Adaptive Predictor Structure Based Interpolation For Reversible Data Hiding

CONFERENCE PAPER · OCTOBER 2014

DOI: 10.13140/2.1.1733.3767

Sunil Jaiswal

The Hong Kong University of Science and T…

Oscar C. Au

Vinit Jakhetiya

Nanyang Technological University

Yuanfang Guo

Chinese Academy of Sciences

Adaptive Predictor Structure Based Interpolation For Reversible Data Hiding

Sunil Prasad Jaiswal, Oscar Au, Vinit Jakhetiya, Andy Yuanfang Guo, and Anil K. Tiwari*

The Hong Kong University of Science and Technology, Hong Kong
* Indian Institute of Technology, Rajasthan, India
Email:{spjaiswal,eeau,vjakhetiya,eeandyyguo}@ust.hk,*akt@iitj.ac.in

Abstract. In this paper, we present an additive prediction error expansion (PEE) based reversible data hiding scheme that gives overall low distortion and relatively high embedding capacity. Recently reported interpolation based PEE method uses fixed order predictor that fails to exploit the correlation between the neighborhood pixels and the unknown pixel (to be interpolated). We observed that embedding capacity and distortion of PEE based algorithm depends on the prediction accuracy of the predictor. In view of this observation, we propose an interpolation based method that predicts pixels using predictors of different structure and order. Moreover, we use only original pixels for interpolation. Experimental results demonstrate that the proposed algorithm outperforms the state-of-the-art algorithms both in terms of embedding capacity and Peak Signal to Noise Ratio.

Keywords: Reversible image watermarking, predictor order, embedding capacity, interpolation.

1 Introduction

Reversible watermarking receives much popularity in recent years because of copyright protection of data. Reversible image watermarking means embedding a specific information into an image in such a way that the original image can be recovered at the decoder. Reversible watermarking is also useful in other applications such as image/video coding [1].

Reversible image watermarking (RIW) can be divided into two categories [2]: histogram shifting methods based data hiding [3]-[5], and difference expansion (DE) based reversible data hiding (RDH) methods [6]-[8]. Algorithms based on DE method have achieved better embedding capacity with low computation cost as compared to other methods. Algorithms in [9]-[11] have extended DE based methods by prediction error expansion (PEE) method i.e. by embedding watermarking data into prediction error sample instead of difference between paired pixels as done in DE schemes and leads to achieve improved embedding capacity. Luo et. al [11] proposed an interpolation based PEE (IPEE) method and embedded data into interpolation error instead of prediction error.

1. Though the interpolation technique used in [11] is adaptive to edges but uses a fixed 4^{th} order predictor and hence fails to exploit correlation between the neighborhood pixels and the unknown pixel.
2. Moreover, the interpolation algorithm used in [11] is a two pass process. In first pass, $(1/4)^{th}$ of total missing samples are interpolated using the original pixels and in second pass, rest of the missing samples are interpolated using both original pixels and interpolated pixels. Thus use of the interpolated pixels in the second pass causes error propagation from first pass to second pass and hence decreases the prediction accuracy.

In view of the above mentioned problems, main contributions of proposed algorithm are as follows:

1. We propose a prediction method that uses only original pixels for interpolation in both the passes and thus no error propagation occurs from first pass to second pass which increases the prediction accuracy.
2. Also, we propose a higher order predictor as against of 4^{th} order proposed by IPEE [11].

The rest of paper is organized as follows. In Section 2, additive PEE based data hiding is explained. Existing interpolation based PEE algorithm is discussed in section 3 while proposed method is given in section 4. Section 5 includes simulation results and concluding remarks are given in section 6.

