J Electr Eng Technol.2016; 11(4): 974-986
http://dx.doi.org/10.5370/JEET.2016.11.4.974

ISSN(Print) 1975-0102
ISSN(Online) 2093-7423

Reversible Data Hiding Using a Piecewise Autoregressive Predictor Based on Two-stage Embedding

Byeong Yong Lee*, Hee Joon Hwang* and Hyoung Joong Kim†

Abstract – Reversible image watermarking, a type of digital data hiding, is capable of recovering the original image and extracting the hidden message with precision. A number of reversible algorithms have been proposed to achieve a high embedding capacity and a low distortion. While numerous algorithms for the achievement of a favorable performance regarding a small embedding capacity exist, the main goal of this paper is the achievement of a more favorable performance regarding a larger embedding capacity and a lower distortion. This paper therefore proposes a reversible data hiding algorithm for which a novel piecewise 2D auto-regression (P2AR) predictor that is based on a rhombus-embedding scheme is used. In addition, a minimum description length (MDL) approach is applied to remove the outlier pixels from a training set so that the effect of a multiple linear regression can be maximized. The experiment results demonstrate that the performance of the proposed method is superior to those of previous methods.

Keywords: Context prediction, Least-squared-based method, Minimum description length, Piecewise auto-regression, Prediction-error expansion, Reversible data hiding.

1. Introduction

Reversible or lossless data hiding techniques hide data in the host signal (for example, text, image, audio, or video) and allow for not only the extraction of the embedded message, but also the recovery of the original host signal [1-3]. The techniques can be applied to military or medical images that demand an exact representation of the original image. Reversible watermarking technology requires a perfect reversibility and a high performance for which a high embedding capacity and a low distortion are required.

Since the difference expansion (DE) method was introduced by Tian in 2003 [2], a number of algorithms for the production of far smaller difference values have been proposed [3, 4]. These methods accomplish reversible watermark embedding by expanding the prediction error that is the difference between the original and predicted pixel values. Prediction-error expansion (PEE) is a generalized form of difference expansion whereby the prediction error is expanded instead of the inter-pixel differences. The greater the accuracy of the applied prediction method, the lower the image distortion that occurs; consequently, the prediction method is one of the most important factors for the enhancement of the performance of reversible watermarking algorithms.

The first version of a reversible watermarking predictor was proposed by Thodi and Rodriguez [5]. Their "JPEG-LS" is a 3rd-order median edge detector (MED) predictor for which the lossless image compression standard is adopted [6]. MED is a context-based adaptive predictor for which three simple predictors that are related to the three neighboring pixels of west, north, and northwest are combined. Chen et al. [11] compared the performances of many predictors such as the MED and the 4th-order gradient-adjusted predictor (GAP) in the CALIC [10]. The edge-directed prediction (EDP) version that is based on piecewise 2D auto-regression (P2AR) [8] is a predictor with least-squared (LS) approaches for which the prediction coefficients are locally optimized inside a causal window, i.e., a training window.

Kau and Lin [9] proposed the edge-look-ahead (ELA) scheme for which LS-based prediction is used with an efficient edge detector to maximize the edge-directed characteristics. The performances of all of these predictors have been compared fairly in several papers [9, 10]; basically, the higher the order of the predictor, the higher the accuracy of the prediction, and among the considered predictors, the ELA yields the best performances for most of the images [14]. All of these predictors are proposed for image compression whereby only the causal pixels can be used, as causality must be maintained to ensure a perfect reversibility.

In this paper, the target pixel ("TP" or "current pixel" [13]) is the pixel represented by $x(n)$, and the corresponding prediction error is calculated. The LS method computes the coefficients of the neighboring support pixels (SPs) of the TP, while a window pane confines the shape and number of the SPs; for example, if four pixels are in the window, the 4th order pane is used. As stated previously, the supporting pixels should be causal to ensure reversibility,

† Corresponding Author: Graduate School of Information Security, Korea University, Seoul, Korea. (khj-@korea.ac.kr)

* Graduate School of Information Security, Korea University, Seoul, Korea. ({rom109101, ableman82}@gmail.com)

Received: September 1, 2015; Accepted: December 17, 2015

and if the original values of the pixels that are used for the prediction are retained, they are considered causal. Since the computation of the prediction error is based on the supporting pixels, the pixel values should be the same before both the embedding process and the decoding process. In image coding, the SPs consist of the pixels prior to the TP.

Advanced methods expand the LS-optimization problem in the training set (TS) by using multiple SP sets that are selected around the TS. For each TS pixel, any SS that is similar to the given support set of the TP is chosen as a member of the SP sets.

To enhance the reversible watermarking performance, the prediction does not need to be restricted to the causal pixels that are available before the TP in the scanning order. Raster scanning is the most popular order. Chen et al. [14] proposed a 4^{th}-order full-context prediction scheme for which the four neighboring SPs (i.e., 4th-order such as north, south, east, and west pixels) of a predicted pixel are used. Using the rhombus pattern, Sachnev et al. [1] also proposed a 4th-order full-context prediction scheme that is a two-stage embedding method, whereby the corresponding performance superiority confirms the salience of not only the number of SPs but also the SP locations regarding prediction accuracy.

Sachnev et al. [1] make a full-context prediction possible through an exploitation of two-stage embedding called “two-stage data hiding.” They split the pixels into two sets so that one of the sets can first be used to hide data, followed by the sequential use of the other set; moreover, the two sets (i.e., the dot set and the cross set) are non-overlapping and independent of each other. When the dot set is processed first, the cross set remains intact. The cross set is processed next, and the dot set is considered intact even though it has been modified. Here, the dot set processing is performed during the first stage and the cross set processing occurs during the second stage as a result, since one of the two sets retains the original values, and the other set retains original-like pixel values, full-context modeling becomes possible. Due to this remarkable feature, the pixels before the TP can be causal, and while the feature can play an important role in reversible water-marking, this is not the case for image compression.

