Adaptive Multi-bit Image Steganography using Pixel-Pair Differential Approach

Uttiya Ghosh*, Debanjan Burman*, Smritikana Maity*, Imon Mukherjee**

[1] *Dept. of Computer Science & Engineering,
St. Thomas' College of Engineering & Technology, Kolkata-700023, India.
{uttiyaghosh, burmandebanjan, smritikanamaity94}@gmail.com
[2] **Dept. of Computer Science & Engineering,
Indian Institute of Information Technology, Kalyani, West Bengal-741235, India.
mukherjee.imon@gmail.com

Abstract. With the increase of communication over internet, the issue of security has become an important factor. Steganography is a consequence of the increasing degradation of reliability. The roots of steganography lies in ancient Greek civilisation. With time steganography has moved a long way in the path of advancement. It started off with least significant bit (LSB) embedding which mainly focussed on data security. With time many algorithms have been designed using multibit steganography which takes into account both security and capacity of data embedded. In this paper a new technique have been introduced where various parameters determine the number of bits embedded. This helps to improve the robustness of this method. We show empirically that our method withstand the statistical attacks and benchmark.

Keywords. Multi-bit Steganography, Information hiding, Spatial domain, Data security.

1 Introduction

The technique of hiding a secret data into another file in a secured manner, such that it can be transmitted from sender to receiver without being suspected by any other party who are not the intended recipients, can be defined as *Steganography*. Mainly digital images are used as cover versions and the image carrying the secret data is known as stego version.

Mandal et al. [3] used pixel value differencing (PVD) method to embed data in every component of a colour image. Moreover the number of bits embedded differ with the different pixel component to improve the security of the embedded data. In the process introduced by Mukherjee et al. [4], frequency domain of the image is used to select the potential pixels whereas the spatial domain is used for embedding data. Veron et al. [10] have shown how algebraic coding theory can be used in various ways to formulate secure steganographic techniques. Yang et al. [11] introduced a technique to improve reversible hiding of data based on histogram of greyscale images. Zhang et al. [12] have formulated a new method where data is hidden in last three bits of cover sample using a triple-layered construction. Paul et al. [5]

have introduced keyless steganographic algorithms based on the lattice spin glass model of Ising [1] of physics. This method has been further improved in [6] where the maximum number of bits embedded in one pixel has been increased to five bits. Song et al. [9] aims at reducing histogram fluctuation by using affine transformation which are special functions that are used to maintain collinearity and ratio of distances.

In this paper a new technique is being presented for embedding data in cover image where bits embedded in each pixel varies based on different parameters. This helps to increase the robustness of this method and security of the embedded data. Our algorithm helps to make it difficult for attackers to retrieve the hidden data and can also withstand various attacks.

2 Proposed Method

In this method we have embedded different number of bits in different pixels considering various parameters. This whole process is described below.

2.1 Embedding of data

The novel method proposed here is used for embedding different number of bits in grayscale images as shown in Table 1 using a dual parameter approach.

Table 1. A grayscale image of size $h \times w$.

$I_{1,1}$	$I_{1,2}$	...	$I_{1,w}$
$I_{2,1}$	$I_{2,2}$	...	$I_{2,w}$
...	...	...	...
$I_{h,1}$	$I_{h,2}$	...	$I_{h,w}$

Table 2. A 4×4 block of a digital image.

$\pi_{1,1}$	$\pi_{1,2}$	$\pi_{1,3}$	$\pi_{1,4}$
$\pi_{2,1}$	$\pi_{2,2}$	$\pi_{2,3}$	$\pi_{2,4}$
$\pi_{3,1}$	$\pi_{3,2}$	$\pi_{3,3}$	$\pi_{3,4}$
$\pi_{4,1}$	$\pi_{4,2}$	$\pi_{4,3}$	$\pi_{4,4}$

We first select non overlapping 4×4 block from Table 1 as shown in Table 2. We introduce a pixel pair differential multiset $d = \{y | y = (\omega_{i,j} \sim \omega_{i,j+1}) \wedge y >= \alpha\}$ from Table 2 where $1 \leq i \leq 4$, $j = \{1, 3\}$ and $0 \leq \alpha \leq 255$. The number of data bits(n) which will be hidden in Table 2 is obtained from Equation (1) where $|d|$ denotes the number of elements in d and $2 \leq \beta \leq 6$.

$$n = \begin{cases} \beta \text{ , if } |d| >= 4; \\ \beta - 1 \text{ , otherwise} \end{cases} \tag{1}$$

Consider a text message with l characters as shown in Table 3.

