# REVERSIBLE WATERMARKING BASED ON PMO OF TRIPLETS

*Shaowei Weng*[1], *Yao Zhao*[1], *Jeng-Shyang Pan*[2] *and Rongrong Ni*[1]

[1]Institute of Information Science, Beijing Jiaotong University, Beijing 100044, P.R.China
[2]Department of Electronic Engineering, Kaohsiung University of Applied Sciences, Taiwan

## ABSTRACT

A reversible watermarking algorithm based on piecewise modification operation(PMO) of triplets is proposed in this paper. PMO is designed to embed a bit into any two pixels while remaining the intensity sum of pixels unchanged. Two bits are embedded into a triplet by repeatedly using PMO on every two neighboring pixels. Another advantage of using PMO is that the capacity consumed by the additional information can be largely decreased. As a result, the embedding capacity is considerably increased. A series of experiments is conducted to verify effectiveness and advantages of the proposed approach.

***Index Terms—*** Reversible watermarking, PMO of triplets

## 1. INTRODUCTION

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

A considerable amount of research on reversible watermarking has been done over the last several years, and some reversible watermarking algorithms [1–7] have been proposed in the literature since the concept of the reversible watermarking firstly appeared in the patent by Honsinger *et al.* [8]. In the method [3], the difference expansion method firstly proposed in the paper [2] is extended to embed two bits in a triplet of pixels. For the purpose of controlling the embedding distortion, a threshold is set to the differences of triplets, and only those triplets with differences less than this threshold are expanded during embedding if no overflow or underflow occurs. A location map is created to record the positions of the expandable triplets, which needs to be losslessly compressed to make space for data embedding. Therefore, the embedding capacity depends on the efficiency of location map compression. However, when the threshold is small, the compressibility of the location map is very low. As a result, the embedding capacity is low.

To solve the problems above, a novel reversible embedding scheme based on PMO of triplets is proposed in this paper. PMO is designed to embed a bit into any two pixels while remaining the intensity sum of pixels unchanged. Its introduction is able to largely increase the number of the triplets available for embedding. According, a considerable bias between the numbers of 1s and 0s in the location map is formed. As a result, the strong lossless compression is allowed. The compressibility is largely increased especially when the threshold is a small value. As revealed in the experimental results, the performance of the proposed method exceeds those of difference expansion respectively in papers [2] and [3].

The rest of the paper is organized as follows. In Section 2, the proposed scheme is described. The experimental results are shown in Section 3 and finally we conclude the paper in Section 4.

## 2. REVERSIBLE WATERMARKING SCHEME

For an 8-bit graycale image $I$ with size $r \times c$, three neighboring pixels are grouped into triplets $t = (x, y, z)$ according to a predefined order. A piecewise modification operation(PMO) is proposed so that 1-bit watermark is embedded into two neighboring pixels while the sum of pixels remains unchanged. Two watermark bits are embedded into a triplet by repeatedly using PMO.

### 2.1. The Piecewise Modification Operation(PMO)

For a digital signal $s$, the piecewise modification operation denoted by $M(s, b)$ is defined as

$$s_w = M(s, b) = \begin{cases} \lfloor \frac{s}{2} \rfloor + b & |s| < T_h \\ \mathrm{sign}(s) \times C_o & |s| \geq T_h \end{cases} \quad (1)$$

where $b$ is 1-bit watermark information, i.e., $b \in \{0, 1\}$, $\mathrm{sign}(s)$ is 1 if $s > 0$, 0 if $s = 0$, and -1 if $s < 0$, $T_h$ is a given threshold. The constant value denoted by $C_o$ is $\lfloor \frac{T_h}{2} \rfloor$ if the threshold $T_h$ is set to an even value. Otherwise, it is $\lfloor \frac{T_h}{2} \rfloor + 1$.

