ELectronics EXpress



# Improved reversible data hiding through full employment of image pixels

**Jihong Kang, Honglin Jin[a), and Yoonsik Choe**

*Department of Electrical and Electronic Engineering, Yonsei University, Seoul, Republic of Korea*

a) *hljin@yonsei.ac.kr*

**Abstract:** A new reversible data hiding algorithm for digital images is proposed in this paper. A single key parameter is derived in the algorithm during the hiding process. This parameter must be transferred to extract data. A $3 \times 3$ window slides over the cover image by one pixel unit, and one bit can be embedded at each position of the window. Thus, the ideal maximum hiding capacity is equal to the number of pixels in the image. As a result, significant increases in hiding capacity and better visual quality of message-hidden images may be achieved. The proposed algorithm is verified with simulations.

**Keywords:** moving window, key parameter, reversible data hiding, lossless data hiding, watermarking

**Classification:** Fiber optics, Microwave photonics, Optical interconnection, Photonic signal processing, Photonic integration and systems

## References

[1] H. L. Jin, M. Fujiyoshi, and H. Kiya, "Lossless data hiding in the spatial domain for high quality images," *IEICE Trans. Fundamentals*, vol. E90-A, no. 4, pp. 771–777, April 2007.

[2] S. Arjun and N. Rao, "An approach to reversible information hiding for images," *TENCON 2008 IEEE Region 10 Conference*, Hyderabad, India, Nov. 2008.

[3] H. L. Jin, Y. Choe, and H. Kiya, "Reversible data hiding based on adaptive modulation of statistics invertibility," *IEICE Trans. Fundamentals*, vol. E93-A, no. 2, pp. 565–569, Feb. 2010.

## 1 Introduction

Reversible data hiding refers to a technique that embeds hidden information into digital images and completely recovers the original image after extracting the hidden information. In fields that require absolute integrity of the original image, such as art, medical care, and military applications, this technique can provide extra space in the image itself for confidential information attachment, authentication, security, and copyright protection.



DOI: 10.1587/elex.8.866
Received April 30, 2011
Accepted May 09, 2011
Published June 10, 2011

IEICE Electronics Express, Vol.8, No.11, 866–872

Over the past several years, many algorithms have been studied to satisfy the demanding conditions for reversible data hiding. Such conditions include low complexity of the algorithm, imperceptivity, high hiding capacity, and low degradation of a message-hidden image (stego image). A new technique was introduced in [1] that uses a single key parameter instead of hiding additional information. In this method, a cover image is divided into partially overlapped $3 \times 3$ blocks and message bits can ideally be hidden in up to a quarter of the image pixels. The conventional method [3] improved upon the previous method [1] with a new scheme that increases the real hiding capacity. And, [2] modified the shapes of hiding blocks based on the work in [1]. With the modified hiding blocks, the ideal maximum capacity reaches half of the number of image pixels and, thus, the real capacity is also increased.

This paper, we propose an improved reversible data hiding method that is based on the conventional hiding method [3]. By modified hiding and extraction procedures, a $3 \times 3$ moving window can hide the message bits into all of the image pixels. In experiments, the hiding capacity and the visual quality of the stego image were significantly improved over the conventional method.

## 2 Conventional method

The hiding and extraction processes of the conventional method [3] are described below. The conventional method hides a message bit $w_n$ into the center pixel $g_b$ of the $b$-th overapped $3 \times 3$ block. Each block center is separated vertically and horizontally on the image by two pixel units. Therefore, for an $X \times Y$ image, the ideal maximum capacity $B$ is

$$B = \lfloor (X-1)/2 \rfloor \lfloor (Y-1)/2 \rfloor \quad \text{[bits]}.$$

Before hiding information in an original image $\mathbf{f}$, the key parameter $s$ is derived through the following procedure.

Step 1. $b := 0$.