In this paper, we propose a 6th-order full-context predictor whereby P2AR is combined with the two-stage embedding method. Contrary to the typical P2AR [8] and the full-context predictor of Chen at al. [14], the predicted pixel is surrounded by six SPs (north, southeast, west, northeast, north, and west pixels) that are on the nearest location, and the TS is modified to work properly for the two-stage embedding system. The proposed predictor of this paper outperforms the method of Sachnev et al. [1], which was the state-of-the-art algorithm. The performance of this predictor is also more effective than those of the other predictors that are used for image compression.

Numerous researchers have proposed advanced methods to improve upon the method of Sachnev et al., but they only succeeded with respect to a low embedding capacity [7, 15, 21]. Recently, Dragoi and Coltuc [19] proposed an advanced method that outperforms the method of Sachnev et al. [1]. In this paper, the goal is the proposal of another advanced method for which three kinds of new features are introduced. The first feature is the 6th-order of SPs; while Dragoi and Coltuc [19] used the 4th-order of SPs, higher orders cannot be used for their method due to the violation of causality. The second feature is an exploitation of the sorting that occurs during the selection process regarding the target pixels; after sorting, only the effective TS elements are used for prediction. The third feature is the shape of the TS. Dragoi and Coltuc [19] used a regular rectangle shape, while the proposed method uses a rectangle that is chipped on the bottom to ensure causality; as a result, the proposed method outperforms the methods of both Sachnev et al. [1] and Dragoi and Coltuc [19].

The organization of this paper is as follows: section 2 introduces the previous works that the proposed method is based on; section 3 presents the proposed algorithm; in section 4, we present the results of our experiments to show that our algorithm is superior to the other methods; and the conclusion is presented in section 5.

2. Related Works

2.1 Piecewise-2D Auto-regression (P2AR) Predictor

In this paper, we use $x(n)$ to denote the current TP for prediction where n is the spatial coordinate in an image; n is one-dimensional, but it can also be two-dimensional. Even though Fig. 1 is a two-dimensional TS shape, the pixel positions are represented by a one-dimensional notation. Regarding the P2AR predictor, suppose the image is scanned in a raster-scanning order, and $x(n)$ is predicted by its past causal neighboring pixels through the use of $x(n-1)$ to $x(n-12)$. According to the Nth-order Markovian property, the predicted value is calculated by the N neighboring pixels as follows:

$$\hat{x}(n) = \sum_{k=1}^{N} a(k)x(n-k), \qquad (1)$$

$x(n$-12)	$x(n$-11)	$x(n$-10)	$x(n$-9)	$x(n$-8)
$x(n$-7)	$x(n$-6)	$x(n$-5)	$x(n$-4)	$x(n$-3)
$x(n$-2)	$x(n$-1)	$x(n)$		

Fig. 1. Twelve causal neighboring pixels around the target pixel $x(n)$

where $a(k)$ is the prediction coefficient.

The P2AR predictor adapts to the local features around the target pixel on a pixel-by-pixel basis [11]; that is, the relation between each TS pixel and its SPs is very useful for the prediction of the relation between the TP and its SPs, as shown in Fig. 2. The relation between the TP and its SPs is expressed by $a(k)$, which are the so-called prediction coefficients that are optimized locally inside the TS.

A convenient TS choice is a rectangular window (i.e., in blue color) that contains $M = 2T \cdot (T+1)$ causal neighboring pixels, as shown in Fig. 2. In this figure, there are 12 SPs (i.e., red-color crosses) around the TP, $x(n)$, and since T is 4, M is 40 in the TS. Each pixel in the TS is denoted as $x_k(n)$, where n is the position of the TP and $k = 1, \cdots, M$. Let the TS be denoted by an $M \times 1$ column vector, as follows:

$$X = [x_1(n) \quad x_2(n) \quad \cdots \quad x_M(n)]^T \tag{2}$$

Each pixel in the TS has the SPs that consist of the N closest pixels (in red color), as shown in Fig. 2. Then, the pixels of X and their SPs would form an $M \times N$ matrix C, as follows:

$$C = \begin{bmatrix} x_1(n-1) & \cdots & x_1(n-N) \\ \vdots & \ddots & \vdots \\ x_M(n-1) & \cdots & x_M(n-N) \end{bmatrix} \tag{3}$$

where $x_m(n-k)$ is the *k-th* SP of the *m-th* TS, $x_m(n)$. A certain pixel subset is taken from among the M pixels of the TS for the linear regression.

The prediction coefficients are solved through the following intra-TS LS optimization: $\min\|X - Ca\|$. It is well known that the P2AR optimization comprises the following closed-form solution:

$$a = (C^T C)^{-1} C^T X \tag{4}$$

where $a = [a(1), a(2), \cdots, a(N)]^T$ comprises optimized prediction coefficients that need to be multiplied by the SP values. The latter optimization provides a solution for direction adaptation and the local approximation of the optimal prediction inside the TS.

2.2 Minimum description length (MDL)

The selection of the sample set S out of the TS is a key issue for the prevention of model overfitting [13]. In the LS-based adaptive predictor for which multiple linear regressions are utilized, the similarity between the SPs of the TP and the training pixels is very important; if a number of training pixels that are not similar to the TP are included, the LS-based adaptive predictor does not provide an accurate prediction. Wu et al. [13] proposed the minimum description length (MDL) criterion to properly determine the SP quantity and locations regarding the predictor and for the selection of similar-patterned TS pixels. The MDL concept is applied for the proposed prediction method.

