



Implementation of Histogram Modification Method for Reversible Data Hiding Scheme Based on Pixel Difference Pair Mapping

N.SK.BAJI¹, LAKSHMI NARAYANA²

¹Reserch Scholar, Dept of CSP, LITAM, Sattenapalli, Guntur, AP, India, E-mail: bajiece414@gmail.com.

²Asst Prof, Dept of ECE, LITAM, Sattenapalli, Guntur, AP, India, E-mail: mogili_lakshminarayana@yahoo.com.

Abstract: In this paper, based on two-dimensional difference histogram modification, a novel reversible data hiding (RDH) scheme is proposed by using difference-pair-mapping (DPM). First, by considering each pixel-pair and its context, a sequence consisting of pairs of difference values is computed. Then, a two-dimensional difference-histogram is generated by counting the frequency of the resulting difference-pairs. Finally, reversible data embedding is implemented according to a specifically designed DPM. Here, the DPM is an injective mapping defined on difference-pairs. It is a natural extension of expansion embedding and shifting techniques used in current histogram-based RDH methods. By the proposed approach, compared with the conventional one-dimensional difference-histogram and one-dimensional prediction-error-histogram-based RDH methods, the image redundancy can be better exploited and an improved embedding performance is achieved. Moreover, a pixel-pair-selection strategy is also adopted to priorly use the pixel-pairs located in smooth image regions to embed data. This can further enhance the embedding performance. Experimental results demonstrate that the proposed scheme outperforms some state-of-the-art RDH works.

Keywords: Difference-Pair-Mapping (DPM), Histogram Modification, Reversible Data Hiding (RDH), Two-Dimensional Difference-Histogram.

I. INTRODUCTION

Reversible data hiding (RDH) aims to embed secret message into a cover image by slightly modifying its pixel values, and, unlike conventional data hiding, the embedded message as well as the cover image should be completely recovered from the marked content [2]–[4]. RDH is a special type of information hiding and its feasibility is mainly due to the lossless compressibility of natural images. The reversibility in RDH is quite desirable and helpful in some practical applications such as medical image processing [5], [6], multimedia archive management [7], image trans-coding [8], and video error-concealment coding [9], etc. Generally, the performance of a RDH scheme is evaluated by the capacity-distortion behavior. For a required embedding capacity (EC), to obtain a good marked image quality, one expects to reduce the embedding distortion as much as possible. Many RDH methods have been proposed so far, e.g., the methods based on lossless compression [10]–[12], difference expansion, histogram modification, prediction-error expansion, and integer transform, etc. Among them, the histogram-based ones have attracted much attention. The histogram-based methods modify the histogram in such a way that certain bins are shifted to create vacant space while some other bins are utilized to carry data by filling the vacant space. This type of methods can well control the embedding distortion and provide a sufficient EC. The first histogram-based RDH method is the

one proposed. This method uses peak and minimum points of the pixel-intensity-histogram to embed data. It changes each pixel value at most by 1, and thus a good marked image quality can be obtained.

However, its EC is quite low and this method does not work well if the cover image has a flat histogram. To facilitate it, proposed to utilize the difference-histogram instead. This novel method exploits the correlation among neighboring pixels and can embed larger payload with reduced distortion compared with Ni et al.'s. Moreover, we will see later that Lee et al.'s method can be in fact implemented, in an equivalent way, by modifying the two-dimensional pixel-intensity-histogram according to a pixel-pair-mapping (PPM) which is an injective mapping defined on pixel-pair. In this light, the superiority of Lee et al.'s method over Ni et al.'s is explained in another viewpoint. Afterwards, introduced a method by modifying the histogram of prediction-error. Like difference-histogram, the prediction-error-histogram is also Laplacian-like and sharply distributed which guarantees an excellent embedding performance. Instead of only using the correlation of two adjacent pixels in Lee et al.'s method, Fallahpour's method can exploit the local correlation of a larger neighborhood, and thus can provide relatively better performance. Besides the aforementioned methods, many other works are also based on histogram by incorporating

some strategies such as double-layered embedding, embedding-position-selection, adaptive embedding, context-modification, and optimal-bins-selection, etc. We remark that, the histogram-based RDH methods generally contain two basic steps:

- (Histogram generation) First, each local image region consisting of several pixels (e.g., a pixel-pair consisting of two adjacent pixels) is projected to a one-dimensional space (e.g., difference value of a pixel-pair) to get a scalar sequence. Then, a one-dimensional histogram (e.g., difference-histogram) is generated by counting the frequency of the resulting sequence.
- (Histogram modification) finally, embed data into the cover image by modifying the histogram. In most cases, the histogram bins with high frequencies are expanded to carry data while some others are shifted to ensure the reversibility.