2 Additive Prediction Error Expansion

Watermark data (0 or 1) is embedded by using the prediction error expansion (PEE) method [9]-[11] as discussed below:

2.1 Encoder

Suppose a prediction method is applied on the original pixel $I(n)$ and prediction value $P(n)$ is obtained. For prediction use causal pixels as decoder do not have information of non-causal pixel. Then prediction error is obtained as,

$$e(n) = I(n) - P(n). \tag{1}$$

Thus after getting the prediction error, additive PEE-based embedding is performed as given in (2) .

$$e_w(n) = \begin{cases} e(n) - b.sign(e(n)).Q & -Q \leq e(n) < Q \\ e(n) - Q & e(n) < -Q \\ e(n) + Q & e(n) >= Q \end{cases} \tag{2}$$

Here b is the to-be-embedded watermark bit (0 or 1), $sign(e(n))$ implies +1 if $e(n)$ is positive and (-1) if negative and Q is the capacity parameters. In (2), embedding of bit (b) is done in prediction error samples $(e(n))$ in the range of

$[-Q,\ Q)$ only. Other value of $e(n)$ gets shifted so as to avoid overlapping and no watermark information is embedded. Thus after additive PEE, watermarked pixels can be given by,

$$I_w(n) = P(n) + e_w(n) \tag{3}$$

Additive PEE based embedding algorithms involve error expansion and shifting operations which may cause overflow/underflow problems i.e. some pixels may go outside the range of $[0, 255]$ for an 8 bit image. This may result ambiguity and thus perfect reconstruction at decoder will not be possible. To deal such problem a location map is generated, at encoder, to keep record of location of pixels and embedding operation is not done on such pixels to avoid ambiguity. These location map needs to be send to decoder losslessly to ensure reversibility.

2.2 Decoder

In a reverse order, we can restore the original image ($\mathbf{I}$) from the watermarked image ($\mathbf{I_w}$).

At first, check whether the current pixel location is recorded in location map. If recorded, then the given pixel is the original pixel itself and thus no watermark bit is extracted. Else, the decoding process can be described as follows:

1. First, prediction value $P(n)$ is estimated (as it is obtained at encoder) and thus $e_w(n)$ can be calculated via, $e_w(n) = I_w(n) - P(n)$.
2. Once we get $e_w(n)$, the prediction error pixel ($e(n)$) and the hidden bit (b) can be obtained using (4) given below.

$$e(n) = \begin{cases} e_w(n) \text{ and b} = 0 \\ \quad \text{if } -Q \le e_w(n) < Q \\ \\ e_w(n) - sign(e_w(n)) \times -Q \text{ and b=1} \\ \quad \text{if } -2 \times Q \le e_w(n) < -Q \\ \quad \text{or } Q \le e_w(n) < 2 \times Q \\ \\ e_w(n) - sign(e_w(n)) \times Q \\ \quad \text{otherwise} \end{cases} \tag{4}$$

3. Then original pixel is estimated by, $I(n) = P(n) + e(n)$.

Thus, we can recover both the original image ($\mathbf{I}$) and the watermarked data (b) at the decoder.

The relationship between $e(n)$ and $e_w(n)$ is discussed in method [10] and is given as follows: When an original pixel undergoes watermarking processes by (2), the distortion caused be the process is $I(n) - I_w(n) = (P(n) + e(n)) - (P(n) + e_w(n)) = e(n) - e_w(n) \le Q$. Thus, distortion is small and changes in the watermarked image is imperceptible for smaller values of Q. From (2), we observe that bits (b) are embedded into residue samples ($e(n)$) having values in the range of [-Q,Q]. Thus, better the accuracy of the predictor, more will be the embedding capacity.

3 Existing Interpolation based PEE algorithm

In [11], original image **I** of dimension $W \times H$ is down sampled by a factor of two; row wise and column wise. Due to direct down sampling operation, a low resolution (**LR**) version image of dimension $W/2 \times H/2$ is obtained. The low resolution image (**LR**) is then interpolated to High Resolution Image (**P**) using a two pass algorithm. The odd-odd positioned pixels denoted by black dots in Fig.1 are copied from LR image as shown in (5).

$$P(2i-1, 2j-1) = LR(i,j) \quad \forall\ (i,j) \in LR \tag{5}$$

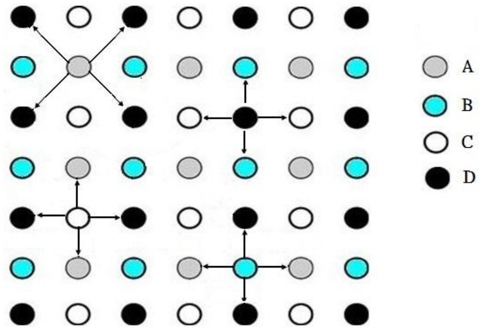

Fig. 1. Black dots represent original pixel of LR image. Rest of the pixel are to be interpolated.

