Reversible Watermarking Based on Eight Improved Prediction Modes

S. W. Weng[1], J. S. Pan[2]

[1]School of Information Engineering
Guangdong University of Technology, P. R. China.

[2]Harbin Institute of Technology Shenzhen
Graduate School, Shenzhen, P. R. China.

ABSTRACT. *A new reversible watermark scheme based on eight prediction modes is presented. In each of eight prediction modes, any modification to some pixels which account for one-fourth of all pixels is not allowed for a single embedding process. By employing smartly these unchanged pixels, each to-be-predicted pixel in any mode is guaranteed to be surrounded by several pixels which constitute a local neighborhood. Each neighborhood has two applications. The first one is that when it is exploited to interpolate its corresponding surrounded pixel, the noticeable improvement in prediction accuracy is obtained. The second one is that its variance is employed to determine which classification (i.e., smooth or complex set) its surrounded pixel belongs to. For any to-be-predicted pixel, the number of embedded bits is adaptively determined according to this pixel's belonging. Experimental results reveal the proposed method is effective.*
Keywords: Reversible Watermarking, Eight Improved Prediction Modes

1. **Introduction.** In some applications, such as the fields of law enforcement, medical and military image system, any permanent distortion induced to the host images by data watermarking techniques is intolerable. In such cases, the original image is required to be recovered without any distortion after extraction of the embedded watermark. The watermarking techniques satisfying these requirements are referred to as reversible (or lossless) watermarking.

A considerable amount of research on reversible watermarking has been done over the last several years since the concept of reversible watermarking firstly appeared in the patent owned by Eastman Kodak [1]. In paper [2], reversible watermarking (RW) is classified into the following three categories: 1) fragile authentication 2) high embedding capacity and 3) semi-fragile authentication. The first two categories of RW are actually a special subset of fragile watermarking, namely, they cannot resist any attacks. From the literature up to today, it is observed that most of the RW scheme is grouped into the second category. Here, we provide a list containing multiple high-capacity RW schemes [3-15] which account for only part of the most representative ones over the last a few years.

RW based on difference expansion was first proposed by Tian [3]. Difference (between a pair of neighboring pixels) is shifted left by one unit to create a vacant least significant bit (LSB), and 1-bit watermark is appended to this LSB. This method is called Difference Expansion (DE). Alattar [4] generalized the DE technique by taking a set containing multiple pixels rather than a pair. Coltuc *et al.* proposed a threshold-controlled embedding

scheme based on an integer transform for pairs of pixels [5]. In Thodi's work [6], histogram shifting was incorporated into Tian's method to produce a new algorithm called Alg. D2 with a overflow map. Weng *et al.* proposed an integer transform based on invariability of the sum of pixel pairs [7]. Weng *et al* also proposed a new integer transform, and the embedding rate can approach to 1 bpp (bit per pixel) for a single embedding layer by overlapped pairing [8]. In Wang *et al.*'s method [9], a generalized integer transform and a payload-dependent location map were constructed to extend the DE technique to the pixel blocks of arbitrary length. Feng *et al.* presented a new reversible data hiding algorithm based on integer transform (proposed by [10]) and adaptive embedding [11]. Luo *et al.* first introduced an interpolation technique into RW to obtain the prediction-errors, which increased prediction-accuracy by using full-enclosing pixels to predict the current pixel [12]. In Li *et al.*'s method [13], an efficient reversible watermarking scheme was proposed by incorporating in prediction-error expansion two new strategies, namely, adaptive embedding and pixel selection. Wu *et al.* proposed reversible image watermarking on prediction errors by efficient histogram modification [14].

Four prediction mode are adopted in Wu *et al.*'s method. In each mode, nearly quarter of all pixel points are predicted by four pixels along two orthogonal directions: two pixels along 45° diagonal and two pixels along 135° diagonal. It is known that the closer the distance between two pixel points, the stronger the correlationship. Therefore, the correlationship between any to-be-predicted pixel and its four pixels along two diagonal directions is weaker than that between this pixel and its four neighboring pixels (i.e. top, left, right, bottom neighbors). Therefore, in the proposed method, to decrease prediction distortion, the class of pixels can also be predicted by their four neighboring pixels by designing eight improved prediction modes. Since this class of pixels accounts for close to one quarter of the entire image and the prediction becomes more accurate, the prediction performance is largely increased.