This work was supported in part by National Natural Science Foundation of China (No. 90604032, No. 60776794, No. 60702013), Specialized Research Fund for the Doctoral Program of Higher Education, Program for New Century Excellent Talents in University, Specialized Research Foundation of BJTU, 973 program (No. 2006CB303104).

### 2.2. The Applying of PMO

For $t = (x, y, z)$, PMO is applied to the first two neighboring pixels, i.e., $x$ and $y$, in a way that the sum of pixels remains unchanged as described in Eq. (2). Obviously, $x + y$ equals $x^{'} + y^{'}$.

$$\begin{aligned} x^{'} &= x + M(d_1, b_1) \\ y^{'} &= y - M(d_1, b_1) \end{aligned} \tag{2}$$

where $d_1 = x - y$. If $|d_1| < T_h$, then $d_1^{'} = x^{'} - y^{'}$ after embedding will fall within a range denoted by $R$ which is related to $T_h$. If $T_h$ is an even number, then $R$ is $[-2T_h + 1, 2T_h - 1]$, while $R$ is $[-2T_h + 1, 2T_h]$ if $T_h$ is odd. For $|d_1| \geq T_h$, since $d_1^{'}$ is changed by $2 \times C_o$, the changed difference value $d_1^{'}$ will fall in another range $\tilde{R}$, where $\tilde{R} \bigcup R = \phi$. Hence, the marked differences can be differentiated from those changed ones without any additional information in the extraction process.

PMO is reapplied to the resulting value $y^{'}$ and $z$ based on Eq. (3). As a result, two bits can be embedded into a triplet.

$$\begin{aligned} y^{''} &= y^{'} + M(d_2, b_2) \\ z^{'} &= z - M(d_2, b_2) \end{aligned} \tag{3}$$

where $d_2 = y^{'} - z$.

On the decoding side, $x$ and $y$ can be correctly restored only after $y^{'}$ and $z$ are successfully retrieved. A detailed description about how to retrieve $y^{'}$ and $z$ is given below.

$y^{'} + z$, $d_2$ have the same parity since $y^{'} + z = d_2 + 2z$ and $2z$ is an even number. That is, $LSB(y^{'} + z)$ equals $LSB(d_2)$, where the operator $LSB(\cdot)$ denotes the least significant bit. It is similar for $d_2^{'}$ and $(y^{''} + z^{'})$, where $d_2^{'} = y^{''} - z^{'}$. Hence, we have $LSB(y^{'} + z)$=$LSB(d_2)$=$LSB(y^{''} + z^{'})$=$LSB(d_2^{'})$.

When $|d_2| < T_h$, $d_2^{'}$ can be calculated as follows

$$d_2^{'} = d_2 + 2 \times \lfloor \frac{d_2}{2} \rfloor + 2b_2 \tag{4}$$

Since $d_2$ may be denoted as $d_2 = 2 \times \lfloor \frac{d_2}{2} \rfloor + LSB(d_2)$ and $LSB(d_2) = LSB(d_2^{'})$, then Eq. (4) is simplified as $d_2^{'} = 2d_2 + 2b_2 - LSB(d_2^{'})$. Accordingly, $d_2$ is calculated as

$$d_2 = \frac{d_2^{'} + LSB(d_2^{'})}{2} - b_2 \tag{5}$$

The value of $d_2$ and watermark bit can be uniquely deduced because the parity of $d_2$ is known and $b_2$ is a binary number. For example, if $y^{'} = 4$, $z = 3$ and $b_2 = 1$, then $y^{''} = 5$, $z^{'} = 2$ after embedding. On the receiving side, $\frac{d_2^{'} + LSB(d_2^{'})}{2}$ is calculated as 2. The parity of $d_2$ can be guaranteed to be same as that of $d_2^{'}$ if and only if $b_2 = 1$. As a result, watermark bit $b_2$ is correctly extracted and the value of $d_2$ is obtained.

For $|d_2| \geq T_h$, $d_2 = d_2^{'} - 2 \times \text{sign}(d_2^{'}) \times C_o$.