Table 3. Text message arrangement

T_1	T_2	...	T_i	...	T_l

Table 4. Binary representation of the message

$b_{1,7}$	$b_{1,6}$	...	$b_{1,0}$	$b_{2,7}$	$b_{2,6}$	...	$b_{i,j}$	...	$b_{l,0}$

We then convert the values of Table 3 to the binary representation of their corresponding ASCII code as shown in Table 4 to obtain a 8 bit code for each character where $b_{i,j}$ denotes the j^{th} bit of the i^{th} character. Sequential n bits are taken from Table 4 to form a number m that is to be embedded in the pixels of

Table 2. Let the pixel in which embedding is to take place is denoted as $\pi_{i,j}$. We first adjust the higher order $8 - n$ bits ($\Omega_{i,j}$) to minimize distortion. This is done with the help of Equation (3) by an adjustment factor (δ) obtained from Equation (2) where MSB denotes the most significant bit and abs denotes absolute value.

$$\delta = \lfloor \frac{mod(\pi_{i,j}, 2^n)}{2^{n-1}} \rfloor - MSB(m) \tag{2}$$

$$\Omega^*_{i,j} = \begin{cases} abs(\Omega_{i,j} + \delta) \text{ , where } \lfloor \log_2(\Omega_{i,j} + \delta) \rfloor \leq \lfloor \log_2(\Omega_{i,j}) \rfloor; \\ \Omega_{i,j} \text{ , otherwise.} \end{cases} \tag{3}$$

The value of stego pixel($\pi^*_{i,j}$) is obtained with the help of the Equation (4).

$$\pi^*_{i,j} = \Omega^* \times 2^n + m \tag{4}$$

Finally we store a status bit (S) in the $(\beta + 2)^{th}$ bit of $\pi_{1,1}$ so that it can be used during the retrieval process for determining the value of n. We calculate the value of S from Equation (5).

$$S = \begin{cases} 0 \text{ , if } |d| >= 4; \\ 1 \text{ , otherwise.} \end{cases} \tag{5}$$

The steps for embedding the message are described in Algorithm 1.

```
Input: An image of size $h \times w$ , a text message of length $l$, $\alpha$ and $\beta$.
Output: The stego image containing the secret data.

Convert the text message into a binary stream as shown in Table 4;
$c \leftarrow 0$;
$f \leftarrow 0$;
while ($c \leq 8 \times l \vee f \leq \frac{h \times w}{16}$) do
    Obtain non overlapping $4 \times 4$ block from Table 1 as shown in Table 2;
    $f \leftarrow f + 1$;
    Create multiset $d$;
    Calculate $n$ from Equation (1);
    Sequential $n$ bits are taken from Table 4 to form a number $m$;
    $c \leftarrow c + n$;
    Calculate $\delta$, $\Omega^*_{i,j}$ and $\pi^*_{i,j}$ from Equation (2), (3) and (4) respectively;
    Calculate $S$ from Equation (5) and store in the $(\beta + 2)^{th}$ bit of $\pi_{1,1}$;
end
Output the embedded stego image;
```

Algorithm 1: Algorithm for embedding data bits into the stego image.

2.2 Retrieval of data

The process of retrieving the embedded message from the stego image starts with partitioning the stego image into non-overlapping 4×4 blocks as shown in Table 5. We retrieve status bit (S) from the $(\beta + 2)^{th}$ bit of $\pi^*_{1,1}$. It is to be noted that the value of β must be same during the process of embedding and retrieval. The number of bits(n) that is embedded in Table 5 is obtained from Equation (6).

$$n = \begin{cases} \beta \text{ , if } S = 0; \\ \beta - 1 \text{ , otherwise} \end{cases} \tag{6}$$

Table 5. 4×4 block of stego image.