Eight different prediction modes relative to our former works [15] are designed in this paper. In each of eight improved prediction mode, each of almost three-fourth of all pixels is predicted by its four neighboring pixels, while the rest pixels are kept unaltered for a single embedding process. The proposed method is based on Wu *et al.*'s method, but in each of eight prediction modes, we aim at further improving prediction accuracy relative to Wu *et al.*'s method.

2. **Problem Statement and Preliminaries.** Please write down your section. When you cite some references, please give numbers, such as, ...In the work of [1-3,5], the problem of....... For more results on this topic, we refer readers to [1,4-5] and the references therein....

3. **The proposed method.**

3.1. **Eight Embedding Modes.** Given an 8-bit graylevel I with size $M \times N$, all the pixel points are partitioned into two non-overlapped sets: the to-be-predicted set and the predicting set. In each of four prediction modes, no modification can be done to all the pixels in the predicting set for a single embedding process. Otherwise, reversibility will be destroyed. Take mode 1 (refer to Fig. 1) for example, the pixels at locations $(2i, 2j)$, i.e., $I(2i, 2j)$, are classified into the predicting set, where $i, j \in \mathbb{Z}$, $1 \leq i \leq \lfloor \frac{M}{2} \rfloor$ and $1 \leq j \leq \lfloor \frac{N}{2} \rfloor$. Each of eight prediction modes works in a two-step process as illustrated in Figs. 1 and 2. Based on the principle that correlationship becomes weaker as the temporal distance between pixels increases, for each pixel in the to-be-predicted set, its several surrounding neighboring pixels from the predicting set, which are most strongly related to it, are used to predict it. Therefore, in the first step (see Fig. 1(a)), for each
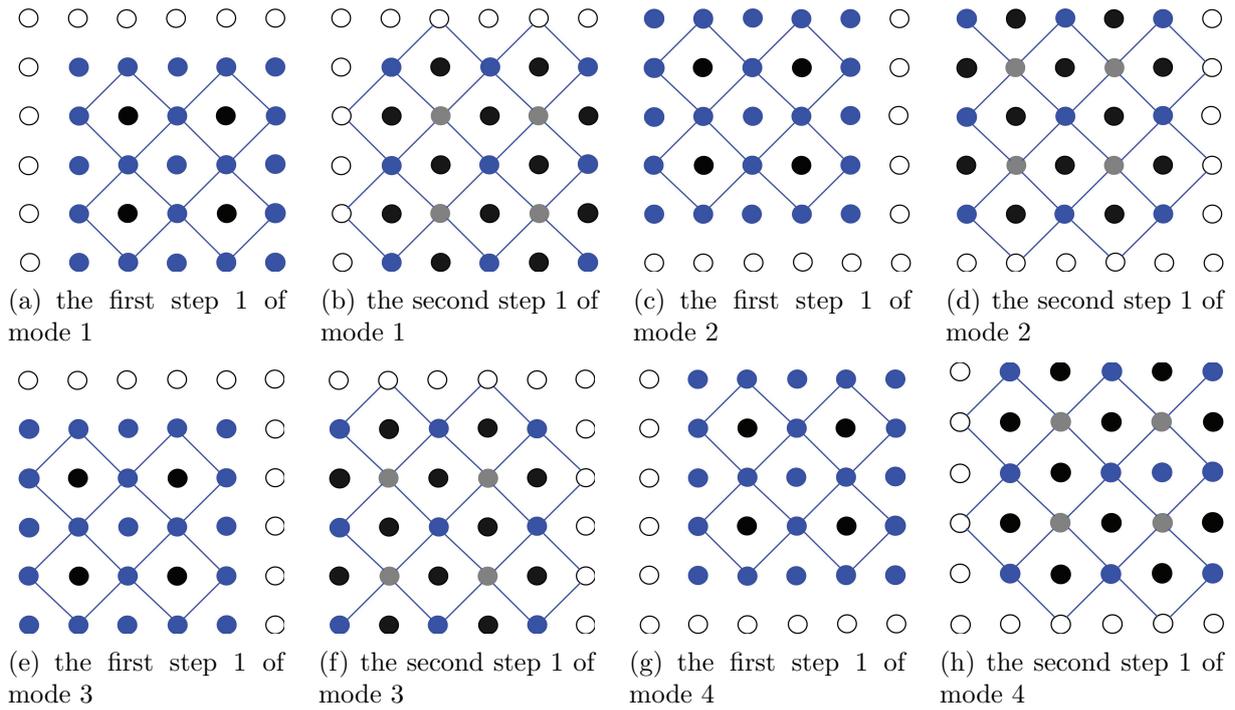
(a) the first step 1 of mode 1

