

Reversible Data Hiding Based on Prediction-Error Expansion

Hsien-Wen Tseng

Department of Information Management,
Chaoyang University of Technology,
Assistant Professor
e-mail : hwtseng@cyut.edu.tw

Chi-Pin Hsieh

Department of Information Management,
Chaoyang University of Technology,
Master
e-mail : s9414624@cyut.edu.tw

Abstract

For some applications such as satellite and medical images, reversible data hiding is the best solution to provide copyright protection or authentication. Being reversible, the decoder can extract the hidden data and recover original image without distortion. In this paper, a reversible data hiding scheme based on prediction-error expansion is proposed. The predictive value is computed by using various predictors. Message data is concealed in the digital image by exploiting the expansion of the difference between a pixel and its predictive value. Experimental results show that our method is capable of providing a great embedding capacity without making noticeable distortion. Besides, the proposed scheme is also applicable to various predictors.

Keywords: Reversible data hiding; prediction; difference expansion

1. Introduction

In recent years, a special kind of digital watermark has draw lots of interest, called lossless (which is also called reversible, invertible, erasable, distortion-free, etc.) data embedding technique. It not only declares the ownership of the digital media by embedding the digital watermark into the original image but also can restore the original image from the watermarked image completely. Subsequently, the watermark can be extracted or detected from the watermarked media to authenticate the completeness of the media. This kind of data hiding scheme is suitable for some specific applications where images are sensitive to further processing, such as medical image, satellite image, and artwork preservation.

In 2001, Honsinger *et al.* [5] proposed the first lossless data hiding concept to provide the

purpose of lossless authentication. However, Honsinger *et al.*'s scheme is not satisfactory in image quality. It suffers seriously from the salt-and-pepper visual artifacts. Then Vleeschouwer *et al.* [10] used the patchwork algorithm and the concept of a circular histogram to solve the noise problem. Although this scheme can survive high quality JPEG compression, its capacity is very limited. Ni *et al.* [7] proposed a reversible data hiding method based on histogram modification. Pixel values which are larger than peak point (P) and smaller than zero point (Z) are added by 1 to produce a gap at pixel value $P+1$. Message data is embedded by modifying the pixel value P . Experimental results demonstrate that the lower bound of the PSNR of a watermarked image is 48.13dB.

Fridrich *et al.* [3] compressed the least significant bit plane of selected pixels by using lossless compression, and then combined the compressed result with the watermark to produce embedded bitstream. Finally, the embedded bitstream is embedded into the cover work by using LSB. Their algorithm achieved reversibility, but its capacity is very limited. In 2002, Fridrich *et al.* [4] proposed another scheme to improve the embedded payload and visual quality. Each disjointed group of the cover work is categorized into one of the three sets: regular (R), singular (S), and unusable (U). The group types R and S are used to embed 1 and 0 separately. During embedding procedure, flipping function is used to flip the embedded group into the specific set. Finally, the R and S sets are losslessly compressed and embedded into the cover work. Celik *et al.* [2] improved Fridrich *et al.*'s scheme in terms of embedding capacity and visual quality and presented reversible data hiding by using generalized-LSB embedding. The scheme first converts binary to L-ary, then lossless compression (CALIC) is used to compress the quantization residues as side information. This scheme efficiently compresses quantization

residues to obtain high embedding capacity.

Tian [9] used the integer Haar wavelet transform to produce the redundancy in an image and embedded information by expanding the difference values of a pair of pixels which will not overflow or underflow after difference expansion. Tian's scheme achieves very high data embedding rates, approaching one bit per pair of pixels. Alatter [1] extended Tian's scheme to embed the secret bit by expanding the difference values of spatial triplets or spatial quads. Thodi *et al.* [8] used the predictor to produce prediction-errors in an image and exploited the difference expansion of the prediction-errors to embed the message data. Experimental results demonstrate that embedding rate of this algorithm almost achieve one bit per pixel. Kamstra *et al.* [6] presented two reversible schemes. One presents the least significant bit prediction. The other improves Tian's method by using low-pass image to predict which locations will be used to embed information. Xuan *et al.* [11] adopt histogram modification to prevent the overflow and underflow and embedded data in the coefficients in high frequency subbands after integer wavelet transformation.

In this paper, the proposed scheme considers the correlation between the current pixel and its neighboring pixels and exploits the difference expansion of prediction-errors to conceal the message data. The proposed scheme provides a high embedding capacity with little perceptual distortion. Additionally, the problem about overflow and underflow is prevented by preprocessing of histogram modification.

