DIFFERENCE-EXPANSION BASED REVERSIBLE AND VISIBLE IMAGE WATERMARKING SCHEME

Chien-Chang Chen (*陳建彰*), *Hsin-Cheng Yeh* (*葉信呈*)

Department of Computer Science and Information Engineering,
Tamkang University, Taipei, Taiwan
E-mail: ccchen34@mail.tku.edu.tw

ABSTRACT

A reversible image watermarking scheme recovers the original cover image after extracting the embedded watermarks. A visible image watermarking scheme embeds watermarks to create a visible effect on the cover image. General reversible image watermarking scheme embeds invisible watermarks. This paper presents a reversible and visible image watermarking scheme with the usage of conventional difference expansion calculation. The cover image is first segmented to non-overlapped $k \times k$ blocks. Each block is then applied to two watermarking schemes including difference-expansion based invisible watermarking scheme and adjustable visible watermarking scheme to embed one watermark bit. The difference expansion based invisible watermarking scheme is adopted for embedding and extracting one watermark bit. Exceeding numbers, larger than 255 or smaller than 0, generated from the difference expansion method should be recorded for lossless recovery. The adjustable visible watermarking scheme embeds significant visible watermarks. Experimental results show that the proposed scheme embeds watermarks with adjustable visual effect and the watermarks can be extracted perfectly. Any attack debases the quality of recovered cover image.

Keywords *reversible watermark, visible watermark, adjustable visibility, difference expansion*

1. INTRODUCTION

Digital images are popularly used for the rapid growth in computer network. The ownership of an image is therefore an important issue. Watermarking techniques become more and more important in protecting vital images. The techniques adjust a host image by embedding user information as watermarks. The watermarks can be classified to reversible or irreversible which means that the host image can be perfectly recovered or not. The watermarks can also be classified to visible or invisible meaning that the embedded watermarks can be noticed or not. In real applications, visible watermarks are popular for their apparent watermarks. Moreover, the reversible property is another important issue when we need to recover the original host image. A watermarking scheme both containing visible and reversible properties are thus required.

Research on reversible watermark approaches are mostly focused in invisible embedding strategy. Ni et al. embedded watermarks by shifting peak-pixel value in histogram [16]. Tian embedded one bit into two pixels by enlarging the pixels difference and inserting the bit to the LSB [18]. Alattar calculated the difference expansion of the integer transformation to embed watermarks [3]. Further studies on difference expansion include 2-D difference expansion [2], prediction based difference expansion [4, 10], two embedding directions [6], location map improvements [7, 9, 12], and partial difference expansion [17]. Other applications like applying to medical images[1], combining with genetic algorithm [8], combining with coding technique [14], combining with interpolation and histogram shifting [15], and edge detection [20] are also studied.

Some visible reversible image watermarking schemes are also presented. Hu and Jeon [5] selected two dependent data sets from the host image for embedding watermark and non-watermark information. Lin et al. [11] embedded reversible and visible watermarks into four similar images that are partitioned from the cover image. Liu and Tsai [13] used a one-to-one compounding mapping function to adjust pixel values for embedding visible watermarks. Watermarks are needed to recover the original host image. Yang et al. [21] embedded the watermarks into user selected region and embedded the reversible information into invisibly watermarked region.

This paper presents a difference-expansion based visible image watermarking scheme with reversibly recovering the original host image. The host image is segmented to non-overlapped blocks. The block can be 2×2, 3×3, to $k \times k$. One watermark bit is embedded in each block with visible and invisible watermarking scheme. Two kinds of watermarking schemes are adopted in each block. One pixel in each block is selected to embed using reversible watermarking scheme and others embed using visible watermarking scheme. Since the difference-expansion method is adopted to embed reversible watermark, some information may require to be recorded for its extending the image pixel value after applying the difference-expansion method. Therefore, some information should be recorded if necessary.

The rest of this paper is organized as follows. Section 2 reviews the difference-expansion reversible embedding method. Section 3 introduces the proposed difference-expansion based reversible and visible image watermarking method. Section 4 presents the experimental results of the proposed method. Comparisons with other related works are also pro-vided. Finally, Section 5 draws a brief conclusion.