(b) the second step 1 of mode 1

(c) the first step 1 of mode 2

(d) the second step 1 of mode 2

(e) the first step 1 of mode 3

(f) the second step 1 of mode 3

(g) the first step 1 of mode 4

(h) the second step 1 of mode 4

FIGURE 1. Prediction Modes 1-4

(a) the first step 1 of mode 5

(b) the second step 1 of mode 5

(c) the first step 1 of mode 6

(d) the second step 1 of mode 6

(e) the first step 1 of mode 7

(f) the second step 1 of mode 7

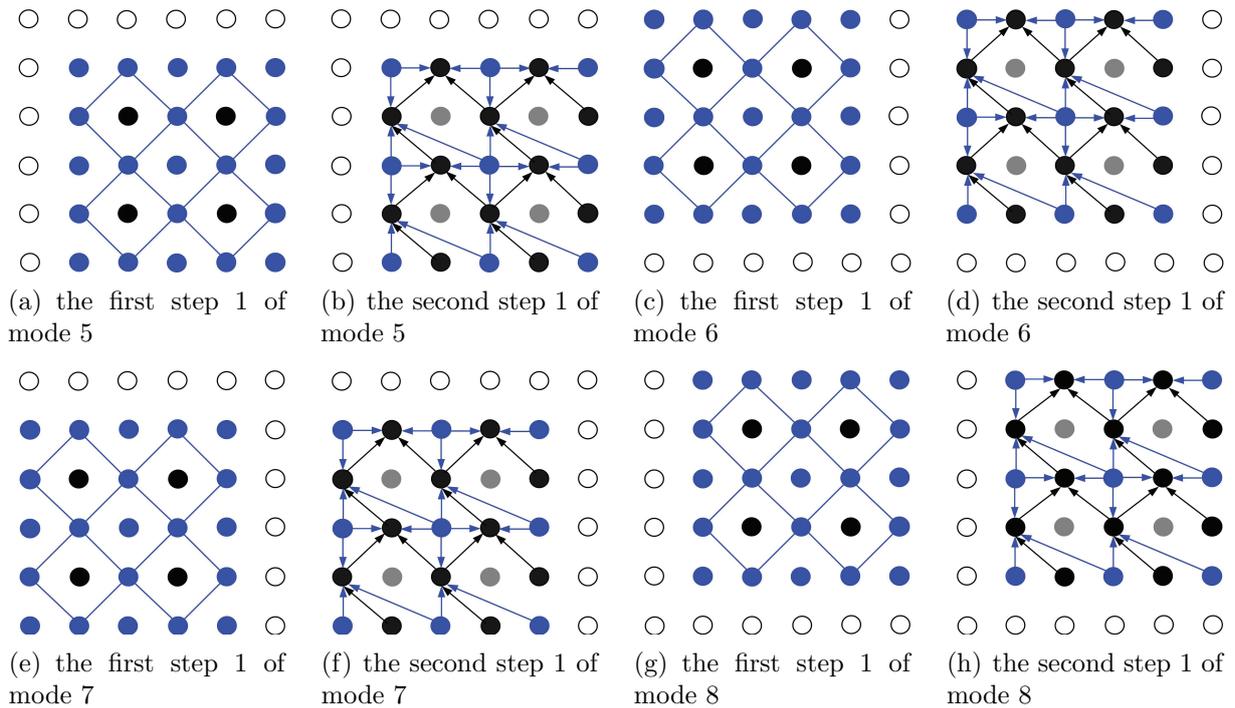(g) the first step 1 of mode 8

(h) the second step 1 of mode 8

FIGURE 2. Prediction Modes 5-8

pixel marked by black-filled circle (i.e., $I(2i + 1, 2j + 1)$ for $i, j \in \mathbb{Z}$, $1 \leq i \leq \lfloor \frac{M}{2} \rfloor$ and $1 \leq j \leq \lfloor \frac{N}{2} \rfloor$), its four neighbors marked in blue-filled circles are used to predict it. Since those pixels marked in black-filled circle have been modified in the first step, they are incapable of being used to predict the others. However, each of these pixels can be firstly predicted by its four diagonal neighbors. Then, the obtained predicted value can be used to predict the others. Here, they are marked by the gray-filled circles. Hence, in the

second step as shown in Fig. 1(b), each of the pixels at locations $(2i, 2j+1)$ or $(2i, 2j+1)$ for $1 \leq i \leq \lfloor \frac{M}{2} \rfloor$ and $1 \leq j \leq \lfloor \frac{N}{2} \rfloor$ can also be predicted by its four neighbors.

Both of modes 5 and 1 consist of the same first step. The difference in modes 5 and 1 is the second step. Data embedding is performed according to left-to-right column order and top-to-bottom row order. For instance, in data embedding process, for the pixel $I(2, 3)$, it is predicted prior to the pixels $I(3, 2)$ and $I(3, 4)$, and hence, its four neighboring and unchanged pixels are composed of $I(2, 2)$, $I(2, 4)$, $I(3, 2)$ and $I(3, 4)$. While for the pixel $I(3, 2)$, it also has four neighboring, unaltered