The rest of the paper is organized as follows. In section 2, a related lossless data hiding by multiple predictors is reviewed. Then, in section 3, the proposed embedding and recovery algorithm is described in detail. Experimental results and comparisons are presented in section 4. Finally, in the last section, we summarize our conclusions.

2. Related Work

Yip *et al.* [12] proposed a generalized lossless data hiding by multiple predictors in 2006. The cover image is scanned in the raster-scan order and users can choose a couple of different predictors to estimate the predictive values of the current pixel $P(x, y)$. Firstly, minimum predictive value ($\min_P$) and maximum predictive value ($\max_P$) are

obtained separately by using Eq. (1) and (2). The difference between $\hat{P}_1(x, y)$ and $\hat{P}_2(x, y)$ is calculated by using Eq. (3).

$$\min_P = \min(\hat{P}_1(x, y), \hat{P}_2(x, y)) \quad (1)$$

$$\max_P = \max(\hat{P}_1(x, y), \hat{P}_2(x, y)) \quad (2)$$

$$\text{Diff_}P = \max_P - \min_P - 1 \quad (3)$$

where $\hat{P}_1(x, y)$ and $\hat{P}_2(x, y)$ denote the predictive values which are estimated by predictor 1 and predictor 2, respectively.

Next, once the current pixel value satisfies the region, minimum, and maximum conditions as follows, it will be chosen as one of the candidate pixels. To prevent a great distortion, only some candidate pixels which satisfy the embedding condition will be used for embedding.

1) Region Condition: $2 < \text{Diff_}P < R$

2) Minimum Condition:

$$\min_P > \text{floor}(1.5 \times R) + 1 + B$$

3) Maximum Condition:

$$\max_P < 255 - \text{floor}(1.5 \times R) - 1 - B$$

4) Embedding Condition:

$$\min_P - \text{floor}(0.5 \times \text{Diff_}P) < P(x, y)$$

$$< \max_P + \text{floor}(0.5 \times \text{Diff_}P)$$

Here, R is a predefined threshold and B is a predefined value to prevent overflow and underflow.

Finally, the candidate pixels which satisfy the embedding condition are used to conceal the message data by performing bijective mirror mapping (BMM). The bijective mirror mapping selects $\min_P$ or $\max_P$ as mirror according to the secret bit. If the secret bit is "1", the $\min_P$ will be selected as mirror as shown in Fig. 1 (a). Otherwise, the $\max_P$ will be chosen as mirror to embed the secret bit "0" as shown in Fig. 1 (b). The $p'(x, y)$ represents a new pixel value. The variable m and n are the differences between $(p(x, y), \min_P)$ and $(p(x, y), \max_P)$, respectively. To ensure the reversibility, other candidate pixels are modified by performing bijective pixel value shifting (BPVS). If the pixel value is smaller than $\min_P$, it will be subtracted by $\text{Diff_}P + 1$ as shown in Fig. 2 (a). Otherwise, it will be added by $\text{Diff_}P + 1$ as shown in Fig. 2 (b).

In the decoding phase, the watermarked image is scanned by using inverse raster-scan