$\pi^*_{1,1}$	$\pi^*_{1,2}$	$\pi^*_{1,3}$	$\pi^*_{1,4}$
$\pi^*_{2,1}$	$\pi^*_{2,2}$	$\pi^*_{2,3}$	$\pi^*_{2,4}$
$\pi^*_{3,1}$	$\pi^*_{3,2}$	$\pi^*_{3,3}$	$\pi^*_{3,4}$
$\pi^*_{4,1}$	$\pi^*_{4,2}$	$\pi^*_{4,3}$	$\pi^*_{4,4}$

Table 6. Binary representation of the message

b_1	b_2	...	b_i	...	$b_{8\times l}$

We then obtain the value of m from the pixel $\pi^*_{i,j}$ with the help of the Equation (7). Then m is converted into its binary equivalent, concatenated with its corresponding value obtained from $\pi^*_{i,j+1} \cdots \pi^*_{4,4}$ to form Table 6.

$$m = \quad \mathrm{mod}\ (\pi^*_{i,j}, 2^n) \tag{7}$$

Sequential 8 bits are taken from Table 6 and they are converted into their decimal equivalent ASCII code which is finally used to generate the corresponding character of the secret data. The process is repeated throughout the stego image to generate the secret data. The steps for retrieving the message are described in Algorithm 2.

```
   Input: A stego image of size $h \times w$ and $\beta$.
   Output: The hidden data.

1  $f \leftarrow 0$;
2  while ($f \leq \frac{h \times w}{16}$) do
3      Obtain non overlapping $4 \times 4$ block as shown in Table 5;
4      $f \leftarrow f + 1$;
5      Retrieve status bit ($S$) from the $(\beta + 2)^{th}$ bit of $\pi^*_{1,1}$;
6      Calculate $n$ from Equation (6);
7      Calculate $m$ from Equation (7);
8      Convert $m$ into its binary equivalent and add it to the binary stream as shown in Table
       6;
   end
9  Obtain 8 sequential bits from Table 6 and converted into their decimal equivalent ASCII
   code;
10 Repeat Step (9) until the entire message is extracted;
11 Output the secret data;
```

Algorithm 2: Algorithm for extracting hidden data bits from the stego image.

3 Experimental Results

The first and foremost target of any steganographic algorithm is to have the least visible distortion in stego images. The proposed method is tested on various standard images and the results are produced below.

3.1 Visual Perceptibility Analysis

Visual analysis of the tested images can be seen in Figure 1 which depicts that there is almost no visual distortion.

3.2 MSE, BER and PSNR

Mean squared error (MSE) is a statistical quantity that is measured by computing the squares of the errors between the original and stego images and then finding

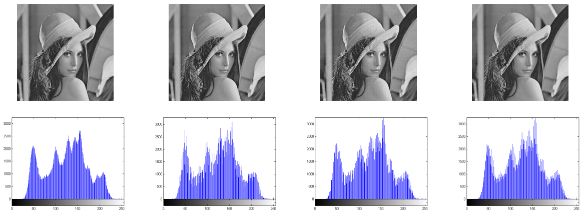

Fig. 1. Cover and stego versions (top row) and their histograms (bottom row) having ($\alpha = 4, \beta = 2$), ($\alpha = 8, \beta = 2$) and ($\alpha = 16, \beta = 2$) of Lena respectively.

their average. It is calculated with the help of Equation (8) where w and h denotes the breadth and height of cover image(say, D) and the stego image (say, D').

$$MSE = \frac{1}{w \times h} \sum_{i=1}^{h} \sum_{j=1}^{w} (D_{i,j} - D'_{i,j})^2 \tag{8}$$

Peak signal-to-noise ratio (PSNR) is a statistical quantity that finds the ratio between the highest possible power of an original signal and the power of debasing noise that affects the reliability of its representation. It is calculated with the help of the Equation (9).

$$PSNR = 20 \log_{10}\left(\frac{MAX}{\sqrt[2]{MSE}}\right) \tag{9}$$

Here MAX is equal to 255 (in case of images).

Bit error rate (BER) is a statistical quantity that measures the number of bit having errors divided by the total number of bits sent or received during a time interval under consideration. It is calculated with the help of the Equation (10).

$$BER = \frac{Total\ number\ of\ bits\ changed}{Total\ number\ of\ bits\ present} \tag{10}$$

The results of these tests are shown in Table 7.

Table 7. NC, MSE, PSNR, BER and Capacity values for sample Lena, Baboon and Cameraman images.