Based on the co-ordinate position of the pixels to be interpolated, we label the pixels of **P** by four symbols; $A(n)$, $B(n)$, $C(n)$, $D(n)$. **A**, **B**, **C** are symbols for the unknown pixels (to be interpolated) located at even-even positions, even-odd positions, odd-even positions respectively while symbols D are the original pixels located at odd-odd positions. Without loss of generality, consider first pixel of the image to be at odd-odd position i.e, (1,1). By using these symbols, a part of given image is shown in Fig.1. The working procedure of [[11]] can be divided into two phase as follows:

3.1 First phase: Watermarking process of $(3/4)^{th}$ of total pixels

1. In first pass, all the even-even positioned pixels (type **A**) denoted by gray dots in Fig.1 are predicted using its 4 diagonal neighboring pixels i.e. the predictor order is 4. Thus in first pass, it uses only original pixels for prediction.
2. In second pass, all the pixels denoted by **B** and **C** (denoted by blue and white dots respectively) in Fig.1 are predicted using four closest neighboring pixels. It involves two original pixels (black dots) and two interpolated pixels (gray dots) for prediction.

It is clear that prediction accuracy decreases in second pass due to involvement of interpolated pixels. Thus two-pass method can propagate errors from first pass to second pass. After predicting $(3/4)^{th}$ of the total pixels in $\mathbf{P}$ and finding its corresponding $e(n)$, additive PEE method is used to obtain the corresponding watermarked pixels.

3.2 Second Phase: Watermarking process of remaining pixels

By the above phase, only $(3/4)^{th}$ pixels of $\mathbf{I}$ undergo watermarking process. The watermarking process for remaining original $(1/4)^{th}$ pixels (type $\mathbf{D}$) can be described as follows:

The odd-odd positioned pixels ($(1/4)^{th}$ pixel of $\mathbf{I}$) can be predicted using watermarked samples along two orthogonal directions as shown in Fig.1. Thus after predicting each pixel of odd-odd position, find its corresponding $e(n)$, and apply additive PEE method to get corresponding watermarked pixels. Thus final watermarked image ($\mathbf{I_w}$) is obtained and then sent to decoder.

At decoder, in a reverse order of watermarking algorithm (performed at encoder), we can recover the original image ($\mathbf{I}$) and the watermark data (b) easily [11].

4 Proposed Algorithm

For PEE based embedding algorithm, embedding capacity and distortion of an algorithm depends on the prediction accuracy of the predictor. Thus the aim of proposed algorithm is to increase the prediction accuracy so as to achieve higher embedding capacity.

4.1 First phase : Watermarking process of $(3/4)^{th}$ pixels of I

In the first phase, we interpolate the LR image to a high resolution image ($\mathbf{P}$) using proposed prediction algorithm as described below.

Prediction of even-even positioned pixels (A type) In first pass, we predict pixels denoted by $\mathbf{A}$. However the proposed predictor structure is different from IPEE [[11]]. Proposed predictor structure consist of 6 original pixels and is shown in Fig. 2(a). We define the linear predictor for $A(n)$ as $\hat{A}(n)$ and is given in (6).

$$\hat{A}(n) = \boldsymbol{a}^T \boldsymbol{\phi}_i = \sum_{k=1}^{6} a_k \phi_i(k) \tag{6}$$

Here $\boldsymbol{a} = [a_1, a_2, a_3, a_4, a_5, a_6]^T$ are prediction coefficients and $\boldsymbol{\phi}_i = [D(n-4), C(n-3), D(n-2), B(n-1), D(n+2), D(n+4)]$ is a vector consisting of neighboring pixels of $A(n)$. Note that the prediction is based on the four odd-odd positioned pixels ($\mathbf{D}$ type) and two causal pixels. Hence, decoder can obtain the same prediction without any ambiguity.