pixels containing $I(2, 2)$, $I(4, 2)$, $I(4, 3)$ and $I(4, 4)$. Therefore, in the second step of Fig. 2(b), each of almost two-fourth of all pixels (i.e., the pixels at locations $(2i, 2j+1)$ or $(2i, 2j+1)$ for $1 \leq i \leq \lfloor \frac{M}{2} \rfloor$ and $1 \leq j \leq \lfloor \frac{N}{2} \rfloor$) can be predicted by its neighboring four original pixels. The eight prediction modes are designed so as to further improve prediction accuracy related to Wu *et al.* 's method.

For any to-be-predicted pixel x_c, if it has a surrounding four-pixel neighborhood, then it can be predicted by this neighborhood simply using the interpolation technique in [12]. The interpolated value of pixel x_c is calculated as

$$p_{i,j} = x_0' \times \frac{\sigma_2}{\sigma_1 + \sigma_2} + x_{90}' \times \frac{\sigma_1}{\sigma_1 + \sigma_2} \tag{1}$$

where

$$\begin{aligned} x_0' &= \frac{I_{i,j-1} + I_{i,j+1}}{2} \\ x_{90}' &= \frac{I_{i-1,j} + I_{i+1,j}}{2} \end{aligned} \tag{2}$$

and

$$\begin{aligned} \sigma_1 &= \frac{(I_{i,j-1}-\mu)^2 + (x_0'-\mu)^2 + (I_{i,j+1}-\mu)^2}{3} \\ \sigma_2 &= \frac{(I_{i-1,j}-\mu)^2 + (x_{90}'-\mu)^2 + (I_{i+1,j}-\mu)^2}{3} \end{aligned} \tag{3}$$

x_0' and x_{90}' are used to denote the interpolation values along two orthogonal directions (i.e., 0° diagonal and 90° diagonal), respectively. $I_{i-1,j}$, $I_{i,j-1}$, $I_{i,j+1}$ and $I_{i+1,j}$ are the top, left, right and bottom neighbors of x_c. These four pixels constitute a neighborhood resembling a diamond-shaped pixel cell denoted as I_C. The mean value in this neighborhood, denoted by μ, is calculated using the following formula: $\mu = \frac{I_{i-1,j} + I_{i+1,j} + I_{i,j-1} + I_{i,j+1}}{4}$. σ_1 and σ_2 are the variance estimation of interpolation errors in two diagonal directions, respectively. From Eq. (1), we can see obviously that more variance, which means weaker relation with the actual value, has less influence on the interpolated value.