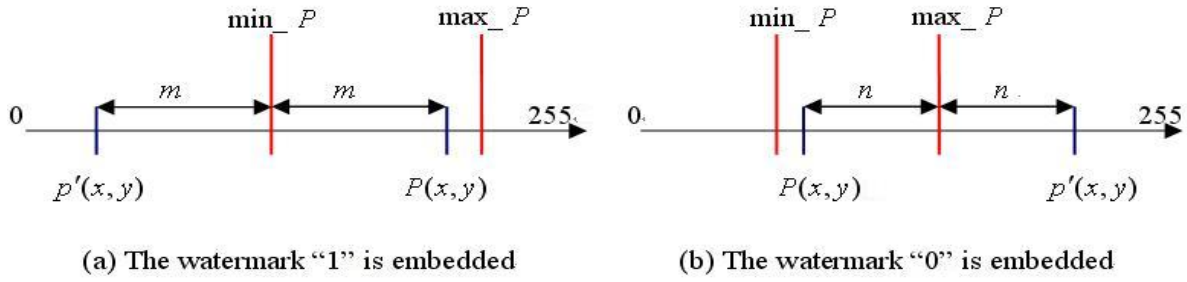


Fig. 1. The illustration of BMM

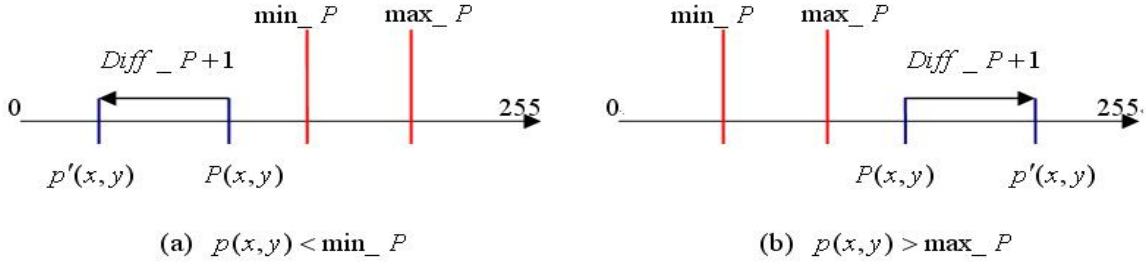


Fig. 2. The illustration of BPVS

order. The candidate pixels in the embedding procedure can be obtained by using region, maximum, and minimum conditions. The watermark can be extracted by comparing the difference between the watermarked candidate pixels which satisfy the extraction condition and their predicted values.

5) Extraction Condition:

$$\begin{aligned} \min_P - \text{floor}(1.5 \times \text{Diff_P}) - 1 &< S(x, y) \\ &< \max_P + \text{floor}(1.5 \times \text{Diff_P}) + 1 \end{aligned}$$

where $S(x, y)$ denotes the candidate pixel.

Once the watermarked candidate pixel is close to $\min_P$, the secret bit "1" is extracted. If the watermarked candidate pixel is close to $\max_P$, the secret bit "0" is extracted. Finally the embedded watermark is extracted. The watermarked image is recovered to original image by computing inverse-BMM and inverse-BPVS. Inverse-BMM restores the watermarked candidate pixels by choosing the closer predictive value as mirror and inverse-BPVS recovers other candidate pixels by added or subtracted $\text{Diff_P} + 1$. Then, the original image can be restored perfectly.

3. The Proposed Approach

In this section, the proposed scheme is described in detail. Let X denotes the cover

image of $w \times h$ pixels. The $p(x, y)$ is a pixel value of X , where $0 \leq x < w$ and $0 \leq y < h$. Furthermore, the watermark bit is denoted as W , where $W \in \{0, 1\}$. The embedding and restoring procedure are described in Section 3.1 and Section 3.2 respectively. In Section 3.3, the overflow and underflow problems are discussed.

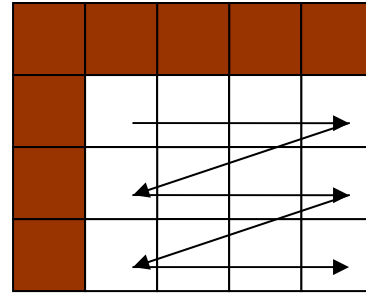


Fig. 3. The two-sided side match with raster-scan order

3.1 Embedding Algorithm

Step 1: Firstly, the two-sided side match prediction method is employed in the proposed scheme. The cover image is embedded in the raster-scan order except for the pixels of the first row and the first column as shown in Fig. 3. Eq. (4) is used to compute the predictive value $\hat{p}(x, y)$ of the current pixel $p(x, y)$

and then the prediction-error d is calculated using Eq. (5).

$$\hat{p}(x,y) = \lfloor (p(x-1,y) + p(x,y-1)) / 2 \rfloor \quad (4)$$

$$d = |\hat{p}(x,y) - p(x,y)| \quad (5)$$

Step 2: A threshold value T is predefined and is regarded as a secret key. In order to extract the message data without confusion, the following three cases are considered in the embedding procedure. The pixel value is modified according to the case which the prediction-error value belongs to.

Case 1: $\lceil T/2 \rceil \leq d < T$

Case 2: $T \leq d < T + \lfloor T/2 \rfloor$

Case 3: $d \geq T + \lfloor T/2 \rfloor$

Step 3: For a natural image, the pixel value and its predicted value are usually very similar so that the prediction-error is very small. This characteristic is applied to improve the embedding capacity. If the prediction-error of the current pixel matches Case 1, the difference expansion

$$p'(x,y) = \begin{cases} \hat{p}(x,y) + 2 \times d + W, & \text{if } p(x,y) > \hat{p}(x,y) \\ \hat{p}(x,y) - 2 \times d - W, & \text{if } p(x,y) < \hat{p}(x,y) \end{cases} \quad (6)$$

is used to embed the watermark W and produce a new pixel value $p'(x,y)$.