In the first step, the complex local image correlation is simplified to a one-dimensional statistic. Clearly, by this simplification, the image redundancy cannot be fully exploited and it only contributes to the second step since a one-dimensional histogram is easy to deal with. Based on this consideration, instead of one-dimensional histogram used in current RDH methods and to better exploit the image redundancy, we propose in this paper a novel RDH scheme by using a two-dimensional difference-histogram. For the proposed method, by considering a pixel-pair and its context, a local image region is projected to a two-dimensional space to obtain a sequence consisting of difference-pairs. Then, a two-dimensional difference-histogram is generated by counting the difference-pairs. Finally, reversible data embedding is implemented according to a specifically designed difference-pair-mapping (DPM). Here, the DPM is an injective mapping defined on difference-pairs, and it is a natural extension of expansion embedding and shifting techniques used in current histogram-based methods. By using the two-dimensional difference-histogram and this specific DPM, compared with the conventional one-dimensional histogram based methods, more pixels are used for carrying data while the number of shifted pixels is reduced as well, and thus an improved embedding performance is achieved.

In addition, inspired by the embedding-position-selection techniques introduced in previous works, a pixel-pair-selection strategy is adopted in our method to priorly use the pixel-pairs located in smooth image regions to embed data. This may further enhance the embedding performance. Experimental results demonstrate that the proposed method outperforms some state-of-the-art works. The rest of the paper is organized as follows. Image Encryption by XOR Algorithm in Section II Section III presents the Histogram Modification and RGB LSB Method for Data Embedding. The Experimental Results are shown in Section IV. And finally Section V concludes this paper.

II. IMAGE ENCRYPTION BY XOR ALGORITHM

First of all, the three-dimensional color image has been taken as the input. Then split the image into its red, green

and blue components. The image redundancy is the property that it utilizes here. The encryption process has to continue for all the three red, green and blue components. The XOR algorithm is utilized for encrypting the component images. In that, block permutation technique is utilized. In this technique, the image can be decomposed into blocks. A group of blocks is taken from the image and these blocks are permuted same as bit and pixel permutation. For better encryption the block size should be lower. If the blocks are very small then the objects and its edges don't appear clearly. At the receiver the original image can be obtained by the inverse permutation of the blocks. The image can be decomposed into blocks; each one contains a specific number of pixels. The blocks are transformed into new locations. For better transformation the block size should be small, because fewer pixels keep their neighbor's.

In this case the correlation will be decreased and thus it becomes difficult to predict the value of any given pixel from the value of its neighbor's. At the receiver side, the original image can be obtained by the inverse transformation of the blocks. The steps of the algorithm are as follows. First load the three dimensional input image. Then split into its red, green and blue components. Then input the 8 bit key. Convert each decimal pixel value into binary. After that repeat step for pixel in all three planes. Rearrange the bits according to the key entered. Then convert the permuted value back to decimal. Transfer a row of pixels into a temporary matrix. Permute the pixels according to the key entered. After that divide the image into 8 blocks, vertically and horizontally finally rearrange the blocks according to the key entered.

III. HISTOGRAM MODIFICATION AND RGBLSB METHOD FOR DATA EMBEDDING

A histogram is a display of statistical information that uses rectangles to show the frequency of data items in successive numerical intervals of equal size. In the most common form of histogram, the independent variable is plotted along the horizontal axis and the dependent variable is plotted along the vertical axis. The data appears as colored or shaded rectangles of variable area. Histogram shifting could be explained as below. Each pixel contained in a digital photograph can have a value between 0 and 255. When the number of pixels having a value of 0, 1, 2, ..., 255, are plotted against the pixel value, it gets the histogram. Histogram modification is used to enhance the image. There are two methods of histogram modification as shown in Fig.1. They are histogram stretching and histogram equalization. If the image is under exposed its values would only occupy the lower part of the dynamic range. The stretching of the histogram was actually performed on the luminance channel after converting the original image to HSL color space. Histogram equalization is a method for spreading the histogram of pixel levels more evenly.