2. REVIEWS OF THE DIFFERENCE-EXPANSION METHOD

This section reviews the difference-expansion reversible embedding method that embeds one bit into a pair of numbers and recover these two numbers after extracting the watermark bit. Let (x, y) denote two numbers and w denote the watermark bit. The embedding procedure is introduced as follows. First, calculate the mean m and difference d between these two numbers by Eq.(1).

$$\begin{aligned} d &= x - y \\ m &= \left\lfloor \frac{x+y}{2} \right\rfloor \end{aligned} \tag{1}$$

Embed the watermark bit w into the difference d by Eq.(2).

$$d' = 2d + w \tag{2}$$

Recover the two embedded numbers (x', y') from m and d' by Eq.(3).

$$\begin{aligned} x' &= m + \left\lfloor \frac{d'+1}{2} \right\rfloor \\ y' &= m - \left\lfloor \frac{d'}{2} \right\rfloor \end{aligned} \tag{3}$$

Therefore, we can embed one bit w into a pair of numbers (x, y) to acquire (x', y') using Eq. (1) through Eq. (3).

The recovery procedure recovers (x, y) and the watermark bit w from (x', y'). First, calculate the mean m and difference d' between two numbers (x', y') using Eq. (4)

$$\begin{aligned} d' &= x' - y' \\ m &= \left\lfloor \frac{x'+y'}{2} \right\rfloor \end{aligned} \tag{4}$$

Extract the watermark bit w and recover the original difference d by Eq. (5)

$$\begin{aligned} w &= d' \% 2 \\ d &= \left\lfloor \frac{d'}{2} \right\rfloor \end{aligned} \tag{5}$$

Recover the original two numbers (x, y) from above calculated m and d by Eq.(6).

$$\begin{aligned} x &= m + \left\lfloor \frac{d+1}{2} \right\rfloor \\ y &= m - \left\lfloor \frac{d}{2} \right\rfloor \end{aligned} \tag{6}$$

We can embed one bit to two numbers using Eqs. (1)~(3) and recover these two numbers with extracting the bit using Eqs. (4)~(6) [3, 18]. Some works [2, 4, 6, 7, 9, 10, 12] further improve the performance of difference expansion method.

3. PROPOSED SCHEME

This section introduces the presented difference-expansion based visible image watermarking scheme. Section 3.1 introduces the watermark embedding algorithm. Section 3.2 introduces the watermark extracting and host image recovery algorithm.

3.1 Watermark Embedding Algorithm

This section introduces the watermark embedding algorithm of the proposed scheme. The proposed scheme partitions the cover image into non-overlapped $k \times k$ blocks. The parameter k is defined by the size within the constraint of $2 \le k \le \min\left\{\left\lfloor \frac{M_c}{M_w} \right\rfloor, \left\lfloor \frac{N_c}{N_w} \right\rfloor\right\}$, where sizes of the cover image and the watermark image are assumed to be $M_c \times N_c$ and $M_w \times N_w$, respectively. The proposed scheme embeds one watermark bit into a block. In each block, these $k \times k$ pixels are partitioned to two sets S_1, S_2 and two pixels y_1, y_2 by the *partition strategy*, in which the set S_1 includes $\left\lfloor \frac{k^2-1}{2} \right\rfloor$ pixels and S_2 has $\left\lfloor \frac{k^2-1}{2} \right\rfloor - 1$ pixels. Two different watermarking schemes, including visible and reversible, are presented to embed one watermark bit w into a block. The proposed visible watermarking strategy embeds one watermark bit into block pixels using parameters *Base*, *ShiftSeed*, and *ShiftQuantity* to create visual effect. The reversible watermarking strategy only embeds into the pixels y_1 and y_2. The embedding algorithm is introduced as follows.

1. Partition the cover image to non-overlapped $k \times k$ blocks with $2 \le k \le \min\left\{\left\lfloor \frac{M_c}{M_w} \right\rfloor, \left\lfloor \frac{N_c}{N_w} \right\rfloor\right\}$, where sizes of the cover image and the embedded watermark image are assumed to be $M_c \times N_c$ and $M_w \times N_w$, respectively.