Step 2. The average of the surrounding pixels $g_{b,0}$, $g_{b,1}$, $\ldots$, $g_{b,7}$ and the difference $d_b$ between $g_b$ and $\bar{g}_b$ are derived as

$$\bar{g}_b = \left\lfloor \frac{1}{8} \sum_{i=0}^{7} g_{b,i} \right\rfloor, \tag{1}$$

$$d_b = g_b - \bar{g}_b. \tag{2}$$

Step 3. The standard deviation $\delta_b$, the hiding basis $\tilde{g}_b$, and the range of the surrounding pixels $u_b$ are calculated as

$$\delta_b = \sqrt{\frac{1}{8} \sum_{i=0}^{7} (g_{b,i} - \bar{g}_b)^2}, \tag{3}$$

$$\tilde{g}_b = \bar{g}_b + 2 \times \text{round}\left( (2 + \delta_b) \left( 1 - \frac{\bar{g}_b}{2^{K-1}} \right) \right), \tag{4}$$

$$u_b = g_{\max,b} - g_{\min,b}. \tag{5}$$


DOI: 10.1587/elex.8.866
Received April 30, 2011
Accepted May 09, 2011
Published June 10, 2011

where $K$ represents the depth of pixels. Also, $g_{\max,b} = \max_i g_{b,i}$ and $g_{\min,b} = \min_i g_{b,i}$ $(i = 0, 1, \ldots, 7)$.

Step 4. The candidate of $s$ is obtained from Eq. (6).

$$s_b = \begin{cases} u_b, & \tilde{g}_b + 2d_b < 0 \text{ or } 2^K - 2 < \tilde{g}_b + 2d_b \\ \infty, & \text{otherwise} \end{cases}. \tag{6}$$

Step 5. $b := b + 1$. If $b < B$, then go to Step 2.

Step 6. The key parameter $s$ is obtained by $s = \min_b s_b$.

By decreasing the values of the derived key parameter $s$, the real capacity of the image can be reduced so as to match the length of the message. Next, the following equation is applied for hiding into the original image $\mathbf{f}$.

$$\hat{g}_b = \begin{cases} \tilde{g}_b + 2d_b + w_n, & u_b < s \\ g_b, & \text{otherwise} \end{cases}. \tag{7}$$

This hiding condition $u_b < s$ ensures that the values of the embedded pixels do not deviate from the dynamic range of $[0, 2^K - 1]$. After applying Eq. (7) to all blocks on the image, the stego image $\hat{\mathbf{f}}$ is obtained.

In the extraction and restoration process, the hidden message bit $w_n$ is extracted by Eq. (8) and the original pixel $g_b$ is recovered with Eq. (9).

$$\text{If } u_b < s, \quad w_n = (\hat{g}_b - \tilde{g}_b) \bmod 2. \tag{8}$$

$$g_b = \begin{cases} \bar{g}_b + \frac{\hat{g}_b - \tilde{g}_b - w_n}{2}, & u_b < s \\ \hat{g}_b, & \text{otherwise} \end{cases}. \tag{9}$$

This method uses only a quarter of the image pixels for hiding targets. However, the proposed method can use all of the pixels for hiding targets through modifications of the conventional algorithm [3].

## 3 Proposed method

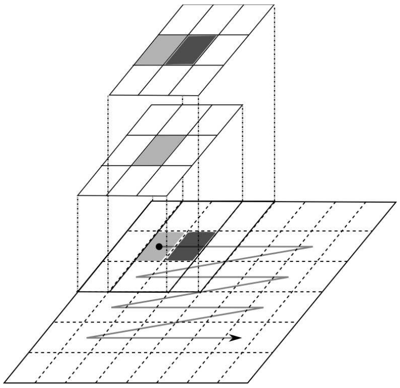
The proposed method, a $3 \times 3$ window is used as a basic unit for both hiding and extraction processes. As shown in Fig. 1 (a), the moving window slides over the target image by one pixel unit at a time in raster-scan order while performing the processes of hiding and key parameter derivation simultaneously. At each position of the window, one bit can be embedded into the center pixel of the window. Therefore, the ideal maximum hiding capacity of an image is equal to the number of pixels in the image.