Here it uses the Least Significant Bit (LSB) replacement for data embedding. LSB replacement replaces the least significant bits of pixels with secret data in a cover image.

Implementation of Histogram Modification Method for Reversible Data Hiding Scheme Based On Pixel Difference Pair Mapping

In LSB matching, it first converts the secret data into a stream of bits. Later the LSB of the cover pixel value is added or subtracted if the LSB of the next cover pixel does not match the next bit of secret data. It uses RGB color space for increasing the embedding capacity. An RGB color space is any additive color space based on the RGB color model. A particular RGB color space is defined by the three chromaticities of the red, green, and blue additive primaries and can produce any chromaticity that is the triangle defined by those primary colors. The complete specification of an RGB color space also requires a white point chromaticity and a gamma correction curve. RGB is a convenient color model for computer graphics because the human visual system works in a way that is similar though not quite identical to an RGB color space. Pixels on the left of the graph represent the dark areas in the photograph, while the pixels on the right side of the graph represent the bright areas in the photograph. While allowing shifting the value of all the pixels to the right or left on the histogram graph, vacant spaces created to hide data. The brightness of the image change accordingly. The histogram is modified to create vacant spaces for hiding the data.

The Least Significant Bit embedding technique suggests that data can be hidden in the least significant bits of the cover image and the human eye would be unable to notice the hidden image in the cover file. This technique can be used for hiding images in 24-bit, 8-bit or gray scale format. In LSB technique, first select the message image that is to be hidden behind the cover image. Embed the required number of bits in order to hide the MSB (Most Significant Bit) of the message image behind the LSB (Least Significant Bit) of the cover image. Since the MSB contains

the most important information of the image and the LSB contains the message. This message can be retrieved only by that receiver who knows that it is a stego image sent by the sender. The data embedding technique of the system is RGB-LSB method. Here, the hidden data is embedded into the least significant bits of the red, green and blue components. In the Reversible data hiding scheme, it will first encrypt the original uncompressed red, green and blue components using an encryption key to produce an encrypted image and then follows the data-hiding process.

In data-hiding, it embeds additional data into the encrypted image using a data-hiding key. Having an encrypted image containing additional data receiver firstly decrypts it using the encryption key and can further extract the embedded data. Thus there are mainly three processes, image encryption, data embedding and data extraction/ image recovery. Here after encrypting the image, the vacant spaces for hiding the data been created by histogram shifting and by the RGB- LSB method. In general, in LSB methods, hidden information is stored into a specific position of LSB of image. In the proposed method, it makes use of LSB of red color value, green color value and red color value. Thus it gets more space for data hiding and thus the embedding capacity is improved. The main aim of the work could be attained with the help of this technique. Moreover, here it uses a pixel pair selection strategy in our mapping stage. This is to represent the three dimensional points on a two dimensional plane. By using this selection strategy, only those pixels on the smooth image regions are used.

IV. EXPERIMENTAL RESULTS

Six 512×512 sized gray-scale images including Lena, Baboon, Barbara, Airplane (F-16), Peppers, and Fishing boat are used in our experiment bits per pixel (bpp). According to this figure, compared with Lee et al.'s method and its improvement, one can see that our superiority is significant. It experimentally demonstrates that the DPM-based scheme can provide a much better performance than PPM. Moreover, compared with the methods of Fallahpour and Hong et al., our superiority is also significant. The two methods are based on the prediction-error-histogram with different predictors. Although these methods may exploit the spatial redundancy for a larger pixel context and perform better than Lee et al.'s in most cases, our scheme can improve them by increasing PSNR by 1–6 dB. Our advantage lies in the utilization of two-dimensional difference-histogram and pixel-pair-selection strategy. Sachnev et al.'s method is based on the prediction-error-histogram incorporating with an embedding-position-selection strategy similar to ours.