Once $d_2$ and $y^{'} + z$ are obtained, $y^{'}$ is calculated as $\lfloor \frac{(y^{'} + z) + d_2}{2} \rfloor$ while $z$ is $\lfloor \frac{(y^{'} + z) - d_2}{2} \rfloor$.

The restoration of $x$ and $y$ may be referred to that of $y^{'}$ and $z$.

### 2.3. Classification

Each triplet $t = (x, y, z)$ is classified into one of two sets $\mathbf{S_1}$ and $\mathbf{S_2}$.

$\mathbf{S_1}$**:** For all values of $b_1 \in \{0, 1\}$ and $b_2 \in \{0, 1\}$, if the transformed pixel triplet $(x^{'}, y^{''}, z^{'})$ based on Eq. (2), ( 3) is a pixel triplet, i.e., $0 \leq x^{'} \leq 255$, $0 \leq y^{''} \leq 255$ and $0 \leq z^{'} \leq 255$, then the pixel triplet $(x, y, z)$ is classified to be in $\mathbf{S_1}$.

$\mathbf{S_2}$**:** The rest of triplets belongs to $\mathbf{S_2}$.

For each $t$ in $\mathbf{S_1}$, if $|d_1| < T_h$ or $|d_2| < T_h$, then $d_1$ or $d_2$ is capable of carrying 1-bit watermark as described in Eq. (2), (3), whereas for $|d_1| \geq T_h$ or $|d_2| \geq T_h$, their introduction will result in an increase in the number of triplets belonging to $\mathbf{S_1}$. As a result, most locations are marked by '1' in the location map, and only a few is marked by '0'. Accordingly, the location map is efficiently compressed at a low rate. $L_S$ is used to denote the bit length of the compressed location map.

### 2.4. Data Embedding

Let the set $\mathcal{D}$ contain all the differences capable for providing space for payload, that is, $\mathcal{D}$ is composed of two subsets: $\mathcal{D}_1 = \{d_1 \in \mathcal{D} : |d_1| < T_h\}$ and $\mathcal{D}_2 = \{d_2 \in \mathcal{D} : |d_2| < T_h\}$. The LSBs of $L_S$ pixels are necessary for carrying the compressed map and the original LSBs after LSB-substitution are also needed to be embedded. Therefore, the maximum hiding capacity of the proposed method is $C_{ap}$ = $(\|\mathcal{D}\| - 2 \times L_S)$ bits, where the operator $\| \cdot \|$ represents the length of a sequence or the cardinality of a set.

Assume the length of the desired payload $\mathcal{P}$ is $\mathcal{P}_L$, and $\mathcal{P}_L \leq C_{ap}$. Let the total number of triplets processed in the embedding procedure be $\xi$. If $\mathcal{P}_L = C_{ap}$, then $\xi = \lfloor \frac{r \times c}{3} \rfloor$. Otherwise, $\xi$ is only a part of all $\lfloor \frac{r \times c}{3} \rfloor$ triplets which is determined as follows: orderly select triplets from the total $\lfloor \frac{r \times c}{3} \rfloor$ triplets; when the selected triplets contains $\mathcal{P}_L + 2 \times L_S$ differences belonging to $\mathcal{D}$, then their number is $\xi$. $\xi$ is the key to extract the embedded message and restore the original image. It can be separately sent to the receiver or embedded into the original image similar to the compressed map. For simplicity, it is sent to the receiver in this paper. Denote the set of all $\xi$ triplets as $t_1, t_2, \cdots, t_\xi$.

- Embedding the compressed map $\mathcal{L}$

  A unique EOS symbol is added at its end of the compressed map, denoted as $\mathcal{L}$ with length $L_S$. The first $\lceil \frac{L_S}{3} \rceil$ of $\xi$ triplets are used to carry bitstream $\mathcal{L}$ by simple LSB replacement. The $L_S$ LSBs to be replaced are also needed to be embedded into the original image so as to reconstruct the original image in the decoding side. They are appended to $\mathcal{P}$. Hence, the to-be-embedded bits are composed of two parts: the payload $\mathcal{P}$ and the bit sequence $\mathcal{C}$ consisting $L_S$ appended bits.