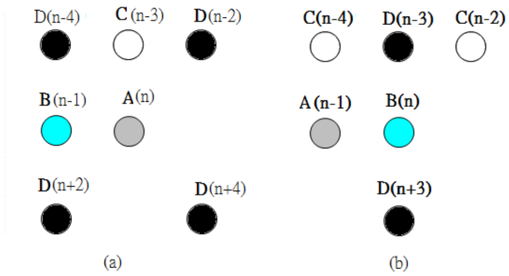

Fig. 2. (a) Predictor structure of A(n) consists of 4 D-type pixels (black dots) and 2 causal pixels, (b) Predictor structure of B(n) consists of 2 D-type pixels (black dots) and 3 causal pixels.

For finding prediction coefficients ($\boldsymbol{a}$), we propose to adopt method given in [[13]]. However, predictor structure of [[13]] consists of only causal pixels whereas proposed predictor structure consists of both causal and non-causal pixels and is described as below:

We first find gradients at the unknown pixel ($A(n)$) in horizontal direction (D_H) and in vertical direction (D_V) and then find relative variation as,

$$S = D_H - D_V \tag{7}$$

where $D_V = |D(n-4) - B(n-1)| + |B(n-1) - D(n+2)| + |D(n-2) - D(n+4)|$ and $D_H = |D(n-4) - C(n-3)| + |C(n-3) - D(n-2)| + |D(n+2) - D(n+4)|$.

After finding gradients, slope S is classified into 7 bins and its description is given Table 1. The bin boundaries given in Table 1 are chosen in such a way that there are approximately same number of pixels in each bin. By doing so, we are doing equal treatment to pixels belonging to various bins [[13]].

1. We classify the pixels of type A ($(1/4)^{th}$ pixels of $\mathbf{I}$) in 7 bins as per the bin boundaries given in Table 1. Each bin consists of pixels which have similar edge structure (Table 1) and thus Least Square (LS) based prediction coefficient is most suitable for precise prediction [[13]]-[[15]].
2. For each bin, LS based prediction coefficients ($\boldsymbol{a}$) is estimated by minimizing the following cost function.

$$J(\boldsymbol{a}) = \sum_{n=1}^{Z} (A(n) - \boldsymbol{a}^T \boldsymbol{\phi}_i)^2 \tag{8}$$

   where, Z is the total number of pixels in corresponding bin. Hence, we obtain optimal the LS based weights $\boldsymbol{a}$ by differentiating (9) with respect to the parameters i.e.

$$\boldsymbol{a}^* = \min_{a} \quad J(\boldsymbol{a}) \tag{9}$$

Table 1. Classification of Slope (S) into bins.

Slope S	Bin	Description
$S > 40$	Bin 1	Sharp edge along vertical direction
$40 \geq S > 20$	Bin 2	Edge along vertical direction
$20 \geq S > 8$	Bin 3	Weak edge along vertical direction
$8 \geq S > -8$	Bin 4	No edge
$-8 \geq S > -20$	Bin 5	Weak Edge along horizontal direction
$-20 \geq S > -40$	Bin 6	edge along horizontal direction
$S \geq -40$	Bin 7	Sharp edge along horizontal direction

It is well-known that the LS optimization has a closed-form solution and is given by

$$\boldsymbol{a}^* = (\boldsymbol{\Phi}^T\boldsymbol{\Phi})^{-1}(\boldsymbol{\Phi}^T\boldsymbol{y}) \tag{10}$$

Here $\boldsymbol{\Phi}$=[$\boldsymbol{\phi}_1$ $\boldsymbol{\phi}_2$ $\boldsymbol{\phi}_k$ . . . $\boldsymbol{\phi}_Z$]^T and $\boldsymbol{y}$=[$A(1)$ $A(2)$ $A(3)$ . . . $A(Z)$]^T. Hence for each bin, we find a 6^{th} order prediction coefficient ($\boldsymbol{a}$) and thus predict all the pixels of that bin using the same prediction coefficients.

3. Repeat step 2 to find prediction coefficients for each bin.

Thus for 7 bins, seven set of prediction coefficients of 6^{th} order are estimated. In this way, all pixels of type **A** are predicted efficiently.

