A Reversible Data Hiding Scheme Using Pixel Location

Rajeev Kumar, Satish Chand, and Samayveer Singh
Division of Computer Engineering, Netaji Subhas Institute of Technology, India

Abstract: *In this paper, authors propose a new reversible data hiding scheme that has two passes. In first pass, the cover image is divided into non-overlapping blocks of 2×2 pixels. The secret data bit stream is converted into 2-bit segments, each representing one of the four values, i.e., 0,1,2,3 and these digits (2-bit segments) are embedded into blocks by increasing/decreasing the pixel value of the block by 1. If the pixel is even valued, then the pixel is increased otherwise it is decreased by 1 to embed the secret data. In second pass, the same process of the first pass embedding is repeated. The second pass embedding helps in achieving better stego-image quality and high data hiding capacity because some of the pixels changed in first pass are recovered to their original form. Basically, the second pass is a complement of the first pass. This scheme can achieve approximately 1 bpp data hiding capacity and more than 55db Peak Signal-to-Noise Ratio (PSNR) for all cover images in our experiments. For ensuring reversibility of the scheme, a location map for each phase is constructed and embedded into the image. Though, the scheme has some overhead in hiding the secret data, yet it provides good quality with high capacity. Since it only increases/decreases the pixel value of at most half of the pixels, it is very simple. The experimental results show that it is superior to the state of the art schemes.*

Keywords: *Reversible data hiding, pixel location, location map, non-overlapping blocks.*

Received January 30, 2015; accepted July 17, 2015

1. Introduction

Data hiding is a process of imperceptibly embedding the secret data into a cover media. When data hiding is performed in a reversible manner, the original cover contents can be entirely restored after data extraction at the receiver side. The usage of reversible data-hiding schemes reduces transmission cost and also increases the security of the information being transmitted. In principle, the reversible data hiding schemes can be classified into three types of domain: spatial domain, frequency domain, and compressed domain. In the spatial domain based schemes, the pixel values are modified directly to embed the secret data [8, 10]. In the frequency domain based schemes, the image pixels are transformed into coefficients by applying some transformations such as discrete cosine transform, wavelet transform [6]. These coefficients are modified in order to embed the secret data. In the compressed domain based schemes, the original image is first compressed using some popular compression algorithm, such as vector quantization [4], block truncation coding [5], that results in compressed codes (compressed image), which is used to hide the secret data. Among these three domains, the spatial domain based techniques are inevitably the least complex techniques because the process of embedding the secret data is directly applied to the pixel values whereas both the others domain (compressed and transform) based techniques first transform/compress the cover media before embedding the secret data which adds some extra computation. The simplest approach used in the spatial domain based techniques is simple least significant bit substitution method in which least significant bit(s) of pixels is/are replaced by the bits of the secret data [18]. However, the scheme does not perform up to the expectations if the amount of secret data to be embedded is very large i.e., more than the number of pixels in which secret data to be hidden. There have been many improvements to overcome its limitations that include Optimal Pixel Adjustment Process (OPAP), exhaustive Least Significant Bit (LSB) etc., [7, 11]. The other important approaches in spatial domain include Difference Expansion (DE), predictive coding, and histogram shifting for embedding the secret data [14]. The difference expansion based techniques [1, 17] embed the secret data into the difference of the pixels values. The techniques have low embedding capacity. The prediction based techniques [16] forecast the pixel values and embed the secret data into the difference of forecasted value and the original pixel value. The histogram shifting techniques embed the secret data using two phases: it first finds the peak and zero points in a cover image using histogram, and then empties the peak points by shifting the bins. Afterwards, the secret data is embedded by making some adjustment in peak point and the emptied one. Basically, histogram shifting techniques have low embedding capacity.

In this paper, authors propose a reversible data hiding scheme, which provides high quality stego-image along with high data hiding capacity. The reason for considering spatial domain is that the methods in spatial domain have less computational complexity. The proposed scheme embeds the secret data in the cover image in layers in order to increase the hiding capacity. A layer consists of two passes so that the distortion gets reduced. In pass one, the cover image is divided into blocks of size 2×2 pixels and one of these pixels hide the two bit secret data by modifying that pixel value by ±1. The pixel is decided by the 2-bit segment value of the secret data. In second pass, the same process is repeated, which again modifies the pixel value by ±1, to hide the secret data. These two passes combined either modify two different pixels by ±1or nullify their effects. This is applicable for any number of layers; increasing the number of layers increases the hiding capacity, without deteriorating the image quality.