- Embedding payload $\mathcal{P}$ and bit sequence $\mathcal{C}$

  For each of $\xi - \lceil \frac{L_S}{3} \rceil$ pixel triplets, if it is in $\mathbf{S_2}$, then it is kept unaltered, i.e., $x^{'} = x, y^{''} = y, z^{'} = z$. Otherwise, for each $t$, if $|x - y| < T_h$, one to-be-embedded bit is embedded into it, while for $|x - y| \geq T_h$, $x$ and $y$ are respectively incremented or decremented by $C_o$ according to Eq. (2). This is similar for $y^{'}$ and $z$ based on Eq. (3). After $\xi$ triplets are processed, a new marked image $I_w$ is obtained.

## 2.5. Data Extraction and Image Restoration

A one-dimensional triplet list $\{\tilde{t}_1, \tilde{t}_2, \cdots, \tilde{t}_{\lceil \frac{L_S}{3} \rceil}, t^{'}_{\lceil \frac{L_S}{3} \rceil + 1}, \cdots, t^{'}_{\lfloor \frac{r \times c}{3} \rfloor}\}$ is formed by tripling all the pixels in $I_w$ according to the same order as in embedding. Since only first $\xi$ triplets are processed during the embedding process, the extraction process mainly consisting of the following three steps is just applied to the first $\xi$ marked triplets.

- Location Map Extraction

  The LSBs of all the triplets are collected into a bitstream $\mathcal{B}$. By identifying the EOS symbol in $\mathcal{B}$, the bits from the start until the EOS comprising the compressed map $\mathcal{L}$ are decompressed by an arithmetic decoder to retrieve the location map.

- Restoration of the first $L_S$ marked pixels

  Since the LSBs of the first $L_S$ marked pixels in $I_w$ have been overwritten and inserted into the residual $\xi - \lceil \frac{L_S}{3} \rceil$ triplets, they must be obtained prior to restoration of these $\lceil \frac{L_S}{3} \rceil$ triplets. Therefore, data extraction is carried out in inverse order of this list so as to obtain the original LSBs.

  For each $t^{'} = (x^{'}, y^{''}, z^{'})$, if $t^{'}$'s location is associated with '0' in the location map, then it is ignored. Otherwise, $y^{'} - z$ is retrieved as follows:

$$y^{'} - z = \begin{cases} \frac{d_2^{'} + LSB(d_2^{'})}{2} - b_2 & d_2^{'} \in R \\ d_2^{'} - 2 \times \mathrm{sign}(d_2^{'}) \times C_o & d_2^{'} \in \tilde{R} \end{cases} \tag{6}$$

  where $d_2^{'} = y^{''} - z^{'}$. For $d_2^{'} \in R$, the detailed description for the restoration of $y^{'} - z$ and the extraction of $b_2$ is referred to in subsection 2.2. $y^{'}$ and $z$ can be respectively calculated due to $y^{'} + z = y^{''} + z^{'}$.

  $x$ and $y$ are obtained from $x^{'}$ and $y^{'}$ in a similar way as in Eq. (6). If $x^{'} - y^{'} \in R$, 1-bit watermark can also be exacted. The extraction procedure is terminated until the number of the extracted bits reaches $L_S$. These bits are reversely ordered to form a bitstream $\mathcal{O} = o_1 o_2 \cdots o_{L_S}$. For the first $L_S$ pixels of $\{\tilde{t}_1, \tilde{t}_2, \cdots, \tilde{t}_{\lceil \frac{L_S}{3} \rceil}\}$, their LSBs are replaced according to Eq. (7) to obtain the integral marked pixels.

$$x_i = 2\lfloor \frac{\tilde{x}_i}{2} \rfloor + o_i \tag{7}$$

  where $o_i \in \mathcal{O}, 1 \leq i \leq \mathcal{L}_S, \tilde{x}_1, \cdots \tilde{x}_{L_S}$ denote the first $L_S$ marked pixels of $\{\tilde{t}_1, \tilde{t}_2, \cdots, \tilde{t}_{\lceil \frac{L_S}{3} \rceil}\}$, $\tilde{x}_i$ is the ith marked pixel.

