New Difference Expansion Method for Lossless Data Hiding

Hyoung Joong Kim, Vasiliy Sachnev, Dong Hoi Kim

I. Introduction

REVERSIBLE data hiding [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [20], which is also called the lossless data hiding, invisibly hides data (which is called a payload) into host data (i.e., pixels in image) in a reversible fashion. Being reversible, both the original data and the embedded data can be completely restored. Two important measures of reversible data hiding are embedding capacity and quality degradation. These two measures conflict with each other. The objective of data hiding is to achieve high capacity and low distortion.

Difference expansion transform, invented by Tian [13], is an outstanding reversible data hiding scheme in terms of high embedding capacity and low distortion in image quality. His method divides the image into pairs of pixels, then embeds one bit of information into the difference of the pixels of each pair from those pairs that are not expected to cause an overflow or underflow. A pair generally consists of two neighboring pixels or two with a small difference value. The location map that indicates the modified pairs is compressed and included in the payload.

The seminal paper by Tian [13] has been a steppingstone to enhanced performance. Alattar [1] has extended the difference expansion transform from a pair of pixels to a triplet, a set of three pixels, to hide two bits in every triplet of pixels. Alattar [2] has derived an enhanced difference expansion transform that is based on a quad, a set of four pixels, to hide two bits in every quad. There are spatial triplets, cross-color triplets, spatial quads, and cross-color quads according to the combination of pixels. Alattar [3] has shown that spatial quads can hide the largest payload at the highest signal-to-noise ratio.

Though Tian's difference expansion transform [13] is a brilliant breakthrough in reversible data embedding, it has a serious weak point: the location map and the correction bits are embedded into the image together with the payload. The location map tells the decoder which pair has been expanded and which pair has not. The correction bits are necessary to recover the exact bits where location map bits are overwritten (see the exact definition of the correction bits in [13]. Needless to say, this location map and correction bits reduce the embedding capacity of the difference expansion transform. The embedding capacity of the difference expansion transform is at best 0.5 bit-per-pixel without embedding the location map. Unfortunately, the location map itself needs 0.5 bit-per-pixel. Of course, excellent compression algorithms like JBIG can compress the location map so that the embedding capacity is maximized while the required bits for the location map is minimized. Thus, a difference expansion transform free from the location map would be highly desirable. However, it is not easy to eliminate the location map. The question is whether the size of a location map can be reduced or the location map simplified so that compression is not necessary. This paper provides an answer to this question.

In this paper, two novel techniques are proposed to improve the Tian's method further. This paper introduces a new location map and new embedding method of the location map. This paper will show that the new location map is smallest in size so far. The method in this paper embeds location and payload sequentially, but no correction bits which should be embedded in Tian's method. Thus, this paper will show that the method proposed in this paper outperforms the existing schemes in terms of embedding capacity and the image quality.

This paper is organized as follows. In Section 2, the difference expansion transform is reviewed. The T-expandable pair is defined in Section 3. Simple encoding and decoding rules are presented. Section 4 shows the effectiveness of the new encoding and decoding rules. Performance comparison with Tian's method [13] shows that the proposed scheme in this paper is better. Section 5 concludes the paper.

II. Difference Expansion Transform

Assume that we have two 8-bit gray-scale value pair (x, y), where $x, y \in \mathrm{Z}$, and $0 \le x, y \le 255$. We can define integer average value l and difference value h from the pair as follows:

$$l = \left\lfloor \frac{x+y}{2} \right\rfloor, \quad h = x - y, \tag{1}$$

where the inverse transform of (1) is given as follows:

$$x = l + \left\lfloor \frac{h+1}{2} \right\rfloor, \quad y = l - \left\lfloor \frac{h}{2} \right\rfloor. \tag{2}$$

The reversible integer transforms in (1) and (2) are also called the integer Haar wavelet transform or S transform. The reversible integer transforms set up a one-to-one mapping between (x, y) and (l, h). The new difference h' is computed by expanding the difference by a factor of 2 and embedding b as follows:

Hyoung Joong Kim is a Professor at CIST, Graduate School of Information Management and Security, Korea University.

Vasiliy Sachnev is a Ph.D. student of Department of Electronics and Telecommunications Engineering, Kangwon National University.