Step 4: In Case 2 and Case 3, the prediction-error is large and is not capable of embedding data. Eq. (7) is performed for the prediction-error values which match Case 2. If the prediction-error values match Case 3, a new pixel value $p'(x,y)$ is computed by Eq. (8). Finally, the watermarked image X_w is produced.

$$p'(x,y) = \begin{cases} p(x,y) - \lfloor T/2 \rfloor, & \text{if } p(x,y) > \hat{p}(x,y) \\ p(x,y) + \lfloor T/2 \rfloor, & \text{if } p(x,y) < \hat{p}(x,y) \end{cases} \quad (7)$$

$$p'(x,y) = \begin{cases} p(x,y) + \lceil T/2 \rceil, & \text{if } p(x,y) > \hat{p}(x,y) \\ p(x,y) - \lceil T/2 \rceil, & \text{if } p(x,y) < \hat{p}(x,y) \end{cases} \quad (8)$$

An illustration of the embedding procedure

30	24	40	58
38	30	33	46
40	37	42	44

(a) A cover image

is given as follows and Table 1. A grayscale data as shown in Fig. 4 (a) is assumed to be the cover image. The embedding data is 10. In this example, the threshold T is set to be 2.

For the pixel $p(2,2) = 30$, the two neighboring pixels have values 24 and 38. The predictive value and the prediction-error are computed as $\hat{p}(2,2) = \lfloor (24 + 38) / 2 \rfloor = 31$ and $d = |31 - 30| = 1$, respectively. Since $p(2,2)$ matches Case 1 and $\hat{p}(2,2) > p(2,2)$, the embedded result of $p(2,2)$ is computed as $p'(2,2) = 31 - 2 \times 1 - 1 = 28$. For the pixel $p(2,3) = 33$, the two neighboring pixels have values 40 and 30. The predictive value and the prediction-error are computed as $\hat{p}(2,3) = \lfloor (40 + 30) / 2 \rfloor = 35$ and $d = |35 - 33| = 2$, respectively. Since $p(2,3)$ matches Case 2 and $\hat{p}(2,3) > p(2,3)$, the new value of pixel $p(2,3)$ is computed as $p'(2,3) = 33 + 1 = 34$. For the pixel $p(2,4) = 46$, its neighboring pixels are 58 and 33. The predictive value and the prediction-error are computed as $\hat{p}(2,4) = \lfloor (58 + 33) / 2 \rfloor = 45$ and $d = |45 - 46| = 1$, respectively. Since $p(2,4)$ matches Case 1 and $\hat{p}(2,4) < p(2,4)$, the embedded result of $p(2,4)$ is computed as $p'(2,4) = 45 + 2 \times 1 + 0 = 47$.

For the pixel $p(3,2) = 37$, its neighboring pixels are 30 and 40. The predictive value and the prediction-error are computed as $\hat{p}(3,2) = \lfloor (30 + 40) / 2 \rfloor = 35$ and $d = |35 - 37| = 2$, respectively. Since $p(3,2)$ matches Case 2 and $\hat{p}(3,2) < p(3,2)$, the new value of pixel $p(3,2)$ is computed as $p'(3,2) = 37 - 1 = 36$. For the pixel $p(3,3) = 42$, the two neighboring pixels have values 33 and 37. The predictive value and the prediction-error are computed as $\hat{p}(3,3) = \lfloor (33 + 37) / 2 \rfloor = 35$ and $d = |35 - 42| = 7$, respectively. Since $p(3,3)$ matches Case 3 and

30	24	40	58
38	28	34	47
40	36	43	44

(b) A watermarked image

Fig. 4. An example of embedding procedure

$\hat{p}(3,3) < p(3,3)$, the new value of pixel $p(3,3)$ is computed as $p'(3,3) = 42 + 1 = 43$. For the pixel $p(3,4) = 44$, its neighboring pixels are 46 and 42. The predictive value and the prediction-error are computed as $\hat{p}(3,4) = \lfloor (46 + 42) / 2 \rfloor = 44$ and $d = |44 - 44| = 0$, respectively. If the prediction-error of the current pixel is smaller than $\lceil T/2 \rceil$, the pixel keeps unchanged. Thus the new value of pixel $p(3,4)$ is still 44 in this example. Finally, the watermarked image X_w is produced as shown in Fig. 4 (b).

Table 1. An example of embedding procedure

$p(x,y)$	$p(x-1,y)$	$p(x,y-1)$	$\hat{p}(x,y)$	d	W	$p'(x,y)$	Case
30	24	38	31	1	1	28	1
33	40	30	35	2	—	34	2
46	58	33	45	1	0	47	1
37	30	40	35	2	—	36	2
42	33	37	35	7	—	43	3
44	46	42	44	0	—	44	—

3.2 Extraction and Restoration Algorithm

Step 1: The embedded data is extracted from X_w in the raster-scan order except for the first row and the first column. The predicted value $\hat{p}'(x,y)$ and prediction-error d' of the current pixel value are computed by using Eq. (4) and Eq. (5), separately.