Prediction of even-odd positioned pixels (B type) In this pass, we predict pixels denoted by **B**. The proposed algorithm again uses original pixels for prediction and the predictor structure for prediction is shown in Fig.2(b). Proposed predictor structure consist of 5 pixels: two odd-odd positioned pixels and three causal pixels. Hence, decoder can obtain the same prediction and thus no ambiguity occurs.

Here again relative variation S is estimated and is given in (7) with $D_H = |C(n{-}4){-}D(n{-}3)|{+}|C(n{-}2){-}D(n{-}3)|{+}|A(n{-}1){-}((D(n{-}3){+}(D(n{+}3))/2|$ and
$D_V = |C(n-4) - A(n-1)| + |D(n-3) - D(n+3)| + |C(n-2) - D(n+3)|$.

As previously, we classify the pixels into 7 bins by classifying S with the bin boundaries given in Table 1. For each bin, we estimate LS based prediction coefficient of 5^{th} order and predict the pixels denoted by **B** as given below.

$$\hat{B}(n) = \boldsymbol{a}^T\boldsymbol{\phi}_i = \sum_{k=1}^{5} a_k\phi_i(k) \tag{11}$$

Here $\boldsymbol{a} = [a_1, a_2, a_3, a_4, a_5]^T$ is a vector of prediction coefficients and $\boldsymbol{\phi}_i = [C(n-4), D(n-3), C(n-2), A(n-1), D(n+3)]$ is neighboring pixels of $B(n)$. Hence in order to predict all the pixels denoted by **B**, we need seven set of prediction coefficients of order 5.

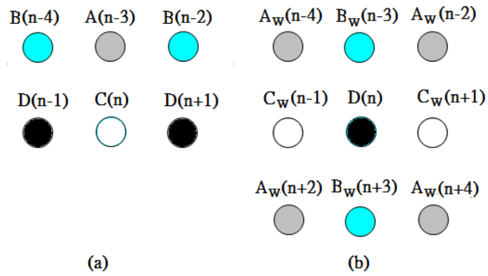

Fig. 3. (a) Predictor structure of C(n): 2 D-type pixels (black dots) and 3 causal pixels, (b) Predictor structure of D(n) consist of watermarked pixels (obtained from First phase)

Prediction of odd-even positioned pixels (C type) Similar to previous pass, the predictor structure consist of 5 original pixels and it is shown in Fig. 3(a). In this case, relative variation S given in (7) is estimated using $D_H = |B(n-4)-A(n-3)|+|B(n-2)-A(n-3)|+|D(n-1)-D(n+1)|$ and $D_V = |B(n-4)-D(n-1)|+|B(n-2)-D(n+1)|+|A(-3)-(D(n-1)+D(n+1))/2|$.

Prediction of **C** typed pixels is done exactly the same way as it is done for previous cases. Predictor structure in this pass is different and is shown in Fig. 3(a). Again, in order to predict all the pixels denoted by **C**, we need seven set of LS based prediction coefficients of order 5.

Unlike [[11]], the proposed algorithm interpolates the LR image ie, predicts pixels of even-even, even-odd and odd-even positioned pixels by using original pixels only. Thus, no error propagation occurs from one pass to other pass during prediction. After predicting $(3/4)^{th}$ of the total pixels of **I** and finding corresponding $e(n)$, we can apply additive PEE method to get corresponding watermarked pixels.

4.2 Second phase : Watermarking process of remaining $(1/4)^{th}$ pixels - type D pixels

In this phase, pixels denoted by **D** are predicted and then undergoes watermarking process. Prediction is done by using only watermarked samples as decoder does not have the information of neighboring original pixels. The proposed predictor structure is shown in Fig. 3(b).