2.3 Two-stage embedding method using rhombus pattern

In Fig. 2, the predicted TP value is computed based on the pixels marked with red crosses. It is a well-known fact that the nearest four neighboring pixels in the north, south, east, and west of the TP can be the most favorable candidates for prediction; however, the pixels in the east and south are not causal. The most favorable SPs therefore look like the pattern of Fig. 2 whereby a single-stage embedding algorithm has been used.

Sachnev et al. [1] divided an image into two kinds of non-overlapping pixel groups with a rhombus-patterned window, whereby two groups, the cross and dot sets of Fig. 3, are formed; in this figure, the four dots around a center-cross form a rhombus pattern. The predicted values of the four dots are obtained using the arithmetic mean value of the four neighboring pixels around the TP, and this method is regarded as the simplest version of a full-context prediction; for example, a cross-set pixel is predicted by the four-closest dot-set pixels, as shown in Fig. 3. Because of an excellent grouping, this method succeeds in the construction of a full-context prediction, and the corresponding improvement of the prediction performance over all of the other prediction methods, for which the SPs include only the prior region (i.e.,

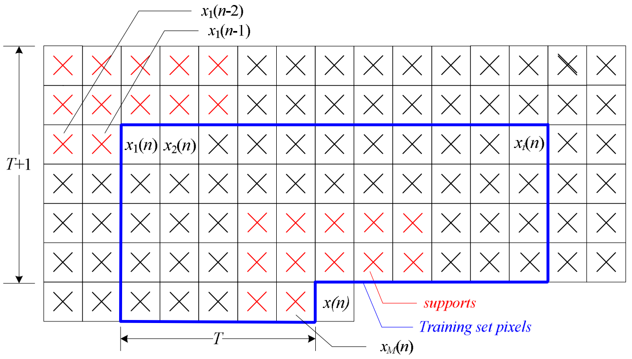


Fig. 2. Training set and support pixels

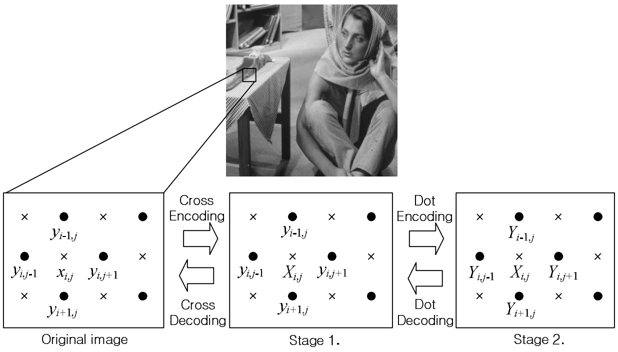


Fig. 3. Two-stage embedding method

approximately westward and northward) of the TP, is significant. Unfortunately, such predictors cannot produce a more effective performance even though high-order modeling such as those of the CALIC and P2AR methods is used [14]. The Sachnev et al. rhombus pattern [1] also allows for the use of the south and east SPs for prediction without any consideration of causality.

2.4 Sorting prediction errors by local variance among four neighboring pixels

In Sachnev et al.'s method [1], the cross and dot sets of the rhombus scheme are independent and non-overlapping regarding each other, meaning that it is possible to calculate the correlation parameter of each pixel of the cross set by using the neighboring four dot-set pixels and vice versa. This is the *local variance value* and is calculated as follows:

$$\sigma^2 = \tfrac{1}{4}\textstyle\sum_{k=1}^{4}(\delta_k - \mu)^2 \tag{5}$$

where

$$\delta_1 = |y_{i,j-1} - y_{i-1,j}|,\ \delta_2 = |y_{i-1,j} - y_{i,j+1}|,$$
$$\delta_3 = |y_{i,j+1} - y_{i+1,j}|,\ \delta_4 = |y_{i+1,j} - y_{i,j-1}|,$$

The notation of the pixel position here is based on Fig. 3.

An important feature of the local variance value tends to be proportional to the magnitude of the prediction error; therefore, for some of the pixels for which the local variance is too high, the embedding of one bit of each is skipped if some pixels with small local variance values remain. Because of this property, the local variance value can also be utilized as a criterion that indicates whether the pixels are worthy of embedding or not.

3. Proposed Algorithm

Compared with the existing reversible data hiding (RDH) methods, a more accurate predictor is used to achieve the improved performance of the proposed method. The proposed idea is mainly based on the two-stage embedding system and the P2AR predictor. We focus on the improvement of the prediction accuracy by ensuring the following:

- Use of two additional critical SPs over the four SPs of Sachnev et al.'s rhombus scheme [1].
- Application of the P2AR predictor after the selection of the TS pixels in accordance with the MDL criterion.
- Filtration of highly improper pixels according to the local variance value whereby bits are not embedded on them.

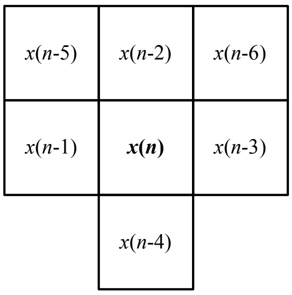


Fig. 4. Support pixels of proposed predictor ($N = 6$)

3.1 P2AR predictor using six-nearest support pixels

Basically, the proposed predictor is based on the two-stage embedding system for which a rhombus pattern is used. Unlike Sachnev et al.'s method [1], it is the 6th-order predictor that is shown in Fig. 4. We include the six SPs that are on the west, north, east, south, northwest, and northeast of the TP, $x(n\text{-}1)$, $x(n\text{-}2)$, $x(n\text{-}3)$, $x(n\text{-}4)$, $x(n\text{-}5)$, and $x(n\text{-}6)$. Because Sachnev et al. [1] used a sorting method, the pixels of the same group as the TP should not be exploited as an SP; that is, for the prediction of the cross pixels, only the dot pixels need to be used. If the cross pixels are used to predict the TP in the cross set and sorting is applied, it is impossible to recover the original order from the sorted pixels during the decoding process, and the reason for this is simple, as follows: Causality can be violated by using their method [1] due to the values of $x(n-5)$ and because the $x(n-6)$ values are not original.