A sorting technique is used in this method to record prediction-errors based on the magnitude of local variance, and a pixel will be priorly embedded if it has a small local variance. This method performs well and it is superior to some typical RDH schemes such as [7]. Referring to Fig. 2, one can see that in most cases, our scheme is better than this

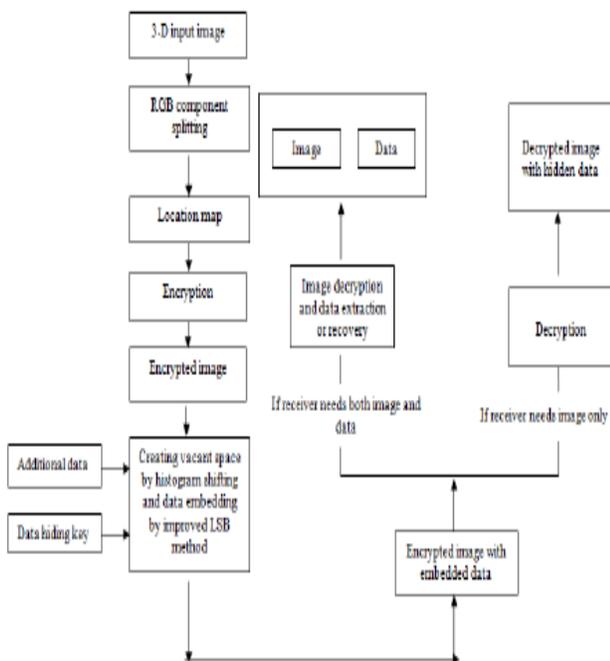


Fig.1. 3-D Difference Histogram Modification for Reversible Data Hiding.

well performed method. But, for some images, when EC approaches its maximum (e.g., for Baboon when EC is larger than 9,000 bits, for Airplane when EC is larger than 45,000 bits), Sachnev et al.'s method can achieve a larger PSNR. The reason is that in such cases, since EC is high, the smooth pixel-pairs are insufficient and our method should necessarily use noisy pixel-pairs to embed data which is unfavorable to the performance. On the other hand, since Sachnev et al.'s method modifies each pixel to embed data (recall that in our method, we modify only one pixel in a pixel-pair), its sorting technique still works well while our pixel-pair-selection strategy is no longer effective when our EC approaches its maximum.

TABLE I: Comparisons of PSNR (In dB) Between Our Method And Four Methods of Sachnev et al, LI et al, Hong, And Hong, For An EC of 10,000 Bits (ER 0.038BPP)

Image	[31]	[24]	[34]	[33]	Proposed scheme
Lena	58.18	58.20	58.78	58.41	59.78
Baboon	54.15	54.03	53.26	54.42	53.96
Barbara	58.15	58.61	58.36	58.21	59.67
Airplane (F-16)	60.38	61.26	62.08	62.35	63.18
Peppers	55.55	56.12	56.07	55.22	57.19
Fishing boat	56.15	55.52	56.64	56.13	57.42
Average	57.09	57.29	57.53	57.46	58.53

TABLE II: Comparisons of Psnr(In dB) Between Our Method And Four Methods of Sachnev et al, LI et al, Hong, And Hong, For An EC of 20,000 Bits (ER 0.076 BPP)

Image	[31]	[24]	[34]	[33]	Proposed scheme
Lena	55.03	54.82	54.92	54.97	56.15
Barbara	55.04	55.29	54.89	55.18	56.24
Airplane (F-16)	57.34	56.84	58.58	58.67	59.45
Peppers	52.30	52.55	52.16	52.20	53.39
Fishing boat	52.65	52.43	52.26	52.37	53.12
Average	54.47	54.39	54.56	54.68	55.67

In this light, Sachnev et al.'s method may perform better than ours in a few cases. However, referring to Tables I and II, our method generally achieves a larger PSNR than Sachnev et al.'s. Our average gains are 1.44 dB for an EC of 10,000 bits, and 1.2 dB for an EC of 20,000 bits. Notice that the result for Baboon is not presented in Table II since our method cannot embed such a payload into this image. Li et al. and Hong's methods are also based on the prediction-error-histogram incorporating with an embedding-position-selection strategy. According to Fig.2, as expected, these methods perform similarly to Sachnev et al.'s and they are better than ours only for some images when our EC approaches its maximum. In most cases, our method can achieve a larger PSNR. Referring to Tables I and II, one can see that our method improves Li et al. and Hong's by increasing PSNR by at least 1 dB in average. We remark that, comparing our method with, our main advantage is the utilization of two-dimensional difference-histogram.