Similar to the previous case, we classify pixels in 7 bins and find predictors of order 4 for each of the classes. However, relative variation S, given in (7), is estimated using $D_H = |A_w(n-4) - B_w(n-3)| + |C_w(n-1) - C_w(n+1)| + |A_w(n+2) - A_w(n+4)|$ and $D_V = |A_w(n-4) - A_w(n+2)| + |B_w(n-3) - B_w(n+3)| + |A_w(n-2) - A_w(n+4)|$. Prediction of pixels denoted by D is given by $\hat{D}(n)$ as below

$$\hat{D}(n) = \boldsymbol{a}^T \boldsymbol{\phi}_i = \sum_{k=1}^{4} a_k \phi_i(k) \tag{12}$$

Here $\boldsymbol{a} = [a_1, a_2, a_3, a_4]^T$ denotes the prediction coefficients and $\boldsymbol{\phi}_i = [B_w(n-3), B_w(n+3), C_w(n-1), C_w(n+1)]$ denotes the neighboring pixels of $D(n)$. Hence in order to predict all the pixels denoted by **D**, we need seven LS based optimized predictor of order 4. After prediction remaining $(1/4)^{th}$ pixels of type D, apply additive PEE method to get corresponding watermarked pixels.

Thus in first phase, for interpolation of LR image to high resolution image (**P**), we need 21 set of prediction coefficients i.e. 7 set of 6^{th} order predictor (for **A** type), 7 set of 5^{th} order predictor (for **B** type), 7 set of 5^{th} order predictor (for **C** type). For second phase, we need 7 set of 4^{th} order predictor (for **D** type). These coefficients estimated at encoder is sent to decoder as an overhead with final watermarked image ($\mathbf{I_w}$). Thus decoder does not need to estimate any coefficients.

4.3 Decoder of Proposed Algorithm

At the decoder side, we have watermarked image ($\mathbf{I_w}$) and 28 set of prediction parameters (sent from encoder). The decoder of proposed algorithm works as follows:

1. For each pixel of type **D**, estimate the slope S, and based on the value of S given in (7), corresponding 4^{th} order coefficients are used to predict the pixels using (12). Then apply inverse PEE using (4) to get the secret bits and original pixels. Thus odd-odd position of the image can be recovered easily.
2. Then follow raster scan order for rest of the pixels. After reconstruction of type **D** pixels, our proposed algorithm formulate the predictor structure for each pixel with the help of its causal neighboring original pixels and neighboring type **D** pixels.
3. From this predictor structure (consists of original causal neighboring pixels and original type **D** pixels), slope S is estimated and based upon the value of S corresponding prediction coefficient are used for prediction of pixels. Then apply inverse PEE method using (4) to get secret bits and original pixels.
4. Repeat steps 2 and 3 until all the pixels in the entire image are recovered. Recovering is done in raster scan order.

5 Simulation Result

We implemented our proposed reversible data hiding algorithm and compared its performance with existing methods. For this purpose, we use standard test images of 8 bit resolution and of 512×512 dimension.

We compared the predictor order of proposed algorithm with IPEE [[11]] and is shown in Table 2. From Table 2, we can say that proposed algorithm uses higher order predictor and is adaptive as well. Moreover, our prediction algorithm uses only original pixels for interpolation (first phase) and thus prediction accuracy increases as compared to IPEE. We compared the prediction

Table 2. Predictor Order (PO) comparison

Methods	Type A	Type B	Type C	Type D
IPEE [11]	4	4	4	4
Proposed	6	5	5	4

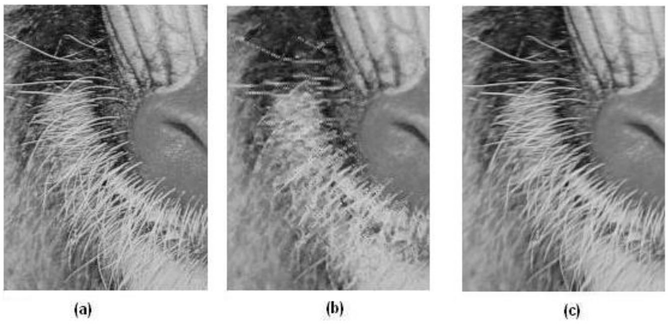

Fig. 4. Portions of (a) Original Baboon Image, (b) Interpolated image by IPEE [11] (c) Interpolated image by Proposed Method

performance of proposed algorithm with IPEE after interpolating LR image to high resolution image (i.e. after predicting pixels of type **A**,type **B** and type **C** respectively). Visual quality obtained by both the algorithms are shown in Fig. 4. We can find that IPEE produces blurred image as compared to original image where edges are more preserved by proposed algorithm. It is because proposed algorithm uses only original pixels for prediction. Moreover proposed algorithm obtains a higher order predictor as against the fixed 4^{th} order in IPEE (shown in Table 2).