2. Calculate the local mean value l_m among the watermark embedded region, where size of the region is defined by $k \times M_w \times N_w$.
3. Obtain the threshold L_m using l_m and Eq. (7)

$$L_m = \begin{cases} l_m + 30\,, & if\ l_m < 128 \\ l_m - 30\,, & if\ l_m \geq 128 \end{cases} \tag{7}$$

4. For the $M_w \times N_w$ watermark image and the $M_c \times N_c$ cover image, apply the following steps to embed one watermark bit w into corresponding $k \times k$ block.

4.1 Calculate the block mean value b_m among the $k \times k$ block.

4.2 Use pre-defined parameters *Base* and *Shift*(i,j) to embed w into all pixels in the block by Eq. (8)

$$x'(i,j) = \begin{cases} x(i,j) + w \cdot (Base + Shift(i,j))\,, & if\ b_m < L_m \\ x(i,j) - w \cdot (Base + Shift(i,j))\,, & if\ b_m \geq L_m \end{cases} \tag{8}$$

where *Base* is a shift constant and *Shift* is a set of random numbers with each value generated from the seed number *ShiftSeed* within the range of $0 \leq Shift(i,j) < ShiftQuantity$.

4.3 Use *partition strategy* to partition pixels in the block to two sets S_1, S_2, and two pixels y_1, y_2, where S_1 includes $\left\lfloor \frac{k^2-1}{2} \right\rfloor$ pixels and S_2 includes $\left\lfloor \frac{k^2-1}{2} \right\rfloor - 1$ pixels.

4.4 Calculate block differences d_1, d_2 between S_1, S_2, and y_1, y_2, by Eq. (9)

$$\begin{aligned} d_1 &= \sum_{x'_1(i,j) \in S_1} x'_1(i,j) - \sum_{x'_2(i,j) \in S_2} x'_2(i,j) - y_1 \\ d_2 &= \sum_{x'_1(i,j) \in S_1} x'_1(i,j) - \sum_{x'_2(i,j) \in S_2} x'_2(i,j) - y_2 \end{aligned} \tag{9}$$

4.5 Use the difference-expansion method to embed the reversible watermark bit w into the block difference d_1 by Eq. (10) and replace the corresponding pixel y_1

$$y'_1 = 2 * d_1 + w + \left\lfloor \frac{\sum_{x'(i,j) \in \{S_1 \cup S_2\}} x'(i,j)}{2 \times \left\lfloor \frac{k^2-1}{2} \right\rfloor - 1} \right\rfloor \tag{10}$$

4.6 When the watermark bit w is 1, use the difference-expansion method to embed the sign bit *sign* into the block difference d_2 by Eq. (11) and replace the corresponding pixel y_2

$$y'_2 = \begin{cases} 2 * d_2 + 0 + \left\lfloor \frac{\sum_{x'(i,j) \in \{S_1 \cup S_2\}} x'(i,j)}{2 \times \left\lfloor \frac{k^2-1}{2} \right\rfloor - 1} \right\rfloor, & if\ b_m < L_m \\ 2 * d_2 + 1 + \left\lfloor \frac{\sum_{x'(i,j) \in \{S_1 \cup S_2\}} x'(i,j)}{2 \times \left\lfloor \frac{k^2-1}{2} \right\rfloor - 1} \right\rfloor, & if\ b_m \geq L_m \end{cases} \tag{11}$$

where the sign bit *sign* denotes the added 0 or 1 in Eq.(11).

Since the proposed scheme partitions a block pixels into one pixel y and other pixels in which include two sets S_1, S_2. Assume that sizes of host image and watermark image be 512×512 and 128×128, respectively. Thus, k can be 2, 3, or 4. The parameter k determines the scale of the visible watermark. Larger k value acquires larger watermark. Table 1 shows partition examples of k = 2 and 3. Since number of S_2 is one less than number of S_1. Therefore, the block difference d, acquired from Eq. (8), can be taken as difference of the block because the positive number equals to the negative number. Consequently, in most of image blocks, the block difference can be very small and the calculated difference *y'* in Step 2.4 can therefore be small, too. Figure 1 depicts the 3×3 embedding example illustrated in Table 1. In this figure, sets S_1, S_2, and pixel y are used to calculate block difference d using Eq. (7). The watermark bit w is embedded into different pixels of the block B using Eq. (8) and Eq. (9).