The rest of the paper is organized as follows. Section 2 discusses the related works. Section 3 discusses our proposed work and, in section 4, the experimental results are discussed. Finally, in section 5, the paper is concluded.

2. Previous Works

In this section, we briefly review the important and relevant methods for reversible data hiding. The concept of reversible data hiding came into existence by Barton [2]. This scheme recovers the original cover image, but sometimes proper restoration of the cover image and extraction of the secret data is not possible because of the underflow and overflow problems. These problems later addressed by the Honsinger *et al*. [9] method. Tian [17] introduces a data hiding method which uses difference of a couple of pixels to hide the secret data. This method provides decent quality stego-image, but suffers from low embedding capacity. Alattar [1] overcomes the shortcoming of the Tian's method [17] by generalizing the difference expansion of Tian's method. This scheme is also known as Generalized Difference Expansion (GDE) scheme. Thodi and Rodriguez [16] discuss a predictor based scheme which is basically based on the correlation of the neighboring pixels. It also uses histogram shifting scheme for embedding the secret data into the cover image. Tai *et al*. [15] discuss a histogram based reversible data hiding scheme which uses the differences between adjacent pixels. It leverages a binary tree structure to avoid the problem of communicating pairs of peak points. The distribution of pixel differences is utilized in such a way so that high embedding capacity is achieved. The problem of overflow and underflow are solved using histogram shifting strategy. However, both high hiding capacity and good quality stego image are not achieved at the same time using this scheme. Weng *et al*. [20] introduced a reversible data hiding scheme which divides the original image into blocks: smooth and complex. The category of the block is decided through the inter-pixel correlation. This scheme hides more bits in the smooth blocks as compared to the complex blocks. Still, it suffers from hiding capacity perspective when complex images are encountered. Lin *et al*. [12] discuss weighted prediction based reversible data hiding scheme which also uses histogram shifting strategy. Here, this scheme tries to provide the optimal weights using the solution of least square problem so that distortion is reduced. Though it achieves good stego-image quality, yet its hiding capacity is not up to the mark. Zhao *et al*. [21] introduce multi-level histogram based reversible data hiding scheme which is basically designed for natural images. In natural images, the difference of neighborhood pixels is negligible or close to zero. This scheme is able to use more peak points for secret bits modulation and thus the hiding capacity is increased. Chung *et al*. [8] discuss a histogram modification-based reversible data hiding scheme which minimizes the distortion of the stego-image. Lu *et al*. [13] introduced a hybrid method for reversible data hiding using difference expansion, interpolation and histogram shifting methods. It hides the secret data into the difference of interpolated pixels and embeddable pixels. It achieves high hiding capacity. Wang *et al*. [19] propose a histogram based reversible data hiding scheme which basically updates the peak points of segments based on image intensity. It replaces the peak points, pixels with another pixel value of the same segment for embedding the secret data. A location map is also designed so that precise extraction of the secret data and restoration of the original image is possible. Succinctly, we find that this scheme provides good quality stego-image, but fails in achieving high embedding capacity at the same time. As discussed above, the existing techniques either suffers from embedding capacity perspective or stego-image quality perspective. In our proposed scheme, our focus is to provide both high embedding capacity and good stego-image quality at the same time.

3. Proposed Method

In proposed scheme, a Location Map (LM1) i.e., a sequence of bits, to know about the characteristics of the original image in terms of even or odd valued pixels, is constructed by scanning the image as detailed in step (1) of embedding algorithm.

In LM1, the even valued pixels are represented by '0' and the remaining pixels as '1'. The LM1 is compressed using algorithm [3] to obtain the Compressed Location Map (CLM1), denoting its length as LCLM1 as detailed in step (2). The LM*i* or CLM*i* or LCLM*i* refers to i^{th} pass of a layer. The CLM1 is appended with a unique bit sequence to mark the end of the location map, which will help at extraction time. The secret data bit stream is divided into 2-bit segments, representing four decimal values:

0(00), 1(01), 2(10), and 3(11). The sender hides the secret data into a cover image in two passes as detailed in step (3). In first pass, the cover image is divided into non-overlapping blocks, each of size 2×2 pixels, denoting the upper left, upper right, lower left, and lower right pixel as P_0, P_1, P_2, and P_3, as shown in Figure 1. One block hides only one segment and the pixel is decided by the segment value.