Dong Hoi Kim is a Assistant Professor of epartment of Electronics and Telecommunications Engineering, Kangwon National University.

$$h' = 2h + b. \tag{3}$$

Note that $2h$ in (3) produces even number regardless of whether h is odd or even. Thus, we have room to hide one bit of binary information b after expanding the difference by a factor 2. However, due to the expansion this transform can cause underflow or overflow errors. That is to say, it implies that not all pairs are expandable. The location map contains the location information of all selected expandable pairs. For the expandable pairs, data embedding procedures are summarized as follows:

$$x' = l + \left\lfloor \frac{h'+1}{2} \right\rfloor, \quad y' = l - \left\lfloor \frac{h'}{2} \right\rfloor. \tag{4}$$

From (4), to prevent the overflow and underflow problems, i.e., to restrict x', y' in the range of [0, 255], it is equivalent to have

$$0 \le l + \left\lfloor \frac{h'+1}{2} \right\rfloor \le 255, \quad 0 \le l - \left\lfloor \frac{h'}{2} \right\rfloor \le 255. \tag{5}$$

for both $b = 0$ and $b = 1$. If the conditions in (5) are satisfied, the h associated with h' is said to be *expandable* under the integer average value l. Expandable difference value h is a candidate for the difference expansion. From the encoder's perspective, the expandable difference value is important since it tells us that it can be expanded to embed one bit of information. However, not every h is expandable (see Figure ??). The encoder expands only expandable difference values. Of course, some expandable difference values are not necessarily expanded for some reasons. For example, excessively large difference values can cause severe degradation in image quality. In that context, decoder must know whether a pair has been expanded or not. Only the location map can provide the decoder information about the expanded pairs. Tian's method [13] can decode payload only when the location map is extracted and decode first and the correction bits are recovered second. However, the location map and the correction bits make an inroad into potential payload.

A difference value h is *changeable* under l if

$$0 \le l + 2\left\lfloor \frac{h}{2} \right\rfloor + b \le 255, \quad 0 \le l - 2\left\lfloor \frac{h}{2} \right\rfloor + b \le 255. \tag{6}$$

for both $b = 0$ and $b = 1$. Note that the expandable difference value is always changeable, but the converse is not always true. For example, assume we have two values $x = 123$ and $y = 33$. Then, l is 78, and h is 90. This difference value h is not expandable, but is changeable. In this example, it is clear that the changeability also cannot guarantee whether the pair has been expanded or not. Thus, the location map is indispensable.

III. New Difference Expansion Transform

A. New Expandability and Changeability

This section describes the difference expansion transform which can be decoded with the simplified location map. If the expandable difference value h satisfies

$$|h| \le T, \tag{7}$$

with the integer average value l, the h is said to be *T-expandable*. The encoder in this paper expands the h only when it is T-expandable. Data embedding capacity and the image quality after embedding depend on the threshold value T. If the changeable difference value h satisfies condition (7) with the integer average value l, the h is said to be *T-changeable*.

B. Simplified Location Map

The difference expansion method by Tian [13] makes a location map which covers all pairs. Thus, the size of the location map is the half of the image size. In other words, if the location map is not compressed, there will be no room for payload. Good compression algorithm can compress the location map so that the size of the location map is hopefully sufficiently small.

However, note that the size can be reduced considerably and further when we exploit the threshold value T. For the sake of convenience, assume that there are only four types of pairs: $h = 0$, $h = 1$, $h = 2$ and $h = 3$. Let T be 1. Then, only $h = 0$ and $h = 1$ can be T-expandable. After expansion, we have four possible values to decode: $h' = 0$, $h' = 1$, $h' = 2$ and $h' = 3$. In this example, we embed the simplified location map into the pairs with $h = 0$ or equivalently $h' = 0$ and $h' = 1$. Let M denote the set of pairs of which the difference values are T-expandable and into which the simplified location map is to be embedded. In this example, the set M is denoted by $M = \{(h) | h' \in \{0, 1)\}\}$. (Note that M is not necessarily dedicated to embedding the location map only because part of the payload can be embedded, which is to be explained later.) Then, the payload can be embedded into the pairs of with $h = 1$ or equivalently $h' = 2$ and $h' = 3$. Let N denote the set of pairs of which the difference values are T-changeable and into which the payload is embedded. In this example, the set N is denoted by $N = \{h | h' \in \{2, 3)\}\}$. (Note that N is not necessarily dedicated to embedding the payload only, which is to be explained later.)