Table 1. Examples of partition strategies of k = 2, 3, 4.

k	block definition	S_1	S_2	y_1	y_2
2	$\begin{array}{\|c\|c\|} \hline b_{1,1} & b_{1,2} \\ \hline b_{2,1} & b_{2,2} \\ \hline \end{array}$	$b_{1,1}$	ϕ	$b_{2,2}$	$b_{1,2}$
3	$\begin{array}{\|c\|c\|c\|} \hline b_{1,1} & b_{1,2} & b_{1,3} \\ \hline b_{2,1} & b_{2,2} & b_{2,3} \\ \hline b_{3,1} & b_{3,2} & b_{3,3} \\ \hline \end{array}$	$b_{1,2}$, $b_{2,1}$, $b_{3,2}$, $b_{2,3}$	$b_{1,3}$, $b_{2,2}$, $b_{3,1}$	$b_{1,1}$	$b_{3,3}$
4	$\begin{array}{\|c\|c\|c\|c\|} \hline b_{1,1} & b_{1,2} & b_{1,3} & b_{1,4} \\ \hline b_{2,1} & b_{2,2} & b_{2,3} & b_{2,4} \\ \hline b_{3,1} & b_{3,2} & b_{3,3} & b_{3,4} \\ \hline b_{4,1} & b_{4,2} & b_{4,3} & b_{4,4} \\ \hline \end{array}$	$b_{1,2}$, $b_{2,1}$, $b_{1,4}$, $b_{2,3}$, $b_{4,1}$, $b_{3,4}$, $b_{4,3}$	$b_{1,1}$, $b_{1,3}$, $b_{2,2}$, $b_{2,4}$, $b_{4,2}$, $b_{4,4}$	$b_{3,3}$	$b_{3,2}$

The proposed scheme partitions the cover image to combination of $k \times k$ blocks and pixels in each block are partitioned to two pixels y_1, y_2 and two sets S_1, S_2. Assume that sizes of cover image and watermark image be 512×512 and 128×128, respectively. Then, k can be 2, 3, or 4. Table 1 shows examples of *partition strategy* for k = 2, 3, and 4. The parameter k determines the scale of the visible watermark. Large k value acquires wide watermark effect. Step 3 adjusts the threshold for generating significant watermark effect and the number 30 in Eq. (7) is empirically determined. Since number of set S_2 is one less than number of set S_1, the block differences d_1 and d_2 acquired from Eq. (9), can be taken as neighboring differences between block pixels. In most image blocks, the block neighboring difference is small and the calculated pixels y'_1 and y'_2 in Eq. (10) and Eq. (11) can therefore be close to their neighboring pixels. The proposed scheme adopts conventional difference-expansion method [16] to embed and reversibly to extract the watermark bit in each block. Figure 1 depicts an example of embedding one watermark bit w into a 3×3 block B using the *partition strategy* defined in Table 1. In this figure, sets S_1, S_2, and two pixels y_1, y_2 are used to calculate block differences d_1 and d_2 using Eq. (9). The watermark bit w is embedded into difference between pixels of the block using Eq. (10). The sign bit, defined by the addition or subtraction in Eq. (8), is embedded into d_2 using Eq. (11). For reducing the watermark embedded distortion, the sign bit is embedded when the watermark bit is 1. Therefore, Step 4.6 is executed in part of blocks.