P_0	P_1	P_0	P_1
P_2	P_3	P_2	P_3
P_0	P_1	P_0	P_1
P_2	P_3	P_2	P_3

Figure 1. Image partitioning method.

If the chosen pixel value is even, its value is increased by 1, otherwise decreased by 1 as detailed in step (5). Now, extract the LSBs of LCLM1 number of pixels and divide it into 2-bit segments. These segments are embedded into subsequent blocks using the same approach as used for hiding the secret data as detailed in step (6). The CLM1 is embedded onto the first LSBs of LCLM1 number of pixels of the resultant image using simple LSB substitution method as detailed in step (7). In second pass, we repeat the same process as done in first pass to further hide the secret data. Performing second pass may recover the original value of some pixels, leading to better stego-image quality, and it further increases the hiding capacity. Thus, it provides a good quality stego-image with high hiding capacity.

3.1. Embedding Algorithm

Algorithm 1: Embedding Algorithm

Input *I: grayscale image of size H × W pixels*
S: Secret Data bit stream
Output *Z: Stego-image*

BEGIN

1. *Scan cover image in raster scan manner and create a location map (LM1) in which '0' bit refers even valued pixels and '1' odd valued pixels.*
2. *Compress LM1 using algorithm [3] to get compressed location map CLM1.*
3. *Divide secret data in 2-bit segments, which are equally hidden in first pass & second pass, denoting them as S_1 and S_2.*
4. *Divide image I into non-overlapping blocks of size 2×2 pixels in raster scan manner (refer Fig. 1).*
5. *Embed S_1 into image I in first pass to get image T as mentioned below:*
   For si $\in S_1$
   //S_1 is secret data, s_i is 2-bit ith segment of S_1.
   Take image block bi $\in$ I
   Get pixel p_j (j=0,1,2,3) in b_i corrosponing to s_i value.
   // bi is ith block of image I.
   If p_j is divisible by 2, then
   $p_j = p_j + 1$;
   Else
   $p_j = p_j - 1$;
   End //if
   End //for
6. *Extract first LSB of first LCLM1 pixels in raster scan manner and divide it into 2-bit segments. Embed these segments using step (5) on remaining pixels.*
7. *Embed CLM1 into first LCLM1 pixels of resultant image using simple LSB substitution to get image P.*
8. *Scan cover image in raster scan manner and create a location map (LM2) in which '0' bit represents even valued pixels and '1' odd valued pixels.*
9. *Compress LM2 using algorithm [3] to get compressed location map CLM2.*
10. *Divide image P into non-overlapping blocks, each of size 2×2 pixels in raster scan manner (refer Fig. 1).*
11. *Embed S_2 into P in second pass to get image Y, i.e.*
    For si $\in S_2$
    Take image block bi $\in$ P
    Get pixel p_j (j=0,1,2,3) in b_i corrosponing to s_i.
    If p_j is divisible by 2, then
    $p_j = p_j + 1$;
    Else
    $p_j = p_j - 1$;
    End //if
    End //for
12. *Extract first LSB of first LCLM2 pixels in raster scan manner and divide it into 2-bit segments. Embed these segments using step (11) in remaining pixels.*
13. *Embed CLM2 into first LCLM2 pixels of resultant image using simple LSB substitution to get image Z.*

END

The obtained stego-image is then transmitted to receiver.