Name		**NC**			**BER**			**MSE**			**PSNR**			**Capacity (in bpp)**		
		$\beta = 2$	$\beta = 3$	$\beta = 4$	$\beta = 2$	$\beta = 3$	$\beta = 4$	$\beta = 2$	$\beta = 3$	$\beta = 4$	$\beta = 2$	$\beta = 3$	$\beta = 4$	$\beta = 2$	$\beta = 3$	$\beta = 4$
Lena	$\alpha = 4$	1.008	1.0039	1.0026	1.87	1.97	2.09	5.38	6.31	7.98	40.82	40.13	39.11	1.5539	2.5539	3.5539
	$\alpha = 8$	1.0015	1.0028	1.0097	1.92	2.11	2.21	6.18	6.58	8.93	40.22	39.95	38.62	1.2458	2.2458	3.2458
	$\alpha = 16$	1.0008	1.0006	1.0012	2.05	2.27	2.35	6.74	8.71	12.13	39.84	38.73	37.29	1.0965	2.0965	3.0965
Baboon	$\alpha = 4$	0.9982	1.0001	1.0141	1.93	2.41	2.90	10.28	11.17	13.24	38.01	37.65	36.91	1.8346	2.8346	3.8346
	$\alpha = 8$	1.0003	1.0008	1.0115	1.78	2.16	2.75	8.65	9.79	10.71	38.76	38.22	37.83	1.5516	2.5516	3.5516
	$\alpha = 16$	1.0029	1.0016	1.0086	1.62	2.20	2.60	7.98	8.18	9.20	39.11	39.00	38.49	1.2195	2.2195	3.2195
Camer-aman	$\alpha = 4$	1.0025	1.0014	1.0069	1.17	2.53	2.76	7.83	8.55	9.50	38.01	37.65	36.91	1.2856	2.2856	3.2856
	$\alpha = 8$	1.0034	1.0017	1.0058	1.85	2.24	2.69	6.54	9.79	8.31	38.76	38.22	37.83	1.1616	2.1616	3.1616
	$\alpha = 16$	1.0040	1.0019	1.0052	1.73	2.33	2.57	6.05	8.18	7.74	39.11	39.00	38.49	1.0808	2.0808	3.0808

3.3 Normalized Correlation

Normalized Corelation (NC) denotes the extent of similarity between a cover image and its stego image. As the difference between the two versions tend to 0, its value tends to 1. The value of NC is obtained from the Equation (11).

$$NC = \frac{\sum_{u=1}^{p} \sum_{v=1}^{q} [D(u,v).D^*(u,v)]}{\sum_{u=1}^{p} \sum_{v=1}^{q} D(u,v)^2} \tag{11}$$

where $D(u,v)$ denotes the pixel intensities of the cover image and $D^*(u,v)$ denotes the pixel intensities of stego image. The values of NC for various images are shown in Table 7.

3.4 Histogram Analysis

Histogram of of an image can be defined as the graphical representation of the frequency of every pixel value versus the pixel value. Histogram is one of those basic tools of quality control which gives an overview of the pixel value distribution of the image at one glance. It gives the frequency distribution of each distinct value in a set of data, which is the number of pixels for each intensity value. The histograms of the cover and the stego versions of different standard images are shown in Figure 1.

3.5 StirMark Analysis

Along with the efficiency, the robustness of all steganographic algorithms requires a proper analysis with respect to some standardized methods. These analyses were performed using StirMark 4.0 as described in [7], [8].
our proposed method was successful in providing some excellent results. The very minute deviation between the values of the original and the stego image shown in Table 8 is an indication to the robustness of the proposed algorithm.

Table 8. StirMark analysis of proposed technique on cover and stego version of Baboon (512×512)

Test Name		Cover	Stego		
			$\beta = 2$	$\beta = 3$	$\beta = 4$
Add Noise (Factor=20)	$\alpha = 4$	7.59261	7.58297	7.60819	7.51232
	$\alpha = 8$		7.5767	9.20584	7.53868
	$\alpha = 16$		7.58218	9.14098	7.54559
PSNR (Factor=10)	$\alpha = 4$	37.6544	37.7299	37.6922	38.2764
	$\alpha = 8$		37.7299	37.485	38.5543
	$\alpha = 16$		37.7674	37.3956	38.5544
Median Cut (Factor=5)	$\alpha = 4$	23.3523	23.316	23.0922	22.4585
	$\alpha = 8$		23.3221	20.4089	22.4511
	$\alpha = 16$		23.2916	20.4253	22.5601
Conv Filter (Factor=1)	$\alpha = 4$	9.37115	9.37031	9.36836	9.36589
	$\alpha = 8$		9.37152	9.07607	9.36597
	$\alpha = 16$		9.37175	9.07542	9.3684

Table 9. Comparison of Capacity and PSNR with existing techniques applied on images of size 512 × 512.