However, discontinuity of the borders of an image causes large errors in the estimation of the target pixel in Eq. (10). This impedes hiding information into other positions in the remainder of the image by forcing the key parameter to be derived as a very low value. As such, we perform the hiding process on the target image by excluding border pixels in our proposed method. The ideal maximum capacity of an $X \times Y$ image then becomes

$$B = (X - 2)(Y - 2) \quad \text{[bits]}.$$


DOI: 10.1587/elex.8.866
Received April 30, 2011
Accepted May 09, 2011
Published June 10, 2011

IEICE Electronics Express, Vol.8, No.11, 866–872



(a) The moving direction of the hiding window

| | | |
|---|---|---|
| $g_{b,0}$ | $g_{b,1}$ | $g_{b,2}$ |
| $g_{b,3}$ | $g_b$ | $g_{b,4}$ |
| $g_{b,5}$ | $g_{b,6}$ | $g_{b,7}$ |

(b) Top left

| | | |
|---|---|---|
| $g_{b,0}$ | $g_{b,1}$ | $g_{b,2}$ |
| $\hat{g}_{b,0}$ | $g_b$ | $g_{b,3}$ |
| $g_{b,4}$ | $g_{b,5}$ | $g_{b,6}$ |

(c) Top row

| | | |
|---|---|---|
| $g_{b,0}$ | $\hat{g}_{b,0}$ | $\hat{g}_{b,1}$ |
| $g_{b,1}$ | $g_b$ | $g_{b,2}$ |
| $g_{b,3}$ | $g_{b,4}$ | $g_{b,5}$ |

(d) Left border

| | | |
|---|---|---|
| $\hat{g}_{b,0}$ | $\hat{g}_{b,1}$ | $g_{b,0}$ |
| $\hat{g}_{b,2}$ | $g_b$ | $g_{b,1}$ |
| $g_{b,2}$ | $g_{b,3}$ | $g_{b,4}$ |

(e) Right border

| | | |
|---|---|---|
| $\hat{g}_{b,0}$ | $\hat{g}_{b,1}$ | $\hat{g}_{b,2}$ |
| $\hat{g}_{b,3}$ | $g_b$ | $g_{b,0}$ |
| $g_{b,1}$ | $g_{b,2}$ | $g_{b,3}$ |

(f) The rest

**Fig. 1.** (a) The scanning order of the proposed method. (b)–(f) The location of genuine pixels $g_{b,i}$ and previously embedded pixels $\hat{g}_{b,j}$ within the window. Each caption under the figures corresonds to the position of the window on the image.

### 3.1 Derivation of the key parameter $s$ and data hiding

The following steps show the derivation of the key parameter $s$ and the hiding process for an original image $\mathbf{f} = \{(x, y) \mid x = 0, 1, \ldots, X-1,\ y = 0, 1, \ldots, Y-1\}$.

Step 1. $s := 2^K - 1$.

Step 2. $b := 0$, $n := 0$, $update\_flag := 0$.

Step 3. $d_b$ is calculated from Eq. (2). Also, $\bar{g}_b$ is obtained from Eq. (10) using both groups of surrounding pixels: genuine pixels ($g_{b,0}$, $g_{b,1}$, $\ldots$, $g_{b,p-1}$) and previously embedded pixels ($\hat{g}_{b,0}$, $\hat{g}_{b,1}$, $\ldots$, $\hat{g}_{b,q-1}$).

$$\bar{g}_b = \left\lfloor \frac{1}{8} \left( \sum_{i=0}^{p-1} g_{b,i} + \sum_{j=0}^{q-1} \hat{g}_{b,j} \right) \right\rfloor. \tag{10}$$

According to the center position of the window $\{b(x, y) \mid x = 1, 2, \ldots, X-2,\ y = 1, 2, \ldots, Y-2\}$, the number of elements in each group $(p, q)$ is given as follows: if $x = 1$, $y = 1$, $(p, q) = (8, 0)$; if $x \neq 1$, $y = 1$, $(p, q) = (7, 1)$; if $x = 1$, $y \neq 1$, $(p, q) = (6, 2)$; if $x = X-2$, $y \neq 1$, $(p, q) = (5, 3)$; otherwise, $(p, q) = (4, 4)$. The location of each group's elements within the window refers to Fig. 1 (b)–(f).