Then the prediction error, denoted by $e_{i,j}$, is calculated via

$$e_{i,j} = x_c - p_{i,j} \tag{4}$$

3.2. **Process of producing classification and embedding strategy for each to-be-predicted pixel.** The local variance, denoted by σ, of cell I_C is used to estimate whether any to-be-predicted pixel has strong correlation with its surrounding cell or not. σ is obtained via

$$\sigma = \sqrt{\frac{(I_{i-1,j}-\mu)^2 + (I_{i,j-1}-\mu)^2 + (I_{i,j+1}-\mu)^2 + (I_{i+1,j}-\mu)^2}{4}} \tag{5}$$

When $\sigma < vT_h$, pixel x_c is regarded as having a strong relation with its cell, and therefore, it is classified into a smooth set, where vT_h is a predefined threshold which is used for distinguishing which classification x_c belongs to. Otherwise, x_c belongs to a complex set.

For any pixel in smooth set, if its prediction-error $e_{i,j}$ falls into the range of $[-pT_h, pT_h)$, $e_{i,j}$ is expanded twice according to Eq. (6), where pT_h is used for estimating the correlation

between the current pixel and its predicted value, w_b represents 2-bit watermark, i.e., $w_b \in \{0, 1, 2, 3\}$.

$$e'_{i,j} = 4 \times e_{i,j} + w_b \tag{6}$$

If $e_{i,j} \geq pT_h$ and $e_{i,j} \leq -pT_h - 1$, it will be shifted by $3pT_h$. Correspondingly, the watermarked prediction-error is

$$e'_{i,j} = \begin{cases} e_{i,j} - 3 \times pT_h & e_{i,j} \leq -pT_h - 1 \\ e_{i,j} + 3 \times pT_h, & e_{i,j} \geq pT_h \end{cases} \tag{7}$$

For any pixel in complex set, if its prediction-error $e_{i,j}$ belongs to the inner region $[-pT_h, pT_h)$, then 1- bit watermark is embedded into $e_{i,j}$ to get the watermarked value as

$$e'_{i,j} = 2 \times e_{i,j} + b \tag{8}$$

where $b \in \{0, 1\}$. If $e_{i,j}$ belongs to the outer region $(-\infty, -pT_h) \bigcup [pT_h, \infty)$, $e_{i,j}$ will be shifted by pT_h according to the following formula

$$e'_{i,j} = \begin{cases} e_{i,j} - pT_h & d \leq -pT_h - 1 \\ e_{i,j} + pT_h, & d \geq pT_h \end{cases} \tag{9}$$

Finally, the watermarked value of x_c, i.e., y_c, is computed by $y_c = p_{i,j} + e'_{i,j}$.

3.3. Data Embedding.
To prevent the overflow/underflow, each to-be-predicted pixel is classified into one of three sets: NO_s, S_P and C_P. Set S_P contains to-be-predicted pixels in smooth set whose watermarked values belongs to the inner region $[0, 255]$. Similarly, for any to-be-predicted pixel in complex set, if its watermarked value falls in the range of $[0, 255]$, then it belongs to set C_P. After any modification above (including Eq. (6) to Eq. (9)), if overflow/underflow occurs, then this pixel belongs to set NO_s.

A location map is generated in which the locations of the to-be-predicted pixel belonging to NO_s are marked by '1' while the others are marked by '0'. The location map is compressed losslessly by an arithmetic encoder and the resulting bitstream is denoted by $\mathcal{L}$. L_S is the bit length of $\mathcal{L}$.