A problem arises at the decoder: it is not clear whether the $h' = 2$, for example, has come from $h = 1$ which has been expanded, or from $h = 2$ which has not been T-expandable. Thus, the location map needs to indicate whether a pair has been expanded or not. Note that N consists of two disjoint subsets N_e and $N_{\bar{e}}$. Let N_e denote the set of pairs of which elements have been expanded from the T-expandable difference values. Similarly, let $N_{\bar{e}}$ denote the set of pairs of which elements have not been expanded since they were not T-expandable. In this example, N_e is represented by $N_e = \{h | h \in \{1\} \cup h' \in \{2, 3\}\}$, and $N_{\bar{e}}$ by $N_{\bar{e}} = \{h | h \in \{2, 3\} \cup h' \in \{2, 3\}\}$.

The decoder chooses a set of pairs that is T-changeable. Among them, there are two subsets: M and P. One subset, N, is possibly for the payload (in this example, $h' = 2$ and $h' = 3$). The other set, M, is for the simplified location map (i.e., $h' = 0$ and $h' = 1$). Since the T-changeable elements are easily identified by (6)-(7), and since the pairs for the potential payload, i.e., N, and for the simplified location map, i.e., M, are clearly distinguished from each other, the location map can be simplified. The simplified location map just covers only the pairs in N and small number of pairs in M which has not been expanded. Thus, the location map does not cover the whole pairs. Therefore, we call it the simplified location map.

For convenience' sake, we state that N is for the payload and M for the simplified location map. However, both M and N are used to hide any kind of information. For example, the simplified location map can be embedded into N and the payload into M, too if necessary. Embedding of the location map and part of the payload into M is also possible if the location map size is smaller than $|M|$. Thus, note that the actual payload size is larger than or equal to $(|M| + |N_e|) - (|N|)$ or $(|M| - |N_{\bar{e}}|)$, where $|X|$ stands for the size of X.

Example 1: Let the frequencies of $h = 0$, $h = 1$, $h = 2$ and $h = 3$ be 200, 100, 50, and 40, respectively. Assume that all pairs of $h = 0$ and $h = 1$ are strictly T-changeable under the another assumption of $T = 1$. Then, only the pairs with $h = 0$ and $h = 1$ are expanded. Thus, after the expansion, the sum of the frequencies of either $h' = 0$ and $h' = 1$ is 200 (i.e., the frequency of $h = 0$), while that of $h' = 2$ and $h' = 3$ is 190 (frequency sum of $h = 1$, $h = 2$ and $h = 3$). Then, it is clear that $|M| = 200$ and $|N| = 190$ while $|N_e| = 100$ and $|N_{\bar{e}}| = 90$.

The encoder has two choices. First choice is to keep the actual size of payload equal to 110 (i.e., $|M| - |N_{\bar{e}}| = 110$) if the location map is not compressed. Second choice is to compress the location map. Then, the actual payload is far more than 10. Thus, needless to say, good compression algorithm can maximize the actual size of payload (i.e., actual embedding capacity).

IV. Experiments

To assist the better understanding of our algorithm, Example 2 is provided.

Example 2: Consider an example with 8 pairs as is shown in Table I. The payload bitstream is "10" or 1 and 0 consecutively in binary numbers. Thus, the payload size is two bits. Let the threshold value T be 1. There are three pairs whose difference values are all 1: namely, pair 3, pair 5, and pair 7. However, the last pair is not T-expandable since its l value violates (4). Thus, those two pairs, (102, 101) and (102, 101), are sufficient to embed two bits of data.

Binary number 1 is embedded into the pair 3, and the resulting expanded values are $x' = 103$ and $y' = 100$. Similarly, binary number 0 is embedded into the pair 5, and the resulting expanded values are $x' = 102$ and $y' = 100$. It is clear that there are three pairs that belong to N: namely, pairs 3, 4, and 5. Among them, the pairs 3 and 5 belong to the set N_e, and the pair 4 belongs to the set $N_{\bar{e}}$. Thus, the simplified location map is represented by three bits of binary data as 101, where 1 denotes the membership of N_e and 0 for $N_{\bar{e}}$. It is obvious that we need three pairs to embed the simplified location map. Fortunately, there are three pairs that belong to the set M: namely, pairs 1, 6, and 8. The bit to embed into the pair 1 is accordingly 1, that into the pair 6 is 0, and that into the pair 8 is 1. Pairs 2, 4, and 7 are neither T-expandable nor T-changeable. Thus, their b values are marked as "X" in Table I.