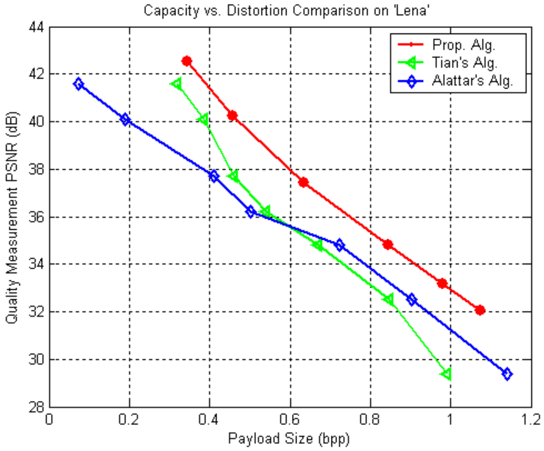

**Fig. 1**. Capacity vs. Distortion Comparison on 'Lena'

- Extraction of payload $\mathcal{P}$

  Except for the processed triplets in Step.2, the residual ones are processed based on Eq. (6) to retrieve correspondingly original ones and extract $\mathcal{P}$. Finally, the original image $I$ is obtained.

## 3. EXPERIMENTAL RESULTS

The proposed scheme is tested on various $512 \times 512$ natural images ranging from smooth images(e.g., 'Lena') to rough images(e.g., 'Baboon'). The capacity vs. distortion comparisons among the proposed method, Alattar's and Tian's are shown in Figs. 1 to 4.

As reported in Fig. 1, the top curve is the proposed method. By embedding a payload of the same size (bit length), the PSNR value of the proposed method is about 1-2**dB** greater than that of the others. For example, when the threshold is set to 4, the length of the compressed map is only 40 bit, and therefore, the embedding rate of close to 0.4587 bpp (bit per pixel) is achieved with 40.2517**dB** in PSNR. This bit rate is higher than those of Tian's and Alattar's at the same PSNR. For 'Boat' image, Fig. 2 shows that the proposed method achieves higher embedding capacity with lower embedding distortion than the other methods.

'Barbara' and 'Baboon' images represent images with large areas of complex texture, so the obtained bit-rate is slightly lower at the same PSNR. As shown in Figs. 3 and 4, the proposed method achieves higher embedding capacity at all PSNRs than the other methods.