Table 3. Performance comparison of Proposed Algorithm with various other methods - Embedding Capacity (number of bits)

Schemes	Lena	Baboon	Sailboat	Plane
Ni [3]	5,460	5,421	7,301	16,171
Lin [4]	59,900	19,130	37,644	80,006
Hu [8]	60,241	21,411	28,259	77,254
SJ [10]	66,512	22,685	42,241	91,890
IPEE [11]	71,674	22,696	38,734	84,050
Proposed	**74,412**	**25,341**	**45,651**	**95,890**

We also compared our results with other algorithms to check the effectiveness of proposed algorithm. The overhead required for sending the location map as

Table 4. E_C refers to embedding capacity in bits and PSNR in dB for plane image

Methods		Q=1	Q=2	Q=3	Q=4
IPEE [11]	E_C	84,050	117,500	164,100	198,450
	PSNR	48.94	43.60	40.5	38.01
Proposed	E_C	95,890	145,667	185,840	211,510
	PSNR	49.03	43.68	40.60	38.36

well as prediction parameters to decoder has been taken account. In Table 3, we consider the case when capacity parameter is one ($Q = 1$). It can be noticed that in all the four images, we achieve higher embedding capacity as compared to existing algorithms. We performed an experiment and is shown in Table 4. We can see that for fixed Q, proposed algorithm has better embedding capacity with less distortion as compared to IPEE.

To show effectiveness of our method for capacity parameters $Q >= 1$, we compared our results with existing algorithms and is shown in Fig. 5. It can be observed that proposed algorithm outperforms the existing algorithms for high capacity parameters ($Q >= 1$) as well.

The encoder of proposed algorithm requires more computational power as compared to IPEE [[11]]. It is because, proposed algorithm needs to estimate 28 set of LS based prediction coefficients which requires addition, multiplication and matrix inversion. These estimated LS based coefficients are quantized and then tagged with the watermarked image and is sent to decoder. Thus, decoder of proposed algorithm remains more simple as it needs not to estimate any prediction coefficients.

6 Conclusion

In this paper, we proposed a novel reversible data hiding algorithm based on additive PEE. We observed that interpolation method based on fixed order predictor fails to exploit the neighborhood pixels effectively for interpolation. In view of this, we developed an efficient interpolation method based on adaptive predictor structure that exploits correlation between the neighborhood pixels and the unknown pixels (to be interpolated) effectively. Moreover we proposed to use only original pixels for interpolation. Experimental results proves that the proposed algorithm results in a better performance than the existing methods.

References

1. R. Li, O. C. Au, C. K. M. Yuk, S. Yip, and T. Chan: Enhanced image trans-coding using reversible data hiding. IEEE Int. Symp. Circuits and Systems ISCAS 2007, pp. 1273-1276.
2. J.B. Feng, I.C. Lin, C.S. Tsai, Y.P. Chu: Reversible watermarking: current status and key issues. International Journal of Network Security, Vol.2, No.3, PP.161-11, May 2006