Step 4. $\delta_b$ is also calculated by including both groups of surrounding pixels. The $\tilde{g}_b$ and $u_b$ terms can be obtained from Eq. (4) and Eq. (5), respectively.

$$\delta_b = \sqrt{\frac{1}{8} \left( \sum_{i=0}^{p-1} (g_{b,i} - \bar{g}_b)^2 + \sum_{j=0}^{q-1} (\hat{g}_{b,j} - \bar{g}_b)^2 \right)}. \tag{11}$$


DOI: 10.1587/elex.8.866
Received April 30, 2011
Accepted May 09, 2011
Published June 10, 2011

IEICE Electronics Express, Vol.8, No.11, 866–872

Step 5. Check the update of $s$.

$$\text{If } (u_b < s) \text{ and } (\tilde{g}_b + 2d_b < 0 \text{ or } \tilde{g}_b + 2d_b > 2^K - 2), \\ \text{then, } s = u_b,\ update\_flag = 1, \text{ and go to Step 7.} \quad (12)$$

Step 6. Hiding is executed with Eq. (13).

$$\hat{g}_b = \begin{cases} \tilde{g}_b + 2d_b + w_n, & u_b < s \text{ and } 0 \le \tilde{g}_b + 2d_b \le 2^K - 2 \\ g_b, & \text{otherwise} \end{cases}. \quad (13)$$

If a message bit $w_n$ is hidden in Eq. (13), then $n := n + 1$.

Step 7. $b := b + 1$. If $b < B$, then go to Step 3.

Step 8. If $update\_flag = 1$, then go to Step 2.

Step 9. A stego image $\hat{\mathbf{f}}$ and the key parameter $s$ are obtained.

At each $b$-th position of the window, the key parameter $s$ is updated through Steps 3–5. The embedded pixel $\hat{g}_b$ is also obtained in Step 6.

### 3.2 Extraction and restoration

The extraction and restoration scheme is similar to that of the conventional method [3], except for the moving direction of the extracting window, which is in the reverse direction of the hiding window. A message bit is extracted from Eq. (8) and the restored pixel in Eq. (9) is immediately applied to Eq. (10) and Eq. (11) as a genuine surrounding pixel for the next position of the extracting window.

### 3.3 Differences with conventional methods

When compared to conventional methods [2, 3], three major changes are implemented in the proposed technique in order to expand the number of target pixels to all pixels on the image. First, the moving window includes the previously embedded pixels as surrounding pixels. Second, the key parameter derivation and hiding processes are combined and performed recursively. Last, the moving direction of the extracting window is opposite to that of the hiding window.