For $x_c \in S_P$, if its prediction-error $e_{i,j}$ falls in the range of $[-pT_h, pT_h)$, then it is classified into E_2, otherwise, it belongs to H_2. Similarly, for $x_c \in C_P$, if $e_{i,j} \in [-pT_h, pT_h)$, then this $e_{i,j}$ is classified into E_1. Otherwise, it is included into H_1. For any pixel, if it is in NO_s, then it is kept unaltered, i.e., $y_c = x_c$. If x_c belongs to set E_2, then 2-bit watermark is embedded into its prediction-error according to Eq. (6). If x_c belongs to H_2, it will be shifted by $3pT_h$ according to Eq. (7). If x_c belongs to set E_1, then 1-bit watermark is embedded into its prediction-error according to Eq. (8). If x_c belongs to H_1, it will be shifted by pT_h according to Eq. (9).

After the first L_S pixels have been processed, the LSBs of y_c is appended to the payload $\mathcal{P}$ and replaced by the compressed location map $\mathcal{L}$. After all to-be-predicted pixels are processed, a new marked image I_w is obtained.

3.4. Data Extraction and Image Restoration.
The LSBs of the pixels in I_w are collected into a bitstream $\mathcal{B}$ according to the same order as in embedding. $\mathcal{B}$ is decompressed by an arithmetic decoder to retrieve the location map.

For each watermarked pixel y_c, if its location is associated with '0' in the location map, then it is ignored. Otherwise, the local variance σ of the cell I_C surrounding y_c is
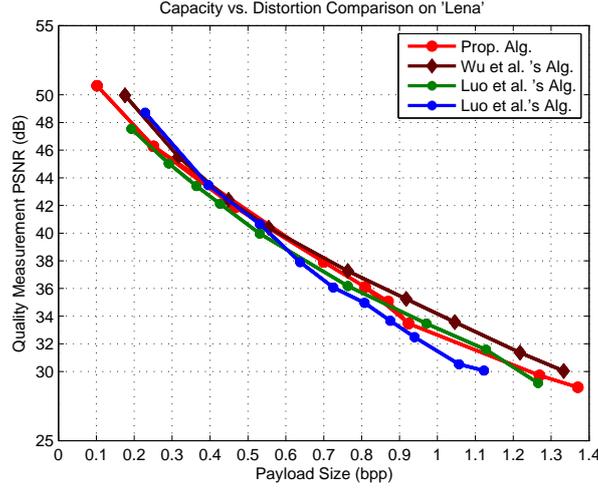
FIGURE 3. Capacity vs. Distortion Comparison on 'Barbara'

calculated, if $\sigma \leq vTh$, then $e_{i,j}$ is retrieved as follows.

$$
e_{i,j} = \begin{cases} \lfloor \frac{e'_{i,j}}{4} \rfloor & e'_{i,j} \in [-4pT_h, 4pT_h - 1] \\ e'_{i,j} + 3 \times pT_h & e'_{i,j} \leq -4pT_h - 1 \\ e'_{i,j} - 3 \times pT_h, & e'_{i,j} \geq 4pT_h \end{cases} \tag{10}
$$

where $e'_{i,j} = y_c - p_{i,j}$. Correspondingly, the embedded watermark is extracted by the following formula: $w_b = e'_{i,j} - 4\lfloor \frac{e'_{i,j}}{4} \rfloor$. Otherwise, $e_{i,j}$ is retrieved as follows.

$$
e_{i,j} = \begin{cases} \lfloor \frac{e'_{i,j}}{2} \rfloor & e'_{i,j} \in [-2pT_h, 2pT_h - 1] \\ e'_{i,j} + pT_h & e'_{i,j} \leq -2pT_h - 1 \\ e'_{i,j} - pT_h, & e'_{i,j} \geq 2pT_h \end{cases} \tag{11}
$$

1-bit watermark is retrieved as $b = e'_{i,j} - 2\lfloor \frac{e'_{i,j}}{2} \rfloor$. Finally, the retrieved prediction-error $e_{i,j}$ and the predicted value $p_{i,j}$ are substituted into Eq. (4) to obtain the original pixel value x_c.

4. **Experimental results.** The proposed reversible watermarking scheme is carried out in MATLAB environment. The capacity vs. distortion comparisons among the proposed method, Wu *et al.*'s, Luo *et al.*'s and Feng *et al.*'s methods are shown in Fig. 3. The 'Lena' image with size 512×512 are downloaded on the network (http://sipi.suc.edu/database). For the convenience of description, Wu, Luo and Feng are used as an abbreviation of Wu *et al.*'s, Luo *et al.*'s and Feng *et al.*, respectively.