If it is assumed that $x(n)$ should belong to the dot set, then $x(n\text{-}1)$, $x(n\text{-}2)$, $x(n\text{-}3)$, and $x(n\text{-}4)$ belong to the cross set and are independent of the TP $x(n)$. Here, even though $x(n\text{-}5)$ and $x(n\text{-}6)$ belong to the same set as the TP, they can still be used for prediction since they are causal to the TP. As a result, six pixels can be used as an SP set.

The P2AR optimization is processed on a pixel-by-pixel basis. According to our method, the prediction value $\hat{x}(n)$ for the TP $x(n)$ can be expressed as Eq. (1). Regarding the P2AR, the prediction method locally optimizes the prediction coefficients a inside a causal window, which is the TS of Fig. 5.

Due to the property of the two-stage embedding system, a number of pixels that do not belong to the TS exist. One example of the TS (e.g., the value of T is 4) is presented in Fig. 5, whereby the TS consists of 35 M elements. Notably, the near-rectangular form of the traditional TS, as shown in Fig. 2., differs from the proposed form of the TS in Fig. 5 that looks like a chipped rectangle at the bottom; this chipped pattern is obtained because of the non-causality property of some of the SPs, and the chipped parts belong to the opposite set (i.e., cross set or dot set). The chipped parts, however, should be excluded for ensuring causality, whereby the new M value should be recalculated, as follows:

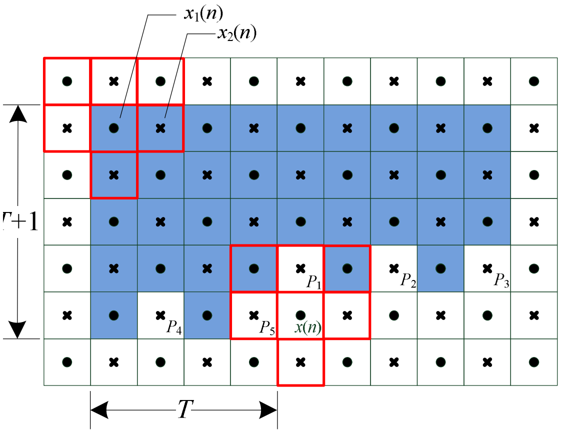


Fig. 5. A training set for prediction (in this example $T = 4$)

$$M = 2T(T+1) - \left\{2\left(\frac{T}{2}\right) + 1\right\} = 2T^2 + T - 1 \qquad (6)$$

where T is an even number. In Fig. 5, basically all of the pixels that belong to the dot set can be included in the TS; alternatively, regarding the cross set, the TS should not include some of the cross-set pixels such as P_1, P_2, P_3, and P_4 even though they are causal. In the case of P_5, it is excluded from the TS since its SPs include the TP $x(n)$. Since the other cross-set elements within the chipped rectangle do not violate causality, they are not excluded from the TS.

3.2 Selecting proper samples from training set using MDL

The proposed method takes advantage of another idea for the selection of more suitable pixels for the purpose of improving the P2AR optimization. The TS of Fig. 5 and Fig. 6 are the same, but the pixels of Fig. 6 set that is within the TS are of a dark color. In Fig. 5, all of the pixels in the TS can be used for the LS optimization, whereas in Fig. 6, only a certain number of candidates are used for the optimization. Further, other pixels are not used if it is possible that the corresponding SPs may not be similar to those of the TP. Similarity is measured according to the MDL.

In most image-block cases, not all of the pixels in the TS are effective for the estimation of prediction coefficients because a number of the pixels are irrelevant to the TP that functions as an outlier. If those pixels affect the P2AR optimization, the accuracy of the prediction can be decreased due to overfitting (i.e., context dilution). We therefore utilize MDL as a criterion for the selection of specific pixels for which there is a sufficient correlation with the TP. Whether the correlation between the pixels of the target set and the TS is sufficient or not can be defined by the MDL criterion.

We apply the MDL criterion τ for the selection of the proper element pixels from the TS and this set is called "selective set S," as follows:

$$D\big(x_i(n)\big) = \textstyle\sum_{k=1}^{N} \|x_i(n-k) - x(n-k)\|, \qquad (7)$$

where $x(n-k)$ is an SP of the TP $x(n)$, $x_i(n-k)$ is an SP of the TP $x_i(n)$, and $\|x\|$ denotes the L_1-norm of x. The D value is used to search for a set of SPs that is similar to the target set. The selected SPs are denoted as "set S" as follows:

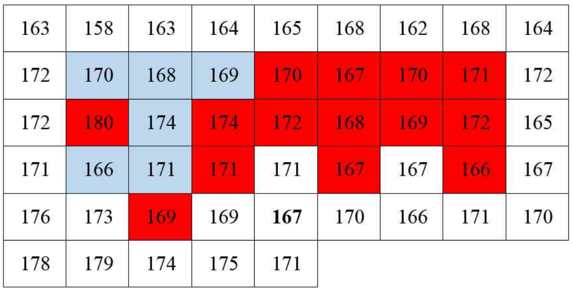
$$S = \{x_i(n-i) | D(x_i(n-i) \le \tau\}, \qquad (8)$$

where $x(n-i) \in X$ such as X is an $M \times 1$ column vector of the TS pixels, as shown in Fig. 5.