Step 2: The same threshold T is used to extract the watermark and restore the cover image. The following three cases are considered in the extraction and restoration procedure. The prediction-error of each pixel value is used to determine which case the pixel value belongs to.

Case 1: $T \leq d' < 2T$

Case 2: $\lceil T/2 \rceil \leq d' < T$

Case 3: $d' \geq 2T$

Step 3: For those prediction-error values which match Case 1, the embedded secret bit can be inferred from LSB of d' .

Calculate the secret bit W as follows:

$$W = d' - 2 \times \lfloor d' / 2 \rfloor. \quad (9)$$

And the pixels are restored to its original value $p^*(x,y)$ after calculating Eq. (10)

as follows. Here, $p^*(x,y)$ is the equal of $p(x,y)$.

$$p^*(x,y) = \begin{cases} \hat{p}(x,y) + \lfloor d/2 \rfloor & \text{if } p(x,y) > \hat{p}(x,y) \\ \hat{p}(x,y) - \lfloor d/2 \rfloor & \text{if } p(x,y) < \hat{p}(x,y) \end{cases} \quad (10)$$

Step 4: For the prediction-error values match Case 2, Eq. (11) as follows are used for restoration.

$$p^*(x,y) = \begin{cases} \hat{p}(x,y) + \lceil T/2 \rceil & \text{if } p(x,y) > \hat{p}(x,y) \\ \hat{p}(x,y) - \lceil T/2 \rceil & \text{if } p(x,y) < \hat{p}(x,y) \end{cases} \quad (11)$$

And the prediction-error values in the Case 3 are recovered by using Eq. (12) as follows.

$$p^*(x,y) = \begin{cases} \hat{p}(x,y) - \lceil T/2 \rceil & \text{if } p(x,y) > \hat{p}(x,y) \\ \hat{p}(x,y) + \lceil T/2 \rceil & \text{if } p(x,y) < \hat{p}(x,y) \end{cases} \quad (12)$$

Finally, the scheme can extract the embedded message data and restore the original pixels correctly.

An example of extraction and restoration procedure is illustrated in Table 2. A watermarked image as shown in Fig. 4 (b) is scanned in raster-scan order.

Table 2. An example of extraction and restoration procedure

$p'(x,y)$	$p'(x-1,y)$	$p'(x,y-1)$	$\hat{p}'(x,y)$	d'	W	$p^*(x,y)$	Case
28	24	38	31	3	1	30	1
34	40	30	35	1	—	33	2
47	58	33	45	2	0	46	1
36	30	40	35	1	—	37	2
43	33	37	35	8	—	42	3
44	46	42	44	0	—	44	—

3.3 Overflow and Underflow

For some images, it has possibility that some extreme pixels of the watermarked image exceed low bound (0) or upper bound (255) of eight-bit grayscale image after embedding procedure. This problem is called underflow/overflow. In the proposed approach, we modify the histogram modification scheme proposed by Xuan *et al.* [11] to prevent overflow