## 4 Experimental results

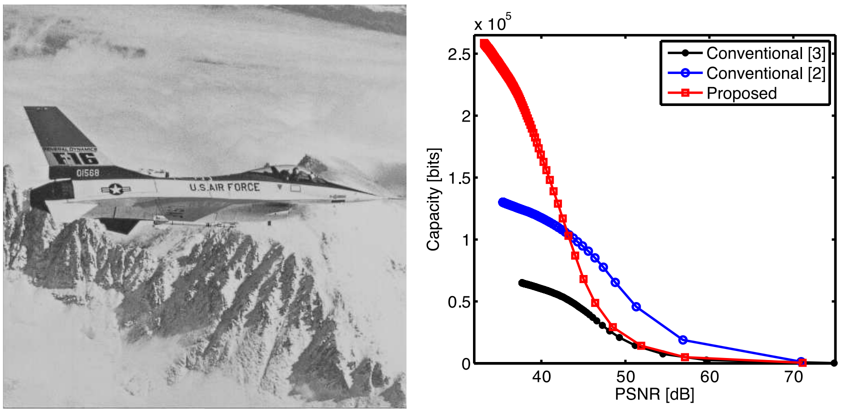
Simulations were performed using $512 \times 512$ grayscale images ($K = 8$). The ideal capacities of the conventional methods outlined in [3] and [2] were 65025 [bits] and 130050 [bits], respectively for the $512 \times 512$ images. In the proposed method, the ideal capacity increased to 260100 [bits]. In addition, as shown in Table I, the real capacity $N$ [bits] significantly increased in most of the test images and reached 258638 [bits].

For the large amount data hiding, the proposed method shows better visual quality of the stego image. The stego image of Airplane in Fig. 2 (a) shows 33.16 [dB] of PSNR when 258638 [bits] are hidden. In Fig. 2 (b), the y-axis represents the number of hidden bits while the x-axis is the PSNR


DOI: 10.1587/elex.8.866
Received April 30, 2011
Accepted May 09, 2011
Published June 10, 2011

**Table I.** Real capacity $N$ [bits] and PSNR [dB].

| Image | Conventional [3] | | Conventional [2] | | Proposed | |
|---|---|---|---|---|---|---|
| | $N$ | PSNR | $N$ | PSNR | $N$ | PSNR |
| Aerial6 | 21860 | 39.80 | 4322 | 63.17 | 7155 | 47.29 |
| Airplane | 65025 | 37.70 | 130050 | 35.44 | 258638 | 33.16 |
| Baboon | 44162 | 34.51 | 31226 | 42.52 | 100157 | 34.38 |
| Barbara | 52870 | 34.30 | 57814 | 45.08 | 174514 | 32.44 |
| Lena | 65025 | 35.60 | 120351 | 38.12 | 251326 | 31.41 |
| Peppers | 64842 | 35.98 | 112220 | 40.27 | 251240 | 32.25 |
| Sailboat | 64524 | 33.91 | 129669 | 32.65 | 201871 | 31.44 |
| Tiffany | 64985 | 35.26 | 72058 | 46.45 | 98563 | 40.29 |




(a) $N$=258638 [bits], PSNR=33.16 [dB]

(b) PSNR vs. Capacity

**Fig. 2.** The stego image and the PSNR comparison of "Airplane".

of the stego image versus the original image. When the length of hidden bits was below 103023 [bits], the PSNR of the proposed method was lower than the conventional method [2]. However, as more amount of data was hidden to the image, the PSNR of the proposed method maintained the highest position. Furthermore, the proposed method always held a dominant position against the conventional method [3] in the graph. The main reason for this improvement is that the increased number of target pixels enables to fully utilize the pixels in the smooth region of the image, which cause small distortion on the stego image. Therefore, when the same length of message is given, the proposed method can achieve the target capacity with less distortion by deriving lower value of key parameter than the conventional method [3].

## 5 Conclusions

We proposed an efficient reversible data hiding method for digital images. The proposed method achieved almost 1 [bits per pixel] of ideal maximum capacity and significant increment in real capacity, while maintaining higher PSNR of the stego image in large amount data hiding than the conventional


DOI: 10.1587/elex.8.866
Received April 30, 2011
Accepted May 09, 2011
Published June 10, 2011

methods [2, 3].

## Acknowledgments

This research was supported by the MKE (The Ministry of Knowledge Economy), Korea, under the ITRC (Information Technology Research Center) support program supervised by the NIPA (National IT Industry Promotion Agency) (NIPA-2010-C1090-1101-0006).


DOI: 10.1587/elex.8.866
Received April 30, 2011
Accepted May 09, 2011
Published June 10, 2011