We select the pixels that satisfy the above condition to obtain S, whereby $D(x_i(n))$, the summation of the absolute values of the differences between the TP's SPs and each TP's SPs, needs to be smaller than τ. The threshold value τ is chosen empirically as 20 to achieve a favorable performance. The minimum number of the elements of the selective set is also important because the prediction performance worsens if it is too small or too large. In this paper, the value of T is set as 8, meaning that M is 135 according to Eq. (6); therefore, the cardinality of the selective set, $\|S\|$, is chosen empirically as 80 or more.

The vector of the prediction coefficients a that is obtained using the selective set S that satisfies the LS optimization comprises a closed-form solution, as shown in Eq. (4). After the prediction coefficients have been obtained on a pixel-by-pixel basis, the $\hat{x}(n)$ can be provided for the TP and the prediction errors are utilized for the embedding scheme.

The following example shows how the MDL works regarding a real-image sample: In Fig. 6, all of the colored pixels are TS pixels, and the center pixel with a bold font (on the 5th column), 167, is the TP; moreover, the SPs of the 167 TP sample are shown in Table 1. In this example, for the TS pixel in the top-left corner, 170, its $D(170) = \|172 - 169\| + \|158 - 171\| + \cdots + \|167 - 163\| = 3 + 13 + 2 + 9 + 8 + 4 = 39$ is computed according to



163	158	163	164	165	168	162	168	164
172	170	168	169	170	167	170	171	172
172	180	174	174	172	168	169	172	165
171	166	171	171	171	167	167	166	167
176	173	169	169	**167**	170	166	171	170
178	179	174	175	171				

Fig. 6. Training set sample selected by MDL

Table 1. Example of target pixel and its support pixels

Support Pixels						Target pixel
$x(n$-1)	$x(n$-2)	$x(n$-3)	$x(n$-4)	$x(n$-5)	$x(n$-6)	$x(n)$
W	N	E	S	NW	NE	Center
169	171	170	171	171	167	167

Eq. (7) (see first row of Table 2). The training-set pixels, $x_i(n)$, and their SPs are shown in Table 2, while the $D(x_i(n))$of each pixel is computed and shown in the last column. Once the $D(x_i(n))$value of the pixels is smaller than 20, they are colored red and selected for the P2AR calculation, as shown in both Fig. 6 and Table 2, whereby the blue-colored pixels are not selected since their D values are larger than 20.

3.3 Encoder and decoder

While this section describes the main steps of the encoding and decoding processes, a description of the full process allows for a more explicit step-by-step expression of the proposed idea. Basically, the embedding process follows a histogram-shifting method whereby two bins are selected based on their indices for the obtained histogram; that is, T_n and T_p, where $T_n < T_p$. In the histogram, the bin indexed as T_n and T_p is used for data hiding for which the following rules apply:

Table 2. Exemplary TS pixels and their SPs

i	x_i(n-1)	x_i(n-2)	x_i(n-3)	x_i(n-4)	x_i(n-5)	x_i(n-6)	x_i(n)	D(x_i(n))
1	172	158	168	180	163	163	170	39
2	170	163	169	174	158	164	168	29
3	168	164	170	174	163	165	169	21
4	169	165	167	172	164	168	170	18
5	170	168	170	168	165	162	167	18
6	167	162	171	169	168	168	170	18
7	170	168	172	172	162	164	171	19
8	172	170	174	166	172	168	180	15
9	180	168	174	171	170	169	174	21
10	174	169	172	171	168	170	174	15
11	174	170	168	171	169	167	172	10
12	172	167	169	167	170	170	168	16
13	168	170	172	167	167	171	169	16
14	169	171	165	166	170	172	172	16
15	171	180	171	173	172	174	166	22
16	166	174	171	169	180	174	171	25
17	171	174	171	169	174	172	171	16
18	171	168	167	170	172	169	167	12
19	167	172	167	171	169	165	166	10
20	173	171	169	174	166	171	169	17

$$e'(k) = \begin{cases} e(n) & \text{if } e(k) = T_n \text{ and } b_k = 0 \\ e(n) - 1 & \text{if } e(k) = T_n \text{ and } b_k = 1 \\ e(n) - 2 & \text{if } e(k) < T_n \end{cases}$$

$$e'(k) = \begin{cases} e(n) & \text{if } e(k) = T_p \text{ and } b_k = 0 \\ e(n) + 1 & \text{if } e(k) = T_p \text{ and } b_k = 1 \\ e(n) + 2 & \text{if } e(k) > T_p \end{cases}$$

3.1.1 Encoder

As shown in Fig. 7, the original image is divided into the Cross set and the Dot set for the two-stage embedding. The following Cross-set-embedding procedure is therefore given:

Step 1. Every pixel of the Cross set is predicted by the proposed predictor. The prediction-error value $e(n) = \hat{x}(n) - x(n)$ and the local variance $\sigma^2(n)$ are computed according to Eq. (5).

Step 2. *Virtual embedding step* to find σ_L^2, which is the last variance value that satisfies the capacity requirement, T_n, and T_p.

Step 2-1. Sort the pixels in the TS according to the local variance $\sigma^2(n)$ and assume that $\bar{E}_S$ is the sorted row of $\bar{E}$ that is a vector of the prediction errors such as $\bar{E} = [e(1), e(2), e(3), \cdots, e(n)]^T$. Properly find