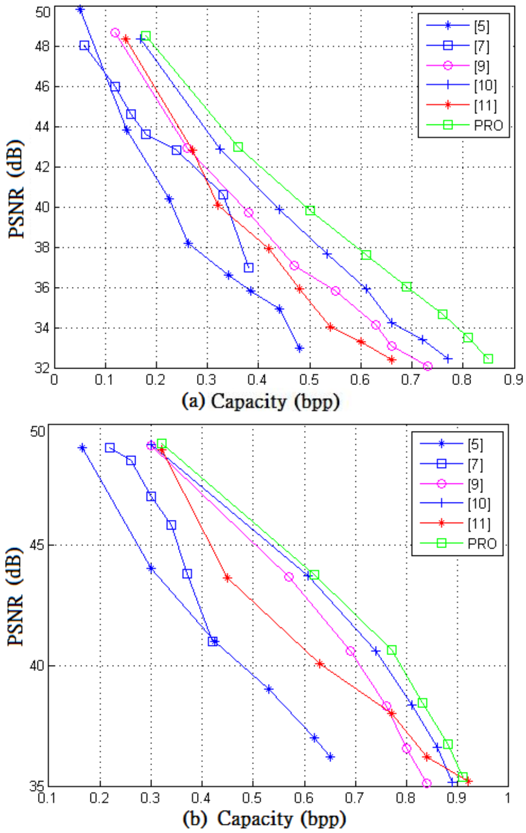

Fig. 5. For higher capacity parameters ($Q >= 1$) : (a) Sailboat (b) Plane. Here PRO refers to proposed algorithm and bpp refers to bits per pixel.

3. Z. Ni, Y. Q. Shi, N. Ansari, and S.Wei: Reversible data hiding. IEEE Trans. Circuits Syst. Video Technol. vol. 16, no. 3, pp. 354-362, 2006.
4. C. C. Lin and N. L. Hsueh: lossless data hiding scheme based on three-pixel block differences. Pattern Recogni., vol. 41, no. 4, pp.1415-1425, Apr. 2008.
5. K.-S. Kim, M.-J. Leea, H.-Y. Leeb, H.-K. Leea: Reversible data hiding exploiting spatial correlation between sub-sampled images. Pattern Recognit., 2009, DOI: 10.1016/j.patcog.2009.04.004.
6. J. Tian: Reversible data embedding using a difference expansion. IEEE Trans. Circuits Syst. Video Technol., vol. 13, no. 8, pp. 890-896, Aug. 2003.
7. H.-J. Kim, V. Sachnev, Y. Q. Shi, J. Nam, and H.-G. Choo: A novel difference expansion transform for reversible data embedding. in IEEE Trans. Inf. Forensic Security, vol. 3, no. 3, pp. 456-465, Sep. 2008.
8. Y. Hu, H.-K. Lee, and J. Li: DE-based reversible data hiding with improved overflow location map. IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 2, pp. 250-260, Feb. 2009.

9. M. Chen, Z. Chen, X. Zeng, and Z. Xiong: Reversible image watermarking based on full context prediction. International conference on Image Processing (ICIP), page 4253-4256. IEEE, (2009).
10. S. P. Jaiswal, O. C. Au, V. Jakhetiya, Y. Guo, A. Tiwari, K. Yue: Efficient Adaptive Prediction Based Reversible Image Watermarking. International conference on Image Processing (ICIP), IEEE, (2013).
11. L. Luo, Z. Chen, M. Chenm, X. Zeng, X. Zhang: Reversible Image watermarking using interpolation technique. IEEE Trans. Inf. Forensic Security, vol. 5, no. 1,pp. 187-193,march 2010.
12. J Tsai, H.M. Hang: Modeling of Pattern Based Block Motion Estimation and Its Application. IEEE Trans. Circuits Syst. Video Technol., vol. 19, no. 1,Jan. 2009.
13. A. K. Tiwari and R. V. Kumar: Least Sqaures Based Optimal Switched Predictor For Lossless Compression of Images. Multimedia and Expo, 2008 IEEE International Conference on , April 26 2008.
14. S.P. Jaiswal, V. Jakhetiya, A.K.Tiwari: A lossless image prediction algorithm using slope estimation and least square optimization. Instrumentation and Measurement Technology Conference (I2MTC), 2012 IEEE International, pp. 1567-1570, May, 2012.
15. V. Jakhetiya, S.P. Jaiswal, A.K.Tiwari: Interpolation based symmetrical predictor structure for lossless image coding. Circuits and Systems (ISCAS), 2012 IEEE International Symposium on, pp.2913-2916, May, 2012.
16. Prasad Jaiswal, S.; Mittal, G.; Jakhetiya, V.; Tiwari, A.K., An efficient two pass lossless invisible watermarking algorithm for natural images, Systems, Signals and Image Processing (IWSSIP), 2012 19th International Conference on , vol., no., pp.164,167, 11-13 April 2012.