Existing Technique	Avg. Capacity (bits)	Avg. PSNR (dB)
A [2]	2097152	41.026
B [3]	1161915	41.663
C [13] (Threshold value 12)	370000	36.19
C [13] (Threshold value 16)	120000	33.20
C [13] (Threshold value 20)	260000	32.27
D [11] (Column Method)	64882	48.72
D [11] (Chessboard Method)	71462	48.80
Proposed Method	2359296	38.196

4 Comparison with Existing Algorithms

Several attributes of the original image and stego image obtained by the proposed technique is calculated for visual quality analysis viz. bit error rate (BER), mean square error (MSE), embedding efficiency and Peak Signal-to-Noise ratio (PSNR). This method is compared with various other existing methods. The fact that our algorithm is better can be seen from the comparative study shown in Table 9.

5 Conclusion

This paper introduces a novel steganographic algorithm where the number of bits embedded in each pixel varies based on the value of α and β. This helps to increase the robustness of the algoirthm along with increasing the security of secret data. Due to the change of value of α and β, various other parameters also changes. It allows the user to choose which parameter he or she wants to focus on while transmitting the secret data. This method can successfully withstand several attacks as shown earlier.

References

1. Cipra, B.: An Introduction to the Ising Model, In: American Mathematical Monthly, Vol. 94, No. 10, pp. 937-959 (December 1987).
2. Ghosh., U., Maity, S., Mukherjee, I.: Statistical Attack Resistant Multi-Bit Steganography Using Mobile Keypad Character Encoding, In: International Conference on Telecommunication Technology and Management (ICTTM-2015), IIT Delhi, India, ISBN: 987-0-9926800-5-3, pp. 19 (April 11-12, 2015).
3. Mandal, J.K., Das, D.: Colour Image Steganography Based on Pixel Value Differencing in Spatial Domain, In: International Journal of Information Sciences and Techniques (IJIST), Vol.2, No.4, pp. 83-93 (July 2012).
4. Mukherjee, I., Podder, A.: DCT Based Robust Multi-bit Steganographic Algorithm, In: 2nd International Conference on Advanced Computing, Networking and Informatics, Vol. 28, Springer (June 2014).
5. Paul, G., Davidson, I., Mukherjee, I., Ravi, S.S.: Keyless Steganography in Spatial Domain using Energetic Pixels, In: International Conference on Information Systems Security (ICISS), Vol. 7671, pp. 134-148 (December 2012).
6. Paul, G., Davidson, I., Mukherjee, I., and Ravi, S, S.: Keyless Dynamic Optimal Multi-bit Image Steganography using Energetic Pixels, In: Multimedia Tools and Applications, Vol. 75, pp. 1-27, Springer (2016), doi: 10.1007/s11042-016-3319-0
7. Petitcolas, F.A.P., Anderson, R.J., Kuhn, M.G.: Attacks on copyright marking systems, In: Second International Workshop on Information Hiding, IH98, Portland, Oregon, U.S.A., Proceedings, LNCS 1525, Springer-Verlag, ISBN 3-540-65386-4, pp. 219-239 (April 15-17, 1998).
8. Petitcolas, F.A.P.: Watermarking schemes evaluation, In: IEEE Signal Processing, Vol. 17, No. 5, pp. 5864 (September 2000).
9. Song, X., Wang, S., Niu, X.: An Integer DCT and Affine Transformation Based Image Steganography Method, In: IEEE, 2012 Eighth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, pp. 102 - 105 (18-20 July 2012).
10. Veron, P.: Code Based Cryptography and Steganography, In: 5th International Conference on Algebraic Informatics, LNCS, Springer, ISBN 978-3-642-40662-1, Vol.8080, pp. 9-46 (September 2013), doi: 10.1007/978-3-642-40663-8_5
11. Yang, C.H., Tsai, M.H.: Improving Histogram-based Reversible Data Hiding by Interleaving Predictions, In: IET Image Processing, Vol. 4, No.4, pp. 223-234 (August 2010), doi: 10.1049/iet-ipr.2009.0316
12. Zhang, X.: Efficient Data Hiding With Plus-Minus One or Two, In: IEEE Signal Processing Letters, Vol. 17, No. 7, pp. 635-638 (July 2010).
13. Zhang, X., Wang, S., Zhou, Z.: Multi-bit Assignment Steganography in Palette Images, In: IEEE Signal Processing Letters, vol. 15, pp. 553-556 (2008), doi: 10.1109/LSP.2008.2001117