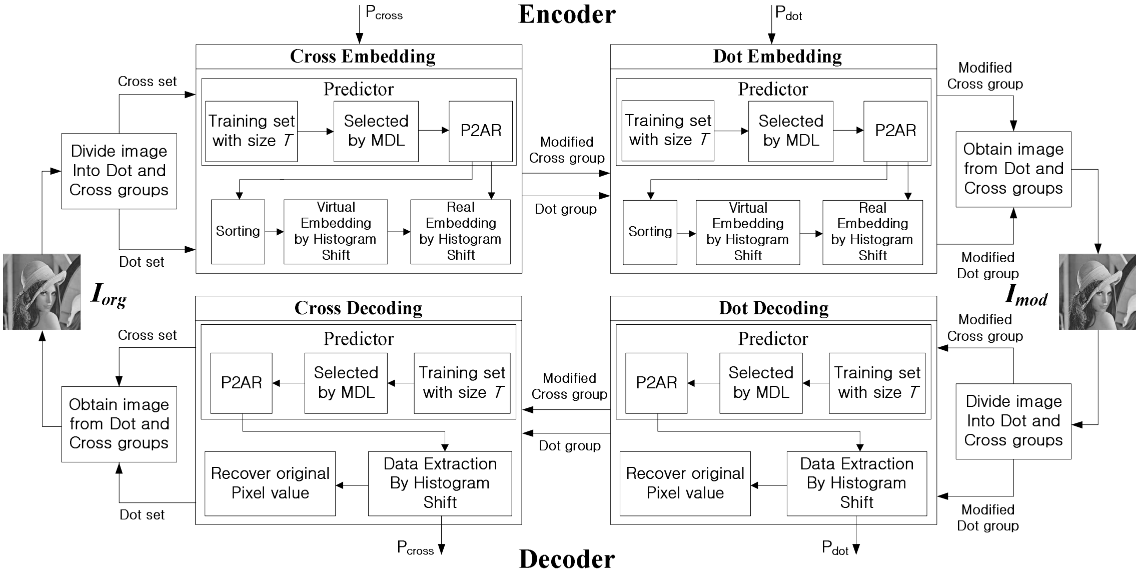


Fig. 7. Encoder and decoder of the proposed method Step 4. Compute the original pixel value by using the recovered prediction-error values.

T_n and T_p for a sufficient number of prediction errors that belong to $[T_n;\ T_p]$ to satisfy the capacity of the Cross-set payload P_c.

Step 2-2. Virtual embedding system for the attainment of σ_c^2. The local variance value $\sigma^2(n)$ is sorted from the smallest to the largest values. Count the number of prediction errors that produce local variance values smaller than σ_c^2. Among the prediction-error values, only P_c errors are selected to hide P_c message bits. Obviously, the P_c and σ_c^2 values should be delivered to the decoder.

Step 3. *Real embedding step* to embed payload

Step 3-1. Find the P_c pixels of which the $\sigma^2(n)$ values are less than σ_c^2, checking in raster-scanning order from the beginning of the image; only the prediction errors of those pixels are exploited for the embedding of the payload using the histogram-shift method. If overflow or underflow occurs in some of the pixels, location-map bits are generated. The size of the location map (usually compressed) should be delivered to the decoder.

Step 3-2. Side information about T_n and T_p, the payload size P_c, the local variance threshold value σ_c^2, and the location-map size can be expressed in under 60 bits. The information must be embedded in the beginning of the Cross-set pixels using the LSB-replacement method. The original LSB values of the 60 pixels need to be collected from the start as well; therefore, these 120 bits are embedded as side information so that this method remains completely reversible.

Step 4. The procedure for Dot-group embedding is the same as that from Step 1 to Step 5, for which the modified Cross-set pixels are used. For the Dot set, the side information regarding T_n and T_p, the payload size P_d, the local variance threshold value σ_c^2, and the location-map size should be decided accordingly.

3.1.2 Decoder

Here, the watermarked image is divided into the Cross set and the Dot set. The order of the two-stage decoding procedure is the inverse of the order of the two-stage encoding procedure; that is, the Dot-set decoding proceeds first, followed by that of the Cross set.

The Dot-set decoding procedure is given as follows:

Step 1. Every pixel of the Dot set is predicted by the proposed predictor. Calculate the prediction-error values and local variance $\sigma^2(n)$.

Step 2. From the beginning, side information is extracted from the 60 pixels.

Step 3. Find the proper pixels with $\sigma^2(n)$ values that are less than σ_d^2, checking in raster-scanning order from the beginning of the image. For the P_d pixels that are found, recover the original prediction error using the decoder of the histogram-shift method and extract the payload P_d. If the pixel belongs to an overflow/underflow case, follow the location-map algorithm according to the method of Sachnev et al. [1].

Step 4. Compute the original pixel value by using the recovered prediction-error values.

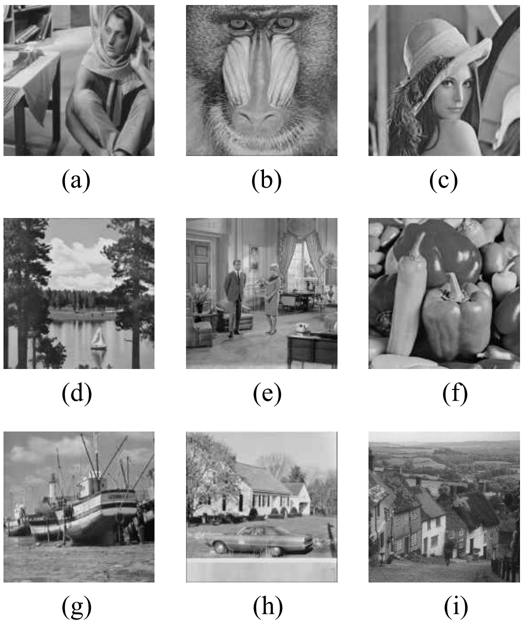
Step 5. Replace the first 60 LSB values with the original LSB values that are stored as side information.

Step 6. The Cross-set decoding follows the same procedure as that from Step 1 to Step 5 and the recovered Dot-set pixels are used.