We now discuss the size of the compressed location map. Notice that in our method, the maximum modification to each image pixel is at most 1 in value, so the overflow/underflow problem may only occur for the pixel-pair (x, y) where $x \in \{0, 255\}$ or $y \in \{0, 255\}$. For the six test images Lena, Baboon, Barbara, Airplane (F-16), Peppers, and Fishing boat, except the last two columns and last two rows which are not used in data embedding, the amount of pixel-pairs (x, y) satisfying $x \in \{0, 255\}$ or $y \in \{0, 255\}$ is 0, 0, 0, 0, 2, and 9, respectively. So, in the test cases, there are only a few "1" in the location map (actually, there is no overflow/underflow in most cases). Thus the size of the compressed location map is rather small. Finally, it should be mentioned that some tested methods here can provide a much higher maximum EC (about 1.0 bpp) than the proposed one (e.g., only 0.14 bpp, for the image Lena). The maximum EC in bits (ER in bpp) for our method is

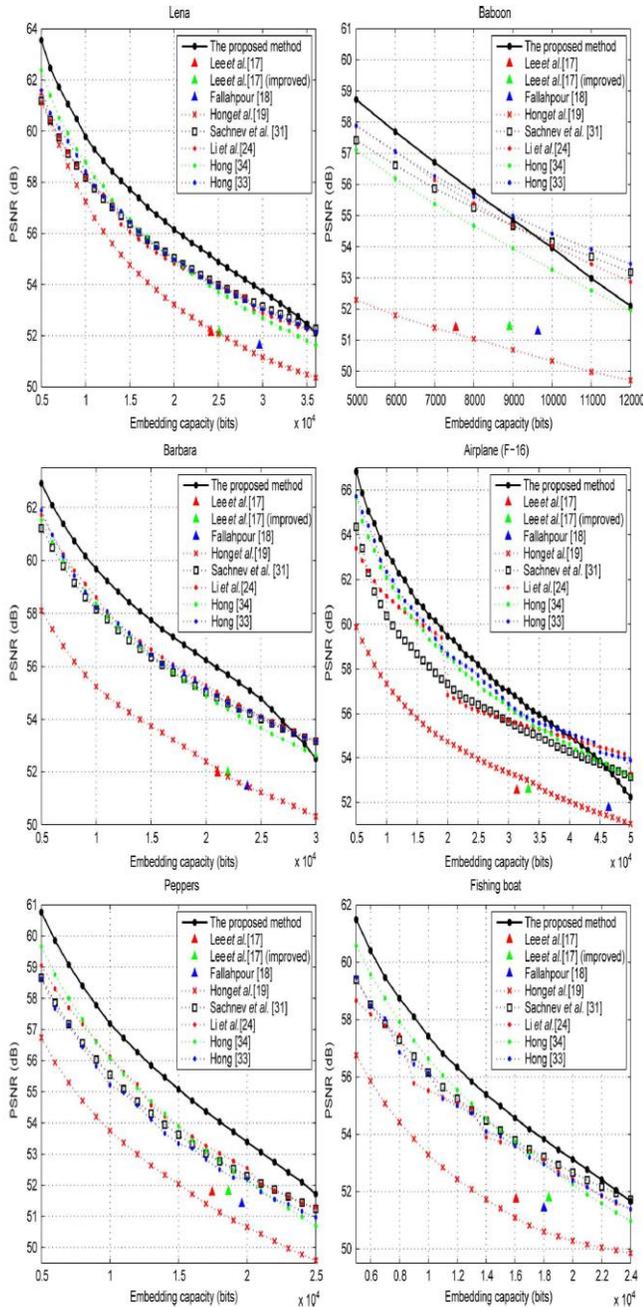


Fig.2. Performance comparisons between our scheme and following seven methods, Fallahpour.