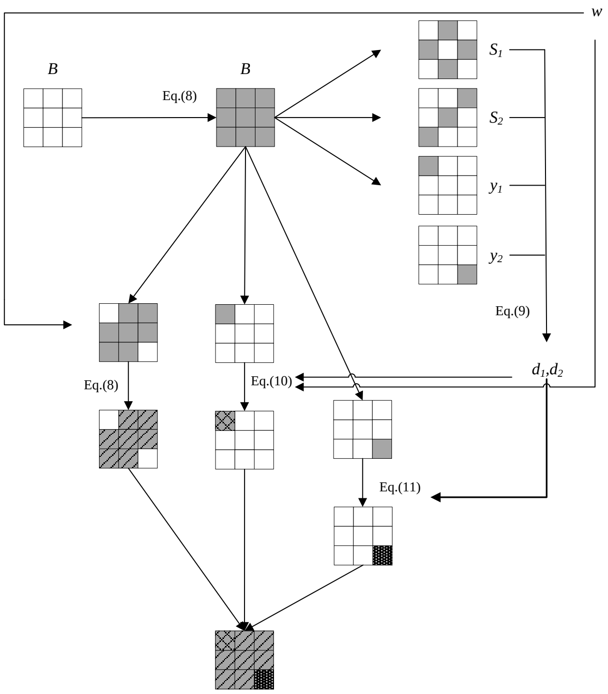


Fig. 1: An example of embedding a watermark bit w into a 3×3 block B.

3.2 Watermark Extracting and Host Image Recovery Algorithm

In the recovery algorithm, the test image is partitioned to non-overlapped $k\times k$ blocks, and pixels in each block are partitioned to two sets S_1, S_2 and two pixels y_1, y_2 by the same *partition strategy* used in embedding algorithm. The parameters *Base*, *ShiftSeed*, *ShiftQuantity* used in embedding algorithm are also needed in the recovery algorithm. The watermark extracting and cover image recovery algorithm is introduced as follows.

1. Partition the test image to non-overlapped $k\times k$ blocks, where k is the block size used in embedding algorithm.
2. Apply the following extraction steps to pixels of each block $x'(i,j)$ for extracting one watermark bit w from a $k\times k$ block B and recover to the original cover image block.

2.1 Calculate two block differences $\overline{d_1}$ and $\overline{d_2}$ between S_1, S_2 and y_1, y_2 using Eq. (12)

$$\overline{d_1} = y_1 - \left\lfloor \frac{\sum_{x'(i,j)\in\{S_1\cup S_2\}} x'(i,j)}{2\times\left\lfloor \frac{k^2-1}{2}\right\rfloor - 1} \right\rfloor$$
$$\overline{d_2} = y_2 - \left\lfloor \frac{\sum_{x'(i,j)\in\{S_1\cup S_2\}} x'(i,j)}{2\times\left\lfloor \frac{k^2-2}{2}\right\rfloor - 1} \right\rfloor \quad (12)$$

in which the definition of sets S_1, S_2, and pixels y_1, y_2 is defined by the same *partition strategy* which is used in embedding algorithm.

2.2 Extract the watermark bit w and the sign bit *sign* using Eq. (13)

$$w = \overline{d_1}\%2,\ \ sign = \overline{d_2}\%2 \quad (13)$$

2.3 Recover the original differences d_1' and d_2' using Eq. (14)

$$d_1' = \left\lfloor \frac{\overline{d_1}}{2} \right\rfloor,\ d_2' = \left\lfloor \frac{\overline{d_2}}{2} \right\rfloor \quad (14)$$

2.4 Use pre-defined parameters *Base* and *Shift*(i,j) to recover all pixels x in block and extracted watermark bit w using Eq. (15)

$$x(i,j) = \begin{cases} x'(i,j) - w\cdot(Base + Shift(i,j)), & if\ \ sign = 1 \\ x'(i,j) + w\cdot(Base + Shift(i,j)), & if\ \ sign = 0 \end{cases} \quad (15)$$

where *Base* and *Shift*(i,j) are the same defined in embedding algorithm.