TABLE I
DATA FOR 8 EXEMPLAR PAIRS

Pairs	1	2	3	4	5	6	7	8
x	100	104	102	103	102	108	255	201
y	100	100	101	100	101	108	254	201
h	0	4	1	3	1	0	1	0
l	100	102	101	101	101	108	254	201
b	1	X	1	X	0	0	X	1
x'	101	104	103	103	102	108	255	202
y'	100	100	100	100	100	108	254	201
h'	1	4	3	3	2	0	1	1
l	100	102	101	101	101	108	254	201

TABLE II
HISTOGRAM OF DIFFERENCE VALUES OF LENA IMAGE

Difference values	0	1	-1	2	-2
Frequencies	14,450	12,102	12,034	10,409	10,168
Difference values	3	-3	4	-4	5
Frequencies	8,266	8,110	6,220	6,130	4,596
Difference values	-5	6	-6	7	...
Frequencies	4,568	3,380	3,366	2,509	...

Part of the frequency of the strictly T-expandable pairs from a 512×512 Lena image is given in Table II. The number of pairs with $h = 0$ is 14,450, that with $h = 1$ is 12,102, that with $h = 2$ is 10,409, that with $h = 3$ is 8,266, and so on (see Table II). When T is 1, the capacity for the simplified location map is 14,450 bits. However, the size of N is 30,777 bits (which is equal to $|N|$ and is the sum of 12,102, 10,409, and 8,266). The embedding capacity is $|M|+|N_e|$, while the size of simplified location map is N (where $N = |N_e| + |N_{\bar{e}}|$). Thus, actual payload size is $(|M| + |N_e|) - (|N|)$ or $(|M| - |N_{\bar{e}}|)$, which is totally predictable once the histogram is available. In case when $(|M| - |N_{\bar{e}}|) > 0$, the map does not need to be compressed. Otherwise, good compression algorithm is necessary to reduce the size of location map considerably.

Table III shows that when we use $h = 0, \pm 1, \pm 2, \pm 3$ to embed data, the actual payload size is 9,847 bits even though the simplified location map is not compressed. In this case, image quality is around 48.04 dB. If we use larger h values, the payload size also increases. It is obvious because the frequency of larger difference values gets smaller. Figure 1 shows that the proposed algorithm keeps very high image quality. The right-hand side image is obtained with $h = 0, \pm 1, \pm 2, \pm 3$ into 512

TABLE III
EMBEDDING RESULTS BASED ON THE PROPOSED ALGORITHM OVER LENA

Difference values	$\lvert M + N_e \rvert$	$\lvert N_e + N_{\tilde{e}} \rvert$	Payload	PSNR
$h = -3, \cdots, 3$	45,227	35,380	9,847 bits	48 dB
$h = -5, \cdots, 5$	56,046	29,982	26,061 bits	45 dB
$h = -7, \cdots, 7$	61,932	23,918	38,014 bits	44 dB
$h = -9, \cdots, 9$	65,071	18,964	46,107 bits	43 dB

Fig. 1. Original (left) and data embedded (right) Lena images

× 521 Lena image.

V. CONCLUSIONS

Among many reversible data embedding algorithms Tian's method [13] has been reviewed and enhanced in this paper. Being reversible, both the original data and the embedded data should be completely restored. Tian's difference expansion transform has been a remarkable breakthrough in reversible data hiding scheme. The difference expansion method achieves high embedding capacity and keeps distortion low. This paper shows that the difference expansion method with the simplified location map, and new expandability and changeability can achieve more embedding capacity while keeping distortion almost the same as the original expansion method. Examples shown illustrate how the proposed method works. Advantages over Tian's method [13] are shown using the simple example with Lena and Mandrill images. Performance comparison results of single embedding over Lena and Mandrill images show that the simplified location map is very much effective. Our scheme hides more data in case of multiple embedding. At the cost of image quality, more than 2 bpp has been achieved into Lena image by multiple embedding compared with Tian's method. In case of the Mandrill image, the maximum achievable embedding capacity is around 1 bpp.