4. Experiment results

We used MATLAB and the following standard test images (512 × 512-sized, 8-bit grayscale images), which are shown in Fig. 8, to implement the proposed reversible data hiding method: (a) *Barbara*, (b) *Baboon*, (c) *Lena*, (d) *Sailboat*, (e) *Couple*, (f) *Pepper*, (g) *Boat*, (h) *House*, and (i) *Goldhill*. The reversibility between the original image and the recovered one is proven throughout the experiment results. We embedded a random bit stream as the watermark message and the real binary data was embedded for side information.

We adopted a TS size T of 8 and an MDL value τ of 20 that were obtained through the conduction of extensive experiments. The peak signal-to-noise ratio (PSNR) is utilized to compare the performances of the reversible data-hiding methods. For comparison, the embedding capacity range is limited from 0 bit per pixel (bpp) to 1 bpp, and the



(a) (b) (c)

(d) (e) (f)

(g) (h) (i)

Fig. 8. Nine standard test images: (a) *Barbara*, (b) *Baboon*, (c) *Lena*, (d) *Sailboat*, (e) *Couple*, (f) *Pepper*, (g) *Boat*, (h) *House*, and (i) *Goldhill*

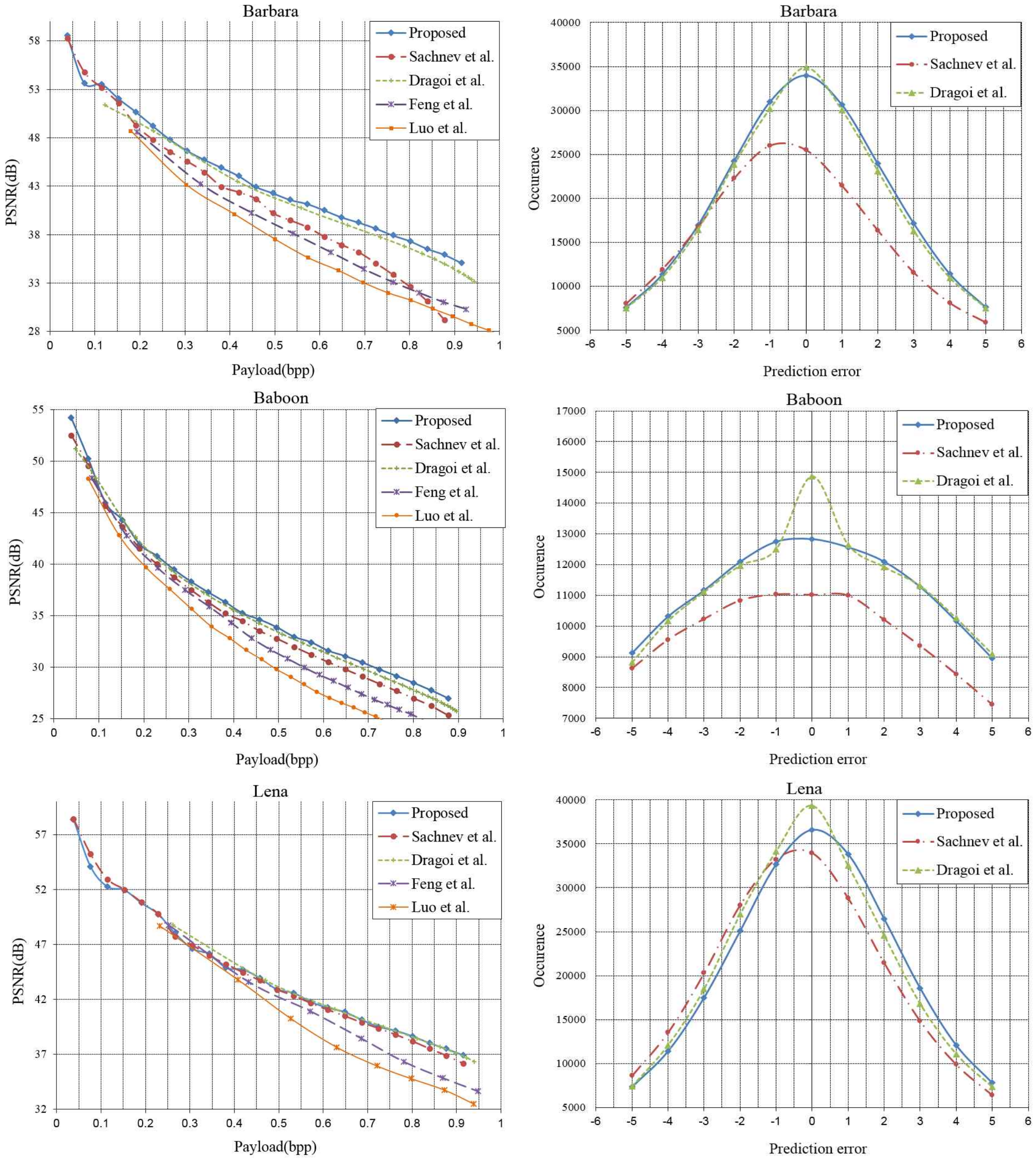


Fig. 9. PSNR and prediction-error histogram (in 10,000 bits) for comparison of *Barbara*, *Baboon*, and *Lena*

proposed method is compared with the works of Sachnev et al. [1], Luo et al. [17], Feng et al. [16], and Dragoi and Coltuc [19].

The results of the comparisons with the other schemes are shown in Fig. 9, Fig. 10, and Fig. 11, wherein the method of Dragoi and Coltuc [19] and the proposed method of this paper outperform the method of Sachnev et al. [1]. It is obvious that the proposed method outperforms the other methods in the case of a large embedding capacity such as those greater than 0.5 bpp. As mentioned in Section 1, the state-of-the-art method of Sachnev et al. [1] had previously been unbeatable, especially regarding a large embedding capacity. Recently, however, the method of Dragoi and Coltuc [19] outperformed Sachnev et al. [1] with respect to a large embedding capacity, positioning Dragoi and Coltuc's method [19] at the center of a new reversible watermarking challenge.