3.2. Extraction Algorithm

On receiver side, the proposed extraction algorithm is applied to recover the cover image and extract the secret data. The extraction algorithm basically works in reverse manner of the embedding algorithm. It also consists of two passes. In first pass, the receiver first gets the CLM2 by extracting the first LSBs of the pixels of the stego image Z in raster scan manner. It is to mention that the CLM2 had been terminated with a unique bit sequence in pass two of embedding process (which helps to extract CLM2 length). Let the last pixel be denoted as PL. Using decompression algorithm [3], the CLM2 is decompressed to obtain LM2. The stego-image is divided into non-overlapping blocks of size 2×2 pixels. The LCLM2 number of bits are extracted from the pixels staring from PL using LM2 and extraction rules (given below). the first LCLM2 pixels are recovered. The secret data is extracted using LM2 and the extraction rule, starting from the very first block. In second pass of extraction, the same process of the first pass is followed to get the complete secret data S and original image I.

3.2.1. Extraction Rule

For each block b_i of image
Obtain location map of block b_i. // it identifies modified pixel.
Identify modified pixel p_j.
If p_j is divisible by 2
$p_j = p_j + 1$;

Else
$p_j = p_j - 1$;
End //if
$s_i = j$; //binary form of s_i has 2 bits secret data.
End//For

4. Experimental Results and Discussions

The performance of the proposed scheme is evaluated in terms of Peak Signal-to-Noise Ratio (PSNR) and hiding capacity. The PSNR measures the quality of a stego-image and the hiding capacity measures the embedded bits per pixel. The scheme has been implemented in MATLAB7.8.0 running on the Intel® Core 2 Duo 2.20 GHz CPU, and 3GB RAM hardware platform with Windows 7 operating system. The PSNR and hiding capacity (in bpp) are given as follows.

$$PSNR = 10\log_{10}\left[\frac{255\times 255}{MSE}\right] \quad (1)$$

where is Mean Square Error (MSE) which is calculated average error between the stego-image and cover image.

$$bpp = \frac{N_{sb}}{P} \quad (2)$$

where, bpp is number of bits embedded into a pixel, N_{sb} is total number of bits embedded into image and P is total number of pixels in cover image.

a) Lena. b) Baboon.
c) Jet. d) Barbara.
e) oat. f) Goldhill.

Figure 2. Cover images, each of size 512x512.

The proposed scheme does not encounter the problem of overflow/underflow as it increases/decreases the pixels values according to its present value. If the pixel is even valued, it increases the pixel value; otherwise decreases the pixel value by 1 for hiding the secret data. In case the pixel value is 0, its value will only be increased and when the pixel value is 255, its value will only be decreased regardless the secret data bits; thus, there is no overflow/underflow problems in our scheme. To analyze the performance of the proposed scheme, we the six commonly used cover images of size 512×512 pixels each as shown in Figures 2(a, b, c, d, e, and f) are used.

The performance of the scheme is compared with that of the Tian's method [17], Alattar's method [1], Thodi and Rodriguez's method [16], Zhao *et al*. [21] Method, and Wang *et al*. [19] Method. The reason for comparing with these method is that they are also spatial domain based schemes as is proposed, and they are quite recent published methods. The results are taken on the maximum hiding capacity for all the schemes. The secret data to be embedded has been generated using random function. The percentage increase for both the parameters: bpp and PSNR for the proposed scheme is calculated with respect Wang *et al*. [19] Method because among all the schemes [1, 16, 17, 21] used for comparison, the Wang *et al*. [19] Method has the best performance. The proposed scheme can hide the secret data in layers that further increases its hiding capacity. The results for embedding in layers from 1 to 4 are illustrated in Tables 1, 2, 3, and 4 for analysis purpose.

a) Lena. b) Baboon.
c) Jet. d) Barbara.
e) Boat. f) Goldhill.

Figure 3. Stego-images (one layer of embedding).

Table 1 shows the results for layer 1 embedding for all the cover images. The proposed scheme achieves 6.41% to 10.64% increase in PSNR and 71.42% to 84.61% increase in bpp as compared to [19]. Thus, the scheme performs much better than all the other recently developed methods. It achieves more than 55 db PSNR for all the cover images. The stego-images of one layer embedding are shown in Figures 3(a-f). This method achieves very good quality because some of the pixel values modified during the first pass of embedding are converted to their original value during the second pass of embedding. Basically, it hides two bits of the secret data into a block of the pixels by changing only one pixel's value of the block by 1 in every pass. In each pass of embedding, the even valued pixels are increased by 1 and the odd valued pixels are decreased by 1. It may be noted that applying two passes of embedding, some pixels may get their original values. If a pixel is not used for hiding in either pass or it is used in both passes of embedding, its value will remain unchanged. If a pixel is used for hiding the secret data in either pass, its value will be decreased or increased by 1 at most. If a pixel is used for data hiding two times in any different layers, it will get its original value. Thus, increasing the number of layer will not deteriorate the image quality.