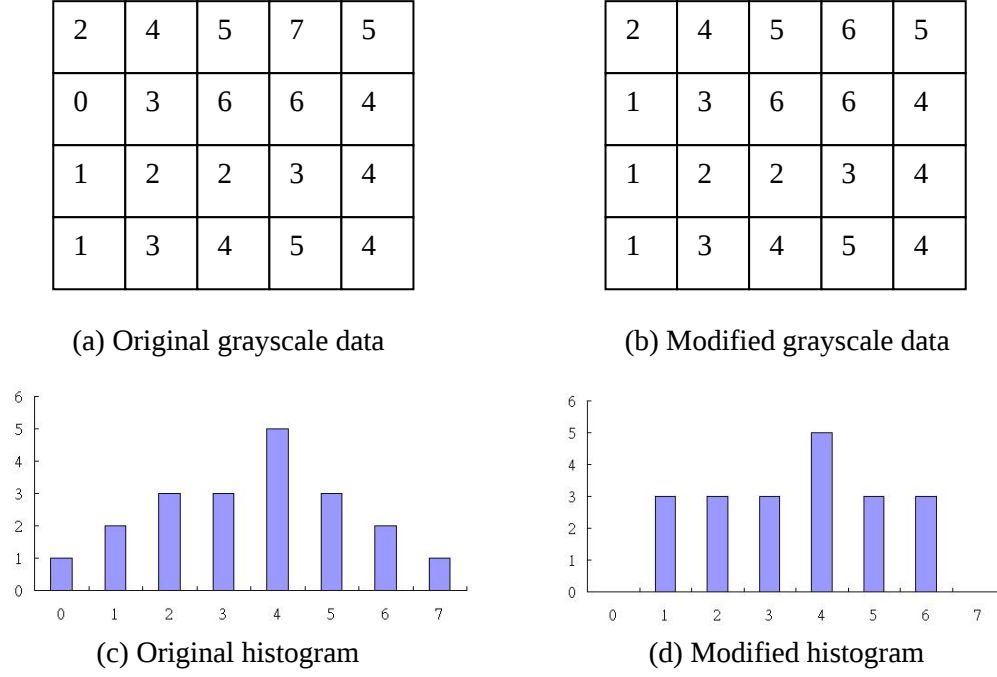


Fig. 5. The illustration of histogram modification ($k = 1$)

L	k	<i>left – side scan sequence</i>	<i>right – side scan sequence</i>
-----	-----	----------------------------------	-----------------------------------

Fig. 6. Overhead information form

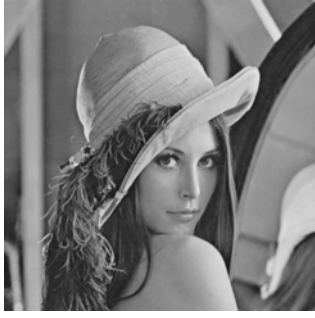
and underflow. The extreme pixel values which may be underflow/overflow are modified by narrowing down the histogram of cover image. In the histogram, $2k$ grayscale values of both sides are added or subtracted by k and therefore the histogram is narrowed down. An illustration of the histogram modification is described as follows. Assume that a three-bit grayscale data with size 4×5 as shown in Fig. 5 is used to be a simple example. In this example, k is set to be 1. Furthermore, the overhead information form consists of four components as shown in Fig. 6. The L with fixed size contains the total length of overhead information. Then k with fixed size is appended sequentially. Scan sequences indicate the locations of the modified pixel values. For example, the original grayscale data is scanned in raster-scan order. Once a grayscale value “0” (“7”) is encountered, a bit “1” is appended to the scan sequence and the value is changed to “1” (“6”). If a grayscale value “1” (“6”) is encountered, a bit “0” is appended to the scan sequence and the value keeps unchanged. Eventually, the left-side scan sequence got “100” and the right-side scan sequence got “100” after scanning.

In the decoding procedure, the total length L is extracted firstly. Then the k is obtained which denotes the number of scan sequences. The size of each scan sequence can be obtained by using modified histogram. The grayscale values can be restored according to scan sequences. Finally, the original histogram can be recovered completely.

4. Experimental Results

In our experiments, three 512×512 standard grayscale images are used to test the performance of our schemes. The test images, Lenna, F16, and Barbara, are shown in Fig. 7. The peak signal to noise ratio (PSNR) is employed as a measure of the stego image quality and the payload capacity is evaluated by using the bit per pixel (bpp). In order to evaluate the performance of the proposed scheme with different predictor, the following four predictors [11] and Side Match predictor are used in the experiments. The locations of four neighboring pixels are shown in Fig. 8.

- a) Horizontal Predictor: $\hat{p}(x, y) = p(x-1, y)$
- b) Vertical Predictor: $\hat{p}(x, y) = p(x, y-1)$



(a) Lenna



(b) F16



(c) Barbara

Fig. 7. Three test images

$p(x-1,y-1)$	$p(x-1,y)$	$p(x-1,y+1)$
$p(x,y-1)$	$p(x,y)$	

Fig. 8. Neighboring pixel locations

- c) Causal Weighted Average:

$$\hat{p}(x,y) = (2 \times p(x-1,y) + 2 \times p(x,y-1) + p(x-1,y-1) + p(x-1,y+1)) / 6$$

- d) Causal SVF:

$$Tgt(x,y) = (p(x-1,y) + p(x,y-1) + p(x-1,y-1) + p(x-1,y+1)) / 4$$

$$\hat{p}(x,y) = \frac{\sum_{n=1}^1 p(x-1,y-n) \times w(x-1,y-n) + p(x,y-1) \times w(x,y-1)}{\sum_{n=1}^1 w(x-1,y-n) + w(x,y-1)}$$

$$w(x+i,y+j) = \exp(-abs(p(x+i,y+j) - Tgt(x,y) \times k)),$$

where k is the controlling factor.