2.5 Recover the shifted pixel y_1' using Eq. (16) and replace the pixel y in block B by the calculated y_1'

$$y_1' = \sum_{x_1(i,j)\in S_1} x_1(i,j) - \sum_{x_2(i,j)\in S_2} x_2(i,j) - d_1' \quad (16)$$

2.6 Recover the shifted pixel y_2' using Eq. (17) and replace the pixel y in block B by the calculated y_2' when the watermark bit w = 1

$$y_2' = \sum_{x_1(i,j)\in S_1} x_1(i,j) - \sum_{x_2(i,j)\in S_2} x_2(i,j) - d_2', \quad if\ \ w=1 \quad (17)$$

3. Collect all extracted watermark bits and recovered cover blocks to acquire the extracted watermark image and recovered cover image, respectively.

Two main steps are presented in the extracting algorithm. Steps 2.1 and 2.2 extract the watermark bit w and the sign bit *sign* embedded in each block. Steps 2.3 to 2.6 recover the block to original block. Especially in Step 2.6, the recovery step is only calculated when the extracted watermark bit is 1. Figure 2 shows an example of extracting one watermark bit w from a 3×3 block and recovering the block to original cover image block.

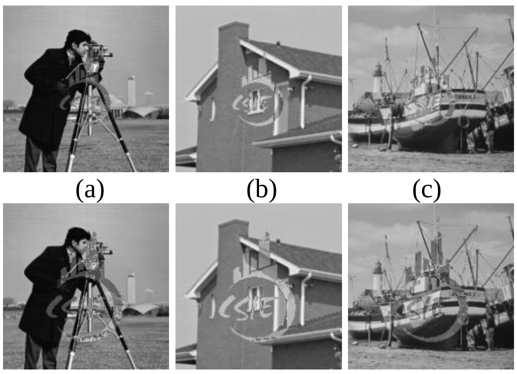
4. EXPERIMENTAL RESULTS

This section presents the experimental results, which were performed on MATLAB 7.11 installed in a Notebook with an Intel i7-2677 CPU and 4GB of RAM, of the proposed scheme. Figs. 3(a)-(c) show three grey-level cover images Cameramen, House, and Boat of size 512×512. Fig. 3(d) shows the binary watermark image of size 128×128. Since the sizes of cover image and watermark image are 512×512 and 128×128, respectively, the partitioned block size $k \times k$ in embedding and recovering algorithm can be 2×2, 3×3, or 4×4.

(a) (b) (c) (d)

Fig. 2: (a)-(c) three test grey images of size 512×512, (d) the binary watermark image of size 128×128.

Figs. 3 and 4 show the watermarked images under different parameters *k* and *Base*. Parameter *Shift*(*i*,*j*) is synthesized to a random integer within the range of $0 \le Shift(i,j) \le 9$. Parameter *k* determines the areas of watermarked region and parameter *Base* determines the significance of the watermark. Fig. 3 shows experimental results of *Base*=60 and *k*=2, 3, 4 by embedding the binary watermark image into three test grey images. The watermark is embedded into center of the cover image. Assigning parameter *k* being 2 leads to the watermarked region being 256×256 when the size of watermark image is 128×128. Large parameter *k* assignment generates wide watermark effect. Assigning parameters *k* being 3 or 4 acquire the watermarked regions of 384×384 or 512×512, respectively. Figs. 3.(a)-(c) show the watermarked region of 256×256 under parameter *k*=2. Figs. 3.(d)-(f) and 3.(g)-(i) show the watermarked region of 378×378 and 512×512 under parameters *k*=3 and 4, respectively. Moreover, Fig. 3 assigning parameter *Base* being 60 acquires clearly visual effect on embedded watermark image.

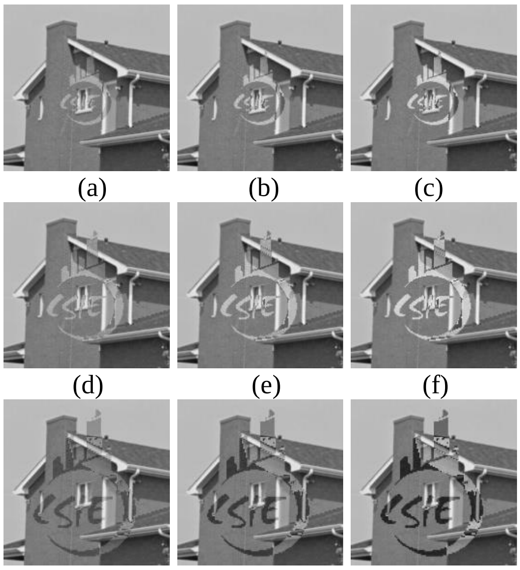


(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 3: Experimental results of *Base*=60, (a)-(c) three watermarked images under parameter *k*=2, (d)-(f) three watermarked images under parameter *k*=3, (g)-(i) three watermarked images under parameter *k*=4.