The proposed scheme gets better PSNR values for complex as well as smooth images as compared to the methods [1, 16, 17, 19, 21] because this method does not depend on the characteristics of the cover image. For layer 2 embedding, the PSNR value varies in the range from 5.73% to 14.94% and bpp in the range from 47.69% to 53.60% with respect to scheme [19] as shown in Table 2. The Minimum and maximum increase in PSNR values are 8.88 and 13.60 for layer 3 embedding and 16.48 and 19.66 for layer 4 embedding as shown in Tables 3 and 4, respectively.

Table 1. Comparison of PSNR, and bits per pixel embedding capacity for one layer embedding.

Cover Image		**Tian's method [17]**	**Alattar's method [1]**	**Thodi & Rodriguez's method [16]**	**Zhao *et al.* Method [21]**	**Wang *et al.* Method [19] (A)**	**Proposed Scheme (B)**	**Percentage increment (B-A)/A %**
Lena	PSNR	34.35	32.68	32.75	44.86	51.71	55.21	6.76
	bpp	0.49	0.74	0.99	0.27	0.54	0.96	77.77
Baboon	PSNR	29.92	25.25	24.40	44.32	50.75	55.12	8.61
	bpp	0.49	0.74	0.99	0.10	0.52	.96	84.61
Jet	PSNR	33.25	30.33	32.67	45.14	51.49	55.12	7.04
	bpp	0.49	0.74	0.99	0.38	0.56	.96	71.42
Barbara	PSNR	26.80	26.52	24.98	44.53	52.03	55.37	6.41
	bpp	0.49	0.73	0.99	0.18	0.54	.96	77.77
Boat	PSNR	31.41	29.88	31.10	44.34	49.81	55.11	10.64
	bpp	0.49	0.74	0.99	0.17	0.55	.96	74.54
Goldhill	PSNR	33.16	30.73	30.42	44.65	51.65	55.09	6.66
	bpp	0.48	0.74	0.99	0.20	0.52	.96	84.61

Table 2. Comparison of PSNR, and bits per pixel embedding capacity with 2 layer embedding.

Cover Image		**Tian's method [17]**	**Alattar's method [1]**	**Thodi & Rodriguez's method [16]**	**Zhao *et al.* Method [21]**	**Wang *et al.* Method [19] (A)**	**Proposed Scheme (B)**	**Percentage increment (B-A)/A %**
Lena	PSNR	30.90	25.18	24.70	41.38	50.31	54.33	7.99
	bpp	0.99	1.47	1.99	0.42	1.27	1.92	51.18
Baboon	PSNR	22.68	19.12	16.93	40.49	49.34	54.53	10.51
	bpp	0.99	1.37	1.93	0.16	1.29	1.92	48.83
Jet	PSNR	29.10	24.37	24.30	42.00	50.12	54.19	8.12
	bpp	0.99	1.47	1.99	0.54	1.30	1.92	47.69
Barbara	PSNR	23.18	21.21	18.25	40.90	51.23	54.17	5.73
	bpp	0.98	1.36	1.93	0.29	1.28	1.92	50.00
Boat	PSNR	28.18	22.93	22.36	40.98	47.56	54.67	14.94
	bpp	0.99	1.45	1.98	0.33	1.29	1.92	48.83
Goldhill	PSNR	28.26	22.52	20.71	41.00	50.12	54.39	8.51
	bpp	0.99	1.47	1.98	0.32	1.25	1.92	53.60

5. Conclusions

In this paper, a new reversible data hiding scheme using pixel location is discussed. It divides a cover image of 2×2 pixel blocks and the secret data into 2-bit segments. One block hides one segment of secret data. If an even valued pixel is used to hide a segment of the secret data, its value is increased by one and if an odd valued pixel is used to hide a segment of the secret data, its value is decreased by 1. If a pixel is used to hide even number of segments of the secret data, its value remains unchanged, which helps in maintaining good quality of the stage-image. This scheme can also be applied on color images by operating on individual color components. Hence, it is applicable to both the grayscale and color images.