The visual quality of embedded images Lenna, F16, and Barbara with $T = 4$ and $T = 6$ are shown in Fig. 9, Fig. 10 and Fig. 11, respectively. The embedding capacity and the PSNR values of each image with different thresholds are given in Table 3, Table 4, and Table 5. It can be observed that the proposed scheme has the best performance for Lenna with $T = 4$ by using Causal SVF. It is able to hide 79,954 bits (0.305 bpp) with high visual quality of 42.06 dB. As for F16, the proposed scheme with $T = 2$ using Side Match predictor is able to hide 74,973 bits (0.286 bpp) with high image quality of 47.24 dB. In test image Barbara, the proposed scheme with $T = 4$ using Vertical predictor is able to hide 63,439 bits (0.242 bpp) with high image quality of 41.85 dB. The experimental results show that the visual quality

(PSNR) can be improved by using better predictors. The more accuracy the predictor make, the higher PSNR it has.

The performance of the proposed scheme is compared with Yip *et al.*'s scheme [11] in terms of PSNR and payload capacity. Table 6 presents the comparison results. It can be seen that the proposed scheme performs much better than Yip *et al.*'s method in terms of embedding capacity and image quality evidently. Experiment results also show that the visual quality of stego image can be improved by using better predictors.

5. Conclusions

In this paper, a high capacity and low distortion reversible data hiding technique using prediction-error expansion is proposed. The correlation between the pixel and its neighboring pixels is considered to perform difference expansion and the location map is not needed to ensure reversibility. The experimental results show that our scheme provides high embedding capacity with low distortion and is suitable for different predictors. Furthermore, the proposed scheme is secure. Since the threshold T is needed in the embedding and extraction procedure. The watermark can not be extracted without the threshold T .

Table 3. Experimental results of embedded image Lenna with different predictors

Image	T	The Proposed Scheme									
		Side Match		Horizontal Predictor		Vertical Predictor		Causal Weighted Average		Causal SVF	
		Payload (bpp)	PSNR	Payload (bpp)	PSNR	Payload (bpp)	PSNR	Payload (bpp)	PSNR	Payload (bpp)	PSNR
Lenna	2	0.220	47.31	0.177	47.46	0.212	47.35	0.230	47.30	0.231	47.30
	3	0.176	43.84	0.151	43.49	0.172	43.81	0.180	43.95	0.180	43.97
	4	0.304	41.95	0.272	41.79	0.302	41.90	0.304	42.05	0.305	42.06
	5	0.214	40.91	0.213	40.31	0.222	40.77	0.207	41.10	0.207	41.13
	6	0.271	39.68	0.280	39.05	0.283	39.51	0.261	39.89	0.260	39.93

Table 4. Experimental results of embedded image F16 with different predictors

Image	T	The Proposed Scheme									
		Side Match		Horizontal Predictor		Vertical Predictor		Causal Weighted Average		Causal SVF	
		Payload (bpp)	PSNR	Payload (bpp)	PSNR	Payload (bpp)	PSNR	Payload (bpp)	PSNR	Payload (bpp)	PSNR
F16	2	0.286	47.24	0.257	47.31	0.249	47.38	0.280	47.29	0.271	47.31
	3	0.160	44.73	0.160	44.41	0.152	44.42	0.154	44.75	0.153	44.66
	4	0.251	43.03	0.263	42.64	0.251	42.70	0.243	43.08	0.244	42.99
	5	0.145	42.30	0.166	41.71	0.161	41.68	0.142	42.27	0.146	42.11
	6	0.181	41.22	0.207	40.62	0.202	40.59	0.178	41.19	0.183	41.04

Table 5. Experimental results of embedded image Barbara with different predictors

Image	T	The Proposed Scheme									
		Side Match		Horizontal Predictor		Vertical Predictor		Causal Weighted Average		Causal SVF	
		Payload (bpp)	PSNR	Payload (bpp)	PSNR	Payload (bpp)	PSNR	Payload (bpp)	PSNR	Payload (bpp)	PSNR
Barbara	2	0.160	47.53	0.122	47.65	0.160	47.52	0.161	47.53	0.159	47.54
	3	0.132	43.32	0.110	43.03	0.136	43.32	0.133	43.33	0.133	43.33
	4	0.234	41.90	0.203	41.79	0.242	41.85	0.231	41.95	0.232	41.93
	5	0.175	40.12	0.167	39.70	0.182	40.14	0.167	40.19	0.169	40.18
	6	0.226	39.18	0.224	38.73	0.236	39.13	0.216	39.27	0.218	39.25

Table 6. Comparisons between the proposed method and Yip *et al.*'s scheme

Images	Yip <i>et al.</i> 's Scheme				The Proposed Scheme	
	Horizontal Predictor & Vertical Predictor		Causal Weighted Average & Causal SVF		Side Match	
	Payload (bpp)	PSNR	Payload (bpp)	PSNR	Payload (bpp)	PSNR
Lenna	0.194	36.29	0.199	38.76	0.220	47.31
Barbara	0.143	37.21	0.152	39.79	0.160	47.53
F16	0.169	36.50	0.167	39.70	0.286	47.24