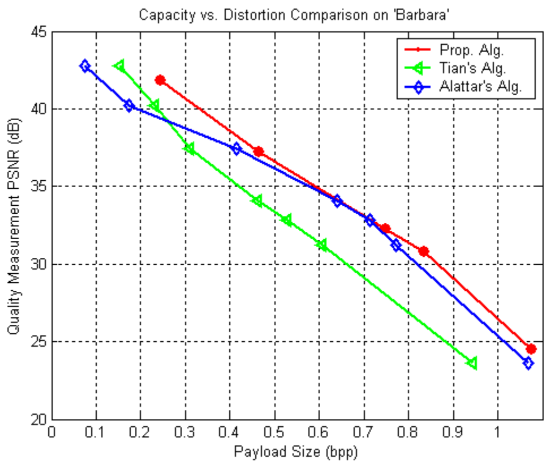

**Fig. 2**. Capacity vs. Distortion Comparison on 'Barbara'

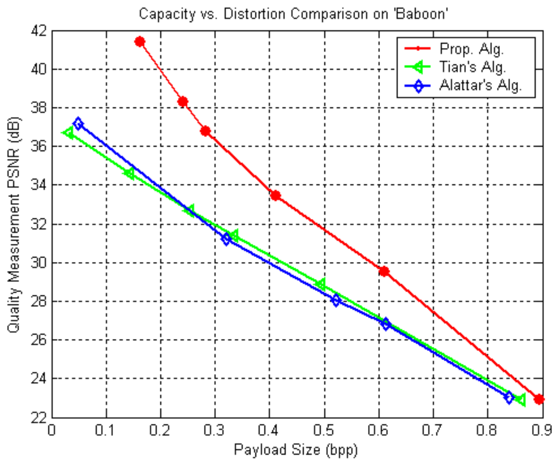

**Fig. 3**. Capacity vs. Distortion Comparison on 'Baboon'

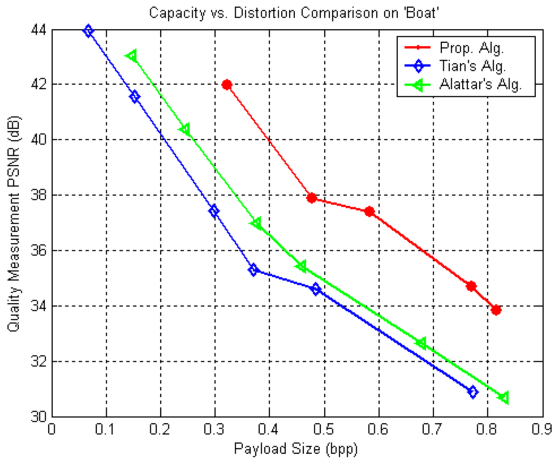

**Fig. 4**. Capacity vs. Distortion Comparison on 'Boat'

## 4. CONCLUSIONS

A novel reversible data hiding scheme based on PMO of triplets is presented in this paper. PMO is designed to embed a bit into any two pixels while remaining the sum of pixels unchanged. Two bits are embedded into a triplet by repeatedly using PMO on every two neighboring pixels. The applying of PMO is able to largely increase the number of the triplets in $\mathbf{S_1}$, so that high compressibility of the location map is achieved. The experimental results reveal that the proposed method outperforms Tian's and Alattar's both in hiding capacity and PSNR value.

## 5. REFERENCES

[1] J. Fridrich, M. Goljan, and R. Du, "Invertible authentication," in *SPIE Proceedings of Security and Watermarking of multimedia Content*, San Jose, 2001, vol. 3971, pp. 197–208.

[2] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Transaction on Circuits and Systems for Video Technology*, vol. 13, pp. 890–896, 2003.

[3] A. M. Alattar, "Reversible watermark using difference expansion of triplets," in *Proceedings of IEEE International Conference on Image Processing*, Barcelona, Spain, 2003, pp. 501–504.

[4] G.R. Xuan, C.Y. Yang, Y.Z. Zhen, and Y.Q. Shi, "Reversible data hiding using integer wavelet transform and companding technique," in *Proceedings of International Workshop on Digital Watermarking*, 2004.

[5] M. Thodi and J.J. Rodrłguez, "Prediction-error based reversible watermarking," in *Proceedings of IEEE International Conference on Image Processing*, 2004, vol. 3, pp. 1549–1552.

[6] D. Coltuc and A. Tremeau, "Simple reversible watermarking schemes," *SPIE Proceedings, Security, Steganography and Watermarking of Multimedia Contents - VII*, vol. 5681, pp. 561–568, 2005.

[7] M.U. Celik, G. Sharma, and A.M. Tekalp, "Lossless watermarking for image authentication a new framework and an implementation," *IEEE Transaction on Image Processing*, vol. 15, pp. 1042–1049, 2006.

[8] C.W. Honsinger, P. Jones, M. Rabbani, and J. C. Stoffel, *Lossless Recovery of an Original Image Containing Embedded Data*, US patent:6278791, 2001.