Table 3. Comparison of PSNR, and bits per pixel embedding capacity with 3 layer embedding.

Cover Image		**Tian's method [17]**	**Alattar's method [1]**	**Thodi & Rodriguez method [16]**	**Zhao *et al.* Method [21]**	**Wang *et al.* Method [19] (A)**	**Proposed Scheme (B)**	**Percentage increment (B-A)/A %**
Lena	PSNR	24.63	20.84	18.48	39.20	48.86	53.89	10.29
	bpp	1.49	2.13	2.95	0.54	1.89	2.88	52.38
Baboon	PSNR	18.23	16.49	12.82	37.87	47.76	53.92	12.89
	bpp	1.45	1.78	2.70	0.23	1.87	2.88	54.01
Jet	PSNR	24.25	20.81	17.98	39.98	48.36	53.82	11.29
	bpp	1.49	2.13	2.95	0.65	1.87	2.88	58.24
Barbara	PSNR	19.51	18.80	14.13	38.49	49.54	53.94	8.88
	bpp	1.42	1.86	2.77	0.39	1.82	2.88	58.24
Boat	PSNR	22.65	19.52	15.92	38.63	46.21	53.83	16.48
	bpp	1.47	2.06	2.90	0.423	1.89	2.88	52.38
Goldhill	PSNR	22.06	17.96	14.10	38.62	48.12	53.87	11.94
	bpp	1.4895	2.0971	2.8573	0.4260	1.84	2.88	56.52

Table 4. Comparison of PSNR, and bits per pixel embedding capacity with 4 layer embedding.

Cover Image		**Tian's method [17]**	**Alattar's method [1]**	**Thodi & Rodriguez's method [16]**	**Zhao *et al.* Method [21]**	**Wang *et al.* Method [19] (A)**	**Proposed Scheme (B)**	**Percentage increment (B-A)/A %**
Lena	PSNR	21.36	17.71	13.11	37.68	47.02	53.72	14.24
	bpp	1.9733	2.6668	3.8004	0.6383	2.66	3.84	44.36
Baboon	PSNR	15.61	15.15	10.61	35.95	46.12	53.65	16.32
	bpp	1.8402	2.0054	3.2068	0.2894	2.70	3.84	42.22
Jet	PSNR	20.97	18.07	13.30	38.58	46.67	53.68	15.02
	bpp	1.9710	2.7092	3.7201	0.7223	2.76	3.84	39.13
Barbara	PSNR	17.43	16.90	10.89	36.74	47.32	53.76	13.60
	bpp	1.8379	2.2273	3.3775	0.4655	2.81	3.84	36.65
Boat	PSNR	19.42	17.36	11.49	36.97	44.81	53.62	19.66
	bpp	1.9465	2.5218	3.5655	0.5355	2.82	3.84	36.17
Goldhill	PSNR	18.04	15.62	10.88	36.93	46.12	53.61	16.24
	bpp	1.9567	2.4677	3.3875	0.5130	2.83	3.84	35.68

References

[1] Alattar A., "Reversible Watermark using the Difference Expansion of a Generalized Integer Transform," *IEEE Transactions on Image Processing*, vol. 13, no. 8, pp. 1147-1156, 2004.

[2] Barton J., "Method and Apparatus for Embedding Authentication Information Within Digital Data," US Patent 5 646 997, 1997.

[3] Carpenter B., Compression Via Arithmetic Coding,http://www.colloquial.com/ArithmeticCoding/, Last Visited, 2002.

[4] Chang C. and Lin C., "Reversible Steganographic Method using SMVQ Approach based on Declustering," *Information Sciences*, vol. 177, no. 8, pp. 1796-1805, 2007.

[5] Chang C., Lin C., and Fan Y., "Lossless Data Hiding for Color Images Based on Block Truncation Coding," *Pattern Recognition*, vol. 4, no. 7, pp. 2347-2357, 2008.

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

[7] Chan C. and Cheng L., "Hiding Data in Images by Simple LSB Substitution," *Pattern Recognition*, vol. 37, no. 3, pp. 469-474, 2004.