ACKNOWLEDGMENT

This research was supported by Korean Ministry of Information and Communication under the project funded by Information Technology Research Center (ITRC).

REFERENCES

[1] A. M. Alattar, "Reversible watermark using difference expansion of triplets," *Proceedings of the International Conference on Image Processing*, vol. 1, pp. 501-504, 2003.

[2] A. M. Alattar, "Reversible watermark using difference expansion of quads, "*Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 3, pp. 377-380, 2004.

[3] 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.

[4] F. Bao, R. H. Deng, B. C. Ooi, and Y. Yang, "Tailored reversible watermarking schemes for authentication of electronic clinical atlas," *IEEE Transactions on Information Technology in Biomedicine*, vol. 9, no. 4, pp. 554 - 563, 2005.

[5] J. M. Barton, "Method and apparatus for embedding authentication information within digital data," U.S. Patent 5,646,997, 1997.

[6] M. Celik, G. Sharma, A.M. Tekalp, and E. Saber, "Reversible data hiding," *Proceedings of the International Conference on Image Processing*, Rochester, NY, pp. 157-160, 2002.

[7] J. Fridrich, M. Goljan, and R. Du, "Invertible authentication," *Proc. SPIE, Security and Watermarking of Multimedia Contents*, San Jose, CA, pp. 197-208, 2001.

[8] M. Goljan, J. Fridrich, and R. Du, "Distortion-free data embedding," *Proceedings of the Information Hiding Workshop*, Pittsburgh, PA, pp. 27-41, 2001.

[9] C. W. Honsinger, P. Jones, M. Rabbani, and J. C. Stoffel, "Lossless recovery of an original image containing embedded data," US Patent 6,278,791 B1, 2001.

[10] B. Macq and F. Deweyand, "Trusted headers for medical images," *DFG VIII-D II Watermarking Workshop*, Erlangen, Germany, 1999.

[11] Z. Ni, Y. Q. Shi, N. Ansari, W. Su, Q. Sun, and X. Lin, "Robust lossless image data hiding," *IEEE International Conference on Multimedia and Expo*, Taipei, Taiwn, pp. 2199-2202, 2004.

[12] D. M. Thodi, and J. J., Rodriguez, "Reversible watermarking by prediction-error expansion," *IEEE Southwest Symposium on Image Analysis and Interpretation*, Lake Tahoe, CA, pp. 21-25, 2004.

[13] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 13, no. 8, pp. 890-896, 2003.

[14] C. de Vleeschouwer, J. F. Delaigle, and B. Macq, "Circular interpretation of bijective transformations in lossless watermarking for media asset management," *IEEE Transactions on Multimedia*, vol. 5, no. 1, pp. 97-105, 2003.

[15] G. Xuan, J. Zhu, J. Chen, Y. Q. Shi, Z. Ni and W. Su, "Distortionless data hiding based on integer wavelet transform," *IEE Electronics Letters*, vol. 38, no. 25, pp. 1646-1648, 2002.

[16] G. Xuan, Y. Q. Shi, Z. C. Ni, J. Chen, C. Yang, Y. Zhen, and J. Zheng, "High capacity lossless data hiding based on integer wavelet transform," *Proceedings of IEEE International Conferenceon Circuits and Systems*, Vancouver, Canada, 2004.

[17] G. Xuan, and Y. Q. Shi, "Integer wavelet transform based lossless data hiding using spread spectrum," *IEEE International Workshop on Multimedia Signal Processing*, Siena, Italy, 2004.

[18] B. Yang, M. Schmucker, W. Funk, C. Busch, and S. Sun, "Integer DCT-based reversible watermarking for images using companding technique," *Proceedings of SPIE*, vol. 5306, pp. 405-415, 2004.

[19] B. Yang, M. Schmucker, C. Busch, X. Niu, and S. Sun, "Approaching optimal value expansion for reversible watermarking," *Proceedings of the 7th workshop on Multimedia and Security*, pp. 95-102, 2005.

[20] D. Zou, Y. Q. Shi, and Z. Ni, "A semi-fragile lossless data hiding scheme based on integer wavelet transform," *IEEE International Workshop on Multimedia Signal Processing*, Siena, Italy, 2004.