(a) $T = 4$



(b) $T = 6$

Fig. 9. Embedded image Lenna using the proposed scheme with side match



(a) $T = 4$



(b) $T = 6$

Fig. 10. Embedded image F16 using the proposed scheme with side match



(a) $T = 4$



(b) $T = 6$

Fig. 11. Embedded image Barbara using the proposed scheme with side match

References

- [1] Alattar, A. M., "Reversible Watermark Using the Difference Expansion of a Generalized Integer Transform," *IEEE Trans. Image Processing*, Vol. 13, no. 8, pp. 1147-1156, Aug. 2004.
- [2] Celik, M. U., Sharma, G., Tekalp, A. M. and Saber, E., "Lossless Generalized-LSB Data Embedding," *IEEE Trans. Image Processing*, Vol. 14, no. 2, pp. 253-265, Feb. 2005.
- [3] Fridrich, J., Goljan, M. and Du, R., "Invertible Authentication," *Proc. SPIE, Security and Watermarking of Multimedia Contents III*, Vol. 3971, pp. 197-208, Jan. 2001.
- [4] Fridrich, J., Goljan, M. and Du, R., "Lossless Data Embedding-New Paradigm in Digital Watermarking," *EURASIP J. Appl. Signal Processing*, Vol. 2002, no. 2, pp. 185-196, Feb. 2002.
- [5] Honsinger, C. W., Jones, P., Rabbani, M. and Stoffel, J. C., "Lossless Recovery of an Original Image Containing Embedded Data," *US Patent Application*, 6 278 791, Aug. 2001.
- [6] Kamstra, L. and Heijmans, Henk J. A. M., "Reversible Data Embedding Into Images Using Wavelet Techniques and Sorting," *IEEE Trans. Image Processing*, Vol. 14, no. 12, pp. 2082-2090, Dec. 2005.
- [7] Ni, Z., Shi, Y. Q., Ansari, N. and Su, W., "Reversible Data Hiding," *IEEE Trans. Circuits Syst. Video Technol.*, Vol. 16, no. 3, pp. 354-362, Mar. 2006.
- [8] Thodi, D. M. and Rodrigues, J. J., "Prediction-Error Based Reversible Watermarking," *Proc. IEEE Int. Conf. Image Processing*, pp. 1549-1552, 2004.
- [9] Tian, J., "Reversible Data Embedding Using a Difference Expansion," *IEEE Trans. Circuits Syst. and Video Technol.*, Vol. 13, no. 8, pp. 890-896, Aug. 2003.
- [10] Vleeschouwer, C. De., Delaigle, J. F. and Macq, B., "Circular Interpretation of Bijective Transformations in Lossless Watermarking for Media Asset Management," *IEEE Trans. Multimedia*, Vol. 5, no. 1, pp. 97-105, Mar. 2003.
- [11] Xuan, G., Yang, C., Zhen, Y., Shi, Yun Q. and Ni, Z., "Reversible Data Hiding Based on Wavelet Spread Spectrum," *IEEE Workshop Multimedia Signal Processing*, pp. 211-214, 2004.
- [12] Yip, S. K., Au, O. C., Wong, H. M. and Ho, C. W., "Generalized Lossless Data Hiding By Multiple Predictors," *ISCAS 2006*, pp. 21-24, May 2006.

植基於預測誤差擴展之可逆式資訊隱藏技術

摘要

面對迅速普及的數位媒體與網路應用，保護著作人的權益已然成為現今數位科技發展的重要議題，而受到廣泛討論之數位浮水印技術正是這類議題的解決方案之一。然而對於衛星、醫學影像等特殊領域的應用，由於其高精確需求和影像取得困難的關係，所以這類使用者不希望浮水印影像出現任何失真的情形。可逆式資訊隱藏技術不但可以讓使用者在浮水印影像中取出秘密資訊，而且無失真地將浮水印影像還原成原始影像。此可逆的特性提供此類特殊領域一個最佳的解決方案。本篇論文提出一種植基於預測誤差擴展之可逆式資訊隱藏技術。首先任意選擇一種預測器作為運算像素之預測值之用，接著擴展預測值和原像素之間的誤差值，完成將秘密資訊嵌入目標影像的目的。實驗證明本篇方法可提供相當好的可嵌入容量，並且不會造成嚴重失真。

關鍵字：可逆式資訊隱藏技術；預測；差分擴展