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 4: Experimental results of test image House by (a)-(c) *k*=2 and *Base*=40, 60, 80, (d)-(f) *k*=3 and *Base*=40, 60, 80, (h)-(i) *k*=4 and *Base*=40, 60, 80.

Fig 4 shows other experimental results of different *Base* values assignment on test image House. Figs. 4(a)-(c) show results of applying *shift*=40, 60, 80 under *k*=2, respectively. Moreover, Figs. 4(d)-(f) show results of applying *k*=3 and *shift*=40, 60, 80, respectively. These experimental results show that larger *shift* assignment leads to significant watermark effect according with more noises appearing. Moreover, the setting of *shift*=128 gains the most significant watermark effect with largest noises appearing probability. Therefore, the determination of parameter *shift* is a trade-off between watermark effect and noising appearing probability (noising effect).

Fig. 5 shows exceeding numbers, which are larger than 255 or smaller than 0, under different parameters and test images. The exceeding numbers should be recorded. Three 512×512 test images are performed under parameter *k*=2, 3, 4 and shift parameter Base = 10 to 90. Table 2 compares the requirements of logo image or cover image on related reversible and visible image watermarking schemes [5, 11, 13, 19, 21] to extract the embedded watermark.

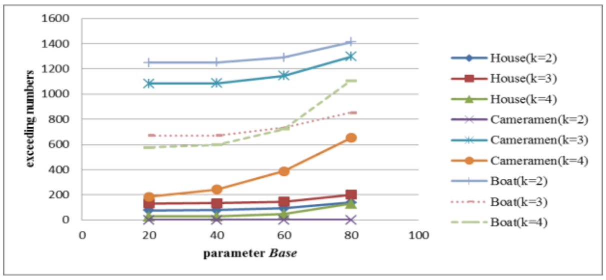


Fig. 5: Exceeding numbers under different test images and parameters

Table 2: Comparisons with other related reversible and visible image watermarking schemes [5, 11, 13, 19, 21]

properties	*recovering without binary logo image*	*recovering without original cover image*
Hu and Jeon [5]	O	O
Lin et al. [11]	×	O
Liu and Tsai [13]	O	×
Tsai and Chang [19]	×	O
Yang et al.[21]	O	O
Proposed scheme	O	O

5. CONCLUSION

This paper utilizes conventional difference-expansion scheme to present a reversible and visible image watermarking scheme. The proposed scheme uses the property that block mean difference is generally small and the reversibility of difference-expansion scheme to embed a watermark bit into a block. The small mean difference property in a block leads to few exceeding pixels that should be recorded. The usage of difference-expansion scheme prevents the requirement that logo or cover images are needed in watermark extracting and cover image recovery algorithm. Therefore, the proposed scheme exhibits significant properties on highly visual effect and logo image free. Reducing the recorded requirement on exceeding pixels merits our future study.

ACKNOWLEDGMENTS

This paper was partially supported by the National Science Council of the Republic of China under contract NSC 103-2221-E-032-051.

REFERENCES

[1] O.M. Al-Qershi, B.E. Khoo, “High capacity data hiding schemes for medical images based on difference expansion,” *The Journal of Systems and Software*, Vol. 84, 105–112, 2011.

[2] O.M. Al-Qershi, B.E. Khoo, “Two-dimensional difference expansion (2D-DE) scheme with a characteristics-based threshold,” *Signal Processing*, Vol. 93, 154–162, 2013.

[3] A.B. Alattar, “Reversible watermark using the difference expansion of a generalized integer transform,” *IEEE Trans. Image Processing*, Vol. 13, 1147–1156, 2004.