[8] Chung K., Huang Y., Yan W., and Teng W., "Distortion Reduction for Histogram Modification-based Reversible Data Hiding," *Applied Mathematics and Computation*, vol. 218, no. 9, pp. 5819-5826, 2012.

[9] Honsinger C., Jones P., Rabbani M., and Stoffel J., "Lossless Recovery of An Original Image Containing Embedded Data," US Patent, 2001.

[10] Kumar R. and Chand S., "A Reversible High Capacity Data Hiding Scheme using Pixel Value Adjusting Feature," *Multimed Tools and Applications*, vol. 75, no. 1, pp. 241-259, 2014.

[11] Kumar R., Saini K., and Chand S., "A New Steganography Technique using Snake Scan Ordering Strategy," *International Journal Image, Graphics and Signal Processing*, vol. 6, pp. 25-32, 2013.

[12] Lin S., Huang C., Liou M., and Chen C., "Improving Histogram-Based Reversible Information Hiding by an Optimal Weight-based Prediction Scheme," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 4, no. 1, pp. 19-33, 2013.

[13] Lu T., Chang C., and Huang Y., "High Capacity Reversible Hiding Scheme based on Interpolation, Difference Expansion, and Histogram Shifting," *Multimedia Tools and Application*, vol. 72, no. 1, pp. 417-435, 2014.

[14] Ramaswamy R. and Arumugam V., "Lossless Data Hiding Based on Histogram Modification," *The International Arab Journal of Information Technology*, vol. 9, no. 5, pp. 445-51, 2012.

[15] Tai W., Yeh C., and Chang C., "Reversible Data Hiding Based on Histogram Modification of Pixel Differences," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 19, no. 6, pp. 906-910, 2009.

[16] Thodi D. and Rodriguez J., "Expansion Embedding Techniques for Reversible Watermarking," *IEEE Transactions on Image Processing*, vol. 16, no. 3, pp. 721-730, 2007.

[17] Tian J., "Reversible Data Embedding using a Difference Expansion," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890-896, 2003.

[18] Wang R., Lin C., and Lin J., "Image Hiding by Optimal LSB Substitution and Genetic Algorithm," *Pattern Recognition*, vol. 34, pp. 671-683, 2001.

[19] Wang Z., Lee C., and Chang C., "Histogram-Shifting-imitated Reversible Data Hiding," *Journal of Systems and Software*, vol. 86, no. 2, pp. 315-323, 2013.

[20] Weng S., Chu S., Cai N., and Zhan R., "Invariability of Mean Value Based Reversible Watermarking," *Journal of Information Hiding and Multimedia Signal Processing*, vol. 4, no. 2, pp. 90-98, 2013.

[21] Zhao Z., Luo H., Lu Z., and Pan J., "Reversible Data Hiding based on Multilevel Histogram Modification and Sequential Recovery," *International Journal of Electronics and Communications*, vol. 65, no. 10, pp. 814-826, 2011.

Rajeev Kumar received his B. Tech. in Information Technology from Utter Pradesh Technical University, Lucknow, India and M. Tech. in Information Systems from Netaji Subhas Institute of Technology, Delhi University, India. He is pursuing his doctoral degree is at the Computer Engineering Department, Netaji Subhas Institute of Technology, Delhi University, Delhi, India. His research areas lie in the area of data hiding and image processing.

Satish Chand received the Master of Science in Mathematics from Indian Institute of Technology, Kanpur, India, Master of Technology in Computer Science from Indian Institute of Technology, Kharagpur, India and PhD in Computer Science from Jawaharlal Nehru University, Delhi, India. Currently he is working as Professor in Computer Engineering Department, Netaji Subhas Institute of Technology, Delhi, India. His research interests include image processing, data hiding, video processing, and wireless sensor networks.

Samayveer Singh received his B. Tech. in Information Technology from Uttar Pradesh Technical University, Lucknow, India in 2007 and his M. Tech. in Computer Science & Engineering from National Institute of Technology, Jalandhar, India, in 2010. He is pursuing his Ph.D. in the Department of Computer Engineering, Netaji Subhas Institute of Technology, New Delhi, India. His research interest includes wireless sensor networks, data hiding, and information security.