Implementation of Histogram Modification Method for Reversible Data Hiding Scheme Based On Pixel Difference Pair Mapping

36,690 (0.14) for Lena, 12,830 (0.049) for Baboon, 30,790 (0.117) for Barbara, 50,399 (0.192) for Airplane (F-16), 25,309 (0.097) for Peppers, 24,137 (0.092) for Fishing boat, respectively. This is a drawback of our method. However, this level of EC is sufficient for many practical applications, e.g., Coatrieux et al. [5] pointed out that an EC of 3,500 bits (about 0.014 bpp for a 512×512 sized image) is enough for the application of RDH in medical image sharing. Improving EC is beyond the scope of this paper, and we will investigate this issue in our future work.

V. CONCLUSION

In this paper, we presented a novel RDH scheme by using a two-dimensional difference-histogram according to a specifically designed DPM. In addition, a pixel-pair-selection strategy is also proposed to further enhance the embedding performance. This work is the first attempt to employ higher dimensional histogram to design RDH. Compared with the previously introduced one-dimensional histogram based methods, our approach can exploit the image redundancy better and achieve an improved performance. However, since only one pixel of a pixel-pair is allowed to be modified by 1 in value, our EC is low. This issue should be investigated in the future. Moreover, utilizing more suitable two-dimensional histogram and designing more meaningful DPM (e.g., in an image dependent way) to achieve the best embedding performance is also a valuable problem.

VI. REFERENCES

- [1] Xiaolong Li, Weiming Zhang, Xinlu Gui, and Bin Yang, "A Novel Reversible Data Hiding Scheme Based on Two-Dimensional Difference-Histogram Modification", *IEEE Transactions on Information Forensics and Security*, Vol. 8, No. 7, July 2013.
- [2] Y. Q. Shi, "Reversible data hiding," in *Proc. IWDW*, 2004, vol. 3304, pp. 1–12, ser. Springer LNCS.
- [3] Y. Q. Shi, Z. Ni, D. Zou, C. Liang, and G. Xuan, "Lossless data hiding: fundamentals, algorithms and applications," in *Proc. IEEE ISCAS*, 2004, vol. 2, pp. 33–36.
- [4] R. Caldelli, F. Filippini, and R. Becarelli, "Reversible watermarking techniques: An overview and a classification," *EURASIP J. Inf. Security*, vol. 2010, 2010, Article 134546.
- [5] G. Coatrieux, C. L. Guillou, J. M. Cauvin, and C. Roux, "Reversible watermarking for knowledge digest embedding and reliability control in medical images," *IEEE Trans. Inf. Technol. Biomed.*, vol. 13, no. 2, pp. 158–165, Mar. 2009.
- [6] M. Fontani, A. D. Rosa, R. Caldelli, F. Filippini, A. Piva, and M. Consalvo, "Reversible watermarking for image integrity verification in hierarchical pacs," in *Proc. 12th ACM Workshop on Multimedia and Security*, 2010, pp. 161–168.
- [7] S. Lee, C. D. Yoo, and T. Kalker, "Reversible image watermarking based on integer-to-integer wavelet transform," *IEEE Trans. Inf. Forensics Security*, vol. 2, no. 3, pp. 321–330, Sep. 2007.

- [8] R. Li, O. C. Au, C. K. M. Yuk, S. Yip, and T. Chan, "Enhanced image trans-coding using reversible data hiding," in *Proc. IEEE ISCAS*, 2007, pp. 1273–1276.
- [9] K.-L. Chung, Y.-H. Huang, P.-C. Chang, and H.-Y. Liao, "Reversible data hiding-based approach for intra-frame error concealment in H.264/AVC," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 11, pp. 1643–1647, Nov. 2010.
- [10] J. Fridrich, M. Goljan, and R. Du, "Lossless data embedding—new paradigm in digital watermarking," *EURASIP J. Appl. Signal Process.*, vol. 2002, no. 2, pp. 185–196, Feb. 2002.
- [11] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Lossless generalized-LSB data embedding," *IEEE Trans. Image Process.*, vol. 14, no. 2, pp. 253–266, Feb. 2005.
- [12] M. U. Celik, G. Sharma, and A. M. Tekalp, "Lossless watermarking for image authentication: A new framework and an implementation," *IEEE Trans. Image Process.*, vol. 15, no. 4, pp. 1042–1049, Apr. 2006.