[4] I.C. Dragoi, D. Coltuc, “Local-Prediction-Based Difference Expansion Reversible Watermarking,” *IEEE Trans. on Image Processing*, Vol. 23, No. 4, 1779–1790, 2014.

[5] Y. Hu, B. Jeon, “Reversible Visible Watermarking and Lossless Recovery of Original mages,” *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 16, 1423–1429, 2006.

[6] Y.J. Hu, H.K. Lee, K.Y. Chen, J.W. Li, “Difference Expansion Based Reversible Data Hiding Using Two Embedding Directions,” *IEEE Trans. on Multimedia*, Vol. 10, No. 8, 1500–1512, 2008.

[7] Y. Hu, H.K. Lee, J. Li, “DE-based reversible data hiding with improved overflow location map,” *IEEE Trans.on Circuits Syst. Video Technol.*, Vol. 19, No. 2, 250–260, 2009.

[8] K. Jawad, A. Khan, “Genetic algorithm and difference expansion based reversible watermarking for relational databases,” *The Journal of Systems and Software*, Vol. 86, 2742–2753, 2013.

[9] H.J. Kim, V. Sachnev, Y.Q. Shi, J. Nam, H.G. Choo, “A Novel Difference Expansion Transform for Reversible Data Embedding,” *IEEE Trans. on Information Forensics and Security*, Vol. 3, No. 3, 456–465, 2008.

[10] C.F. Lee, H.L. Chen, H.K. Tso, “Embedding capacity raising in reversible data hiding based on prediction of difference expansion,” *The Journal of Systems and Software*, Vol. 83, 1864–1872, 2010.

[11] P.Y. Lin, Y.H. Chen, C.C. Chang, J.S. Lee, “Contrast-Adaptive Removable Visible Watermarking (CARVW) mechanism,” *Image and Vision Computing*, Vol. 31, 311–321, 2013.

[12] M. Liu, H.S. Seah, C. Zhu, W. Lin, F. Tian, “Reducing location map in prediction-based difference expansion for reversible image data embedding,” *Signal Processing*, Vol. 92, 819–828, 2012.

[13] T.Y. Liu, W.H. Tsai, “Generic Lossless Visible Watermarking—A New Approach,” *IEEE Trans. on Image Processing*, Vol. 19, 1224–1235, 2010.

[14] T.C. Lu, C.C., Chang, “Lossless nibbled data embedding scheme based on difference expansion,” *Image and Vision Computing*, Vol. 26, 632–638, 2008.

[15] T.C. Lu, C.C., Chang, Y.H. Huang, “High Capacity reversible hiding scheme based on interpolation, difference expansion, and histogram shifting,” *Multimedia Tools Application*, Vol. 72, 417–435, 2014.

[16] Z. Ni, Y.Q. Shi, N. Ansari, W. Su, “Reversible Data Hiding,” *IEEE Trans. on Circuits and Systems for Video Technology*, Vol. 16, 354–362, 2006.

[17] B. Ou, X., Li, Y., Zhao, R., Ni, “Reversible data hiding based on PDE predictor,” *The Journal of Systems and Software*, Vol. 86, No. 10, 2700–2709, 2012.

[18] J. Tian, “Reversible Data Embedding Using a Difference Expansion,” *IEEE Trans. on Circuits Systems and Video Technology*, Vol. 13, 890–896, 2003.

[19] H.M. Tsai, L.W. Chang, “Secure reversible visible image watermarking with authentication,” *Signal Processing: Image Communication*, Vol. 15, 10–17, 2010.

[20] H.C. Wu, C.C. Lee, C.S. Tsai, Y.P. Chu, H.R. Chen, “A high capacity reversible data hiding scheme with edge

prediction and difference expansion," *The Journal of Systems and Software*, Vol. 82, 1966–1973, 2009.

[21] Y. Yang, X.M. Sun, H.F. Yang, C.T. Li, R. Xiao, "A Contrast-Sensitive Reversible Visible Image Watermarking Technique," *IEEE Trans. on Circuits Systems Video Technology*, Vol. 19, 656–667, 2009.