α around the center of the stego image. The transformation r_α is applied to the test image. Small angle rotation, often in combination with cropping, does not usually change the commercial value of the image but can make the embedded data un-detectable. By analyzing the similarity percentage between original and extracted embedding data, we can conclude that in case of robustness, the embedding data still remains after a rotation attack as shown in Table 5. The desired robustness is reached.
- **Cropping.** The attacker extracts a certain region of interest of the stego image while discarding the other portions. Obviously, we cannot invert cropping because it incurs a permanent loss of information about the discarded pixels. It can be noticed that cropping sizes and similarity percentages are rather inversely proportional as shown in Table 5. But, in case of robustness, even a small change of the stego image (a crop by 10 sub-blocks) leads to little different extracted embedded data. In this case, any attempt to alter the stego image will be signaled, thus the image is well authenticated.
- **JPEG compression.** A JPEG compression is applied to the test image depending on a compression level. This attack leads to a change of the representation domain (Spatial to DCT). In this case, the results in Table 5 illustrate a different robustness through JPEG compression. Here, the embedded data still remains after a compression level equal to 2, which is a good result on the different domain for embedding. For the authentication case, however, our scheme does not fulfill robustness to JPEG compression.
- **Gaussian noise.** The embedded image can be also attacked by the addition of a Gaussian noise

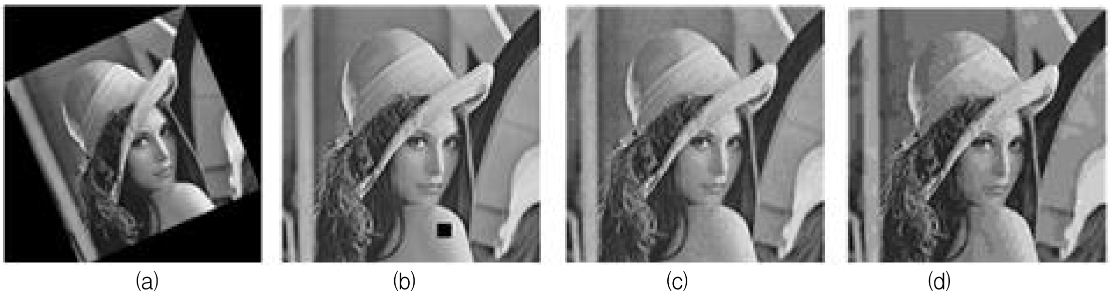


Fig. 7. An attacked images by various image processing. (a) Rotation (α=10°), (b) Cropping (50 sub-blocks), (c) Gaussian noise (dev.=2), (d) JPEG compression (r=10).

Table 5. The results of the robustness test against image processing operations

Attacks	Robustness		
Rotation	Angle(degree)	Similarity(%)	PSNR(dB)
	2	99.65	48.59
	5	98.12	47.85
	10	97.38	45.37
Cropping	Size(blocks)	Similarity(%)	PSNR(dB)
	10	98,41	47.65
	30	96.59	44.26
	50	94.15	40.48
JPEG compression	Compression(rate)	Similarity(%)	PSNR(dB)
	2	82.95	36.12
	5	64.25	27.56
	10	53.13	22.64
Gaussian noise	Standard dev.	Similarity(%)	PSNR(dB)
	1	65.26	29.25
	2	62.18	26.32
	3	54.89	24.61

depending on a standard deviation. According to the increase of deviation, the robustness of the scheme is directly affected to the decrease of the measure of similarity. In this case, our scheme is not robust to such operation as shown in Table 5. This result corresponds to our expectation because the proposed scheme is processed according to the pixel values.

5. CONCLUSION

In this paper, we proposed a robust reversible data hiding technique for grayscale image. The proposed scheme utilizes the difference values between the neighboring pixels in a sub-block to embed the embedding data. Our scheme not only improves the visual quality but also provides larger payload capacity than other related methods. Specifically, the proposed method is able to embed about 5K bytes through 28K bytes into a 512×512 grayscale image while guaranteeing the *PSNR* of the stego image versus the cover image to be above 46 dB. This implies that the proposed scheme can offer high embedding quality with a low image degradation. In addition, our scheme is robust against some attacks based on image processing such as rotation and cropping operations. It is expected that our reversible data hiding technique will be deployed for a wide range of applications in the areas such as secure medical image data system, law enforcement, image authentication and covert secure communication, and so on.

REFERENCES

[1] C.C. Chang and T.D. Kieu, “A Reversible Data Hiding Scheme using Complementary Embedding Strategy,” *Information Sciences*, Vol. 180, No. 16, pp. 3045-3058, 2010.

[2] 김진호, 서용수, 권기룡, “GIS 웹 맵 서비스 구현을 위한 스마트 폰에서의 정보은닉 기법,” *멀티미디어학회논문지*, 제13권, 제5호, pp. 710-721, 2010.

[3] C.C. Chang, P.Y. Pai, C.M. Yeh, and Y.K. Chan, “A High Payload Frequency-Based Reversible Image Hiding Method,” *Information Sciences*, Vol. 180, No. 11, pp. 2286-2298, 2010.