For 'Lena', it also can be seen from Fig. 3 that our method performs well. The PSNR value of our method is the same as those of Wu's when the embedding rates is within the range of [0.4,0.8] bpp. It should be noticed that Wu's can provide a much higher embedding rate than the proposed one when the embedding rate is larger than 0.8 bpp. This is a drawback of our method. Improving embedding rates is beyond the scope of this paper, and we will investigate this issue in our future work.

5. **Conclusions.** A new reversible watermark scheme based on eight prediction modes is presented. In each of eight prediction modes, any modification to some pixels which account for one-fourth of all pixels is not allowed for a single embedding process. By

employing smartly these unchanged pixels, each to-be-predicted pixel in any mode is guaranteed to be surrounded by several pixels which constitute a local neighborhood. Experimental results reveal the proposed method is effective.

REFERENCES

[1] C. W. Honsinger, P. Jones P., M. Rabbani, and J. C. Stoffe, Lossless recovery of an original image containing embedded data, *US patent: 6278791*, 2001.

[2] Y. Q. Shi, Z. C. Ni, D. Zou, C. Y. Liang, and G. R. Xuan, Lossless data hiding: Fundamentals, algorithms and applications, *IEEE International Symposium on Circuits and Systems*, vol. 4. no. 4, pp. 345-370, 1998.

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

[4] A. M. Alattar, Reversible watermark using the difference expansion of a generalized integer transform, *IEEE trans. image processing*, vol. 13, no. 8, pp. 1147-1156, 2004.

[5] D. Coltuc, and J. M. Chassery, Very fast watermarking by reversible contrast mapping, *IEEE Signal Processing Letters*, vol. 14, no. 4, pp. 255-258, 2007.

[6] D. M. Thodi, and J. J. Rodrguez, Expansion embedding techniques for reversible watermarking, *IEEE trans. image processing*, vol. 16, no. 3, pp. 721-730, 2007.

[7] S. W. Weng, Y. Zhao, J. S. Pan, and R. R. Ni, Reversible watermarking based on invariability and adjustment on pixel pairs, *IEEE Signal Processing Letters*, vol. 45, no. 20, pp. 1022-1023, 2008.

[8] S. W. Weng, Y. Zhao, R. R. Ni, and J. S. Pan, Parity-invariability-based reversible watermarking, *IET Electronics Letters*, vol. 1, no. 2, pp. 91-95, 2009.

[9] X. Wang, X. L. Li, B. Yang, and Z. M. Guo, Efficient generalized integer transform for reversible watermarking, *IEEE Signal Processing Letters*, vol. 17, no. 6, pp. 567-570, 2010.

[10] X. Wang, X. L. Li, and B. Yang, High capacity reversible image watermarking based on in transform, *Proc. of 17th IEEE International Conference on Image Processing*, pp. 217-220, 2010.

[11] F. Peng, X. Li, and B. Yang, Adaptive reversible data hiding scheme based on integer transform, *Signal Processing*, vol. 92, no. 1, pp. 54-62, 2012.

[12] L. Luo, Z. Chen, M. Chen, X. Zeng, and Z. Xiong, Reversible image watermarking using interpolation technique, *IEEE Transactions on Information Forensics and Security*, vol. 5, no. 1, pp. 187-193, 2010.

[13] X. L. Li, B. Yang, and T. Y. Zeng, Efficient reversible watermarking based on adaptive prediction-error expansion and pixel selection, *IEEE Trans. Image Processing*, vol. 20, no. 12, pp. 3524-3533, 2011.

[14] H. T. Wu, and J. W. Huang, Reversible image watermarking on prediction errors by efficient histogram modification, *Signal Processing*, vol. 12, no. 12, pp. 3000-30009, 2012.

[15] S. W. Weng, and J. S. Pan, Reversible watermarking based on multiple prediction modes and adaptive watermark embedding, *Multimedia Tools and Applications*, Springer, 2013.