[4] B. Yang, M. Schmucker, X. Niu, C. Busch, and S.H. Sun, “Integer-DCT-based Reversible Image Watermarking by Adaptive Coefficient

Modification," *Proc. of the SPIE, Security, Steganography, and Watermarking of Multimedia Contents*, Vol. 5681, pp. 218-229, 2005.

[5] G. Xuan, Y.Q. Shi, Q. Yao, Z. Ni, C. Yang, J. Gao, and P. Chai, "Lossless Data Hiding using Histogram Shifting Method Based on Integer Wavelets," *International Workshop on Digital Watermarking, Lecture Notes in Computer Science*, Vol. 4283, pp. 323-332, 2006.

[6] C.C. Chang, C.C. Lin, C.S. Tseng, and W.L. Tai, "Reversible Hiding in DCT-Based Compressed Images," *Information Sciences*, Vol. 177, No. 13, pp. 2768-2786, 2007.

[7] Z.M. Lu, J.X. Wang, and B.B. Liu, "An Improved Lossless Data Hiding Scheme Based on Image VQ-Index Residual Value Coding," *Journal of Systems and Software*, Vol. 82, No. 6, pp. 1016-1024, 2009.

[8] C.C. Chang, C.Y. Lin, and Y.H. Fan, "Lossless Data Hiding for Color Images Based on Block Truncation Coding," *Pattern Recognition*, Vol. 41, No.7, pp. 2347-2357, 2008.

[9] J. Tian, "Reversible Data Embedding using A Difference Expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 13, No. 8, pp. 890-896, 2003.

[10] C.C. Chang, P.Y. Lin, and J.S. Yeh, "Preserving Robustness and Removability for Digital Watermarks using Sub-Sampling and Difference correlation," *Information Sciences*, Vol. 179, No. 13, pp. 2283-2293, 2009.

[11] A.M. Alattar, "Reversible Watermark using the Difference Expansion of A Generalized Integer Transform," *IEEE Transactions on Image Processing*, Vol. 13, No. 8, pp. 1147-1156, 2004.

[12] J.Y. Hsiao, K.F. Chan, and J.M. Chang, "Block-Based Reversible Data Embedding," *Signal Processing*, Vol. 89, No. 4, pp. 556-569, 2009.

[13] H.W. Tseng and C.C. Chang, "An Extended Difference Expansion Algorithm for Reversible Watermarking," *Image and Vision Computing*, Vol. 26, No. 8, pp. 1148-1153, 2008.

[14] Z. Ni, Y.Q. Shi, N. Ansari, and W. Su, "Reversible Data Hiding," *IEEE Transactions on Circuits and Systems for Video Technology*, Vol. 16, No. 3, pp. 354-362, 2006.

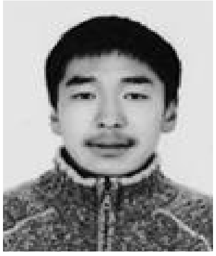
[15] H.C. Huang, W.C. Fang, and I.T. Tsai, "Reversible Data Hiding using Histogram-Based Difference Expansion," *IEEE International Symposium on Circuits and Systems*, pp. 1661-1664, 2009.

[16] C.C. Lin, W.L. Tai, and C.C. Chang, "Multilevel Reversible Data Hiding Based on Histogram Modification of Difference Images," *Pattern Recognition*, Vol. 41, No. 12, pp. 3582-3591, 2008.

[17] A. Gautam, M. Panwar, and R. Gupta, "A New Image Encryption Approach using Block Based Transformation Algorithm," *Journal of Advanced Engineering Sciences and Technologies*, Vol. 1, No. 8, pp. 90-96, 2011.

[18] S.R.M. Prasanna, "Study of Permutations in the Context of Speech Privacy," *in Proceeding. ECCAP*, pp. 99-106, 2000.

[19] M. Kutter and F. Petitcolas, "A Fair Benchmark for Image Watermarking Systems," *Proc. of SPIE conference on Security and Watermaking of Multimedia Contents*, Vol. 3657, pp. 226-239, 1999.



Doyoddorj Munkbaatar

2003 B.S. degree from National University of Mongolia

2011 M.S. degree from Dept. of Information Security, Pukyong National University

2011~onward Ph.D. course in Information Security, Pukyong National University

interesting: steganography, watermarking, image forensics

Youngho Park

2000 B.S. degree in Computer Science from Pukyong National University

2002 M.S. degree in Computer Science from Pukyong National University

2006 Ph.D. in Information Security from Pukyong National University

2011 Post-Doc. Researcher at Pukyong National University

interesting: cryptographic protocols and applications, communication security

Kyung-Hyune Rhee

1982 B.S. degree in Mathematics Education from Kyungpook National University

1985 M.S. Degree in Applied Mathematics from KAIST

1992 Ph.D. in Mathematics from KAIST

1993~onward Professor at Pukyong National University

interesting: information security, cryptography, communication security, multimedia security