For small embedding capacities, numerous methods [7, 15, 21] have achieved performances that are more effective than that of Sachnev et al. [1]; thus, the main goal of this paper is the achievement of an improved performance regarding a large capacity. For the smooth images that are rich in low-frequency components, the P2AR method of Dragoi and Coltuc [19] is more effective, whereas the use of the proposed method achieves a more effective performance regarding complex images that are rich in high-frequency components. In the cases of *Baboon*, *Barbara*, *Pepper*, and *Goldhill*, the proposed method is

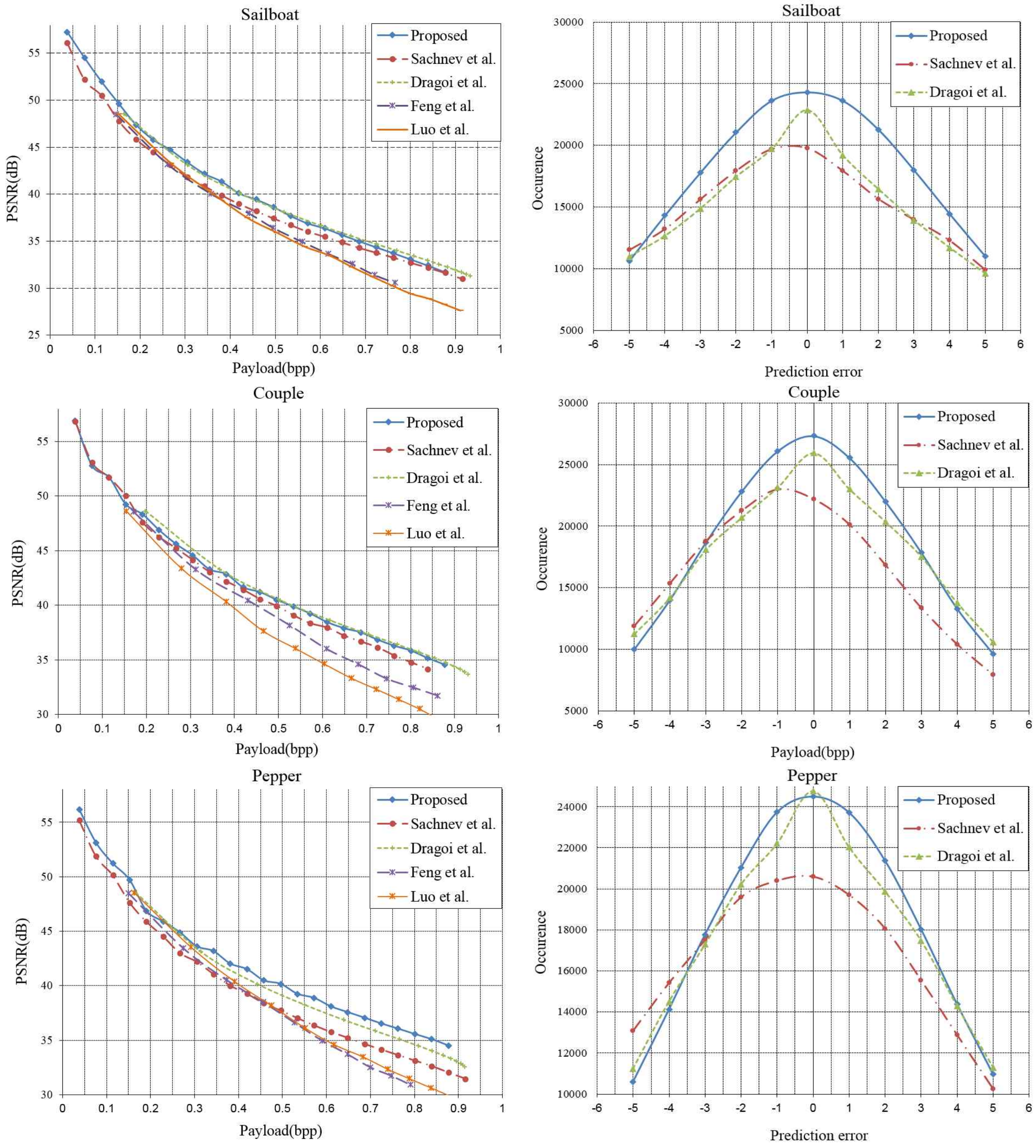


Fig. 10. PSNR and prediction-error histogram (in 10,000 bits) for comparison of *Sailboat, Couple*, and *Pepper*

more effective than any of the other methods. In the case of *Couple*, however, the P2AR method of Dragoi and Coltuc [19] is slightly more effective regarding a small embedding capacity. For the other images such as *Boat*, *House*, *Lena*, and *Sailboat*, the use of both methods resulted in even performances.

Sachnev et al. [1] utilized a rhombus scheme for which the term *full context* applies whereby it is possible to predict the TP by using the four neighboring pixels [14]; alternatively, a further-enhanced prediction method involving the addition of two more SPs is exploited for the proposed idea [1], meaning that the proposed method utilizes six SPs for the prediction of one TP. The TS is also exploited for the calculation of the P2AR-weight coefficients. The proposed method ultimately provides an outstanding prediction performance among the existing methods.

In addition, the attainment of proper weight coefficients is more effective in the high-frequency region (i.e., around the edges); therefore, the best increasing performance occurs in *Barbara*, which has a large amount of well-ordered edges. The *Barbara* performance is even superior to that of *Baboon*, the texture of which is only complicated, rather than being a specific pattern.

The other reason for a high PSNR regarding a high capacity in a high-frequency image is the location-map size that is used for the prevention of over/underflow errors. The location-map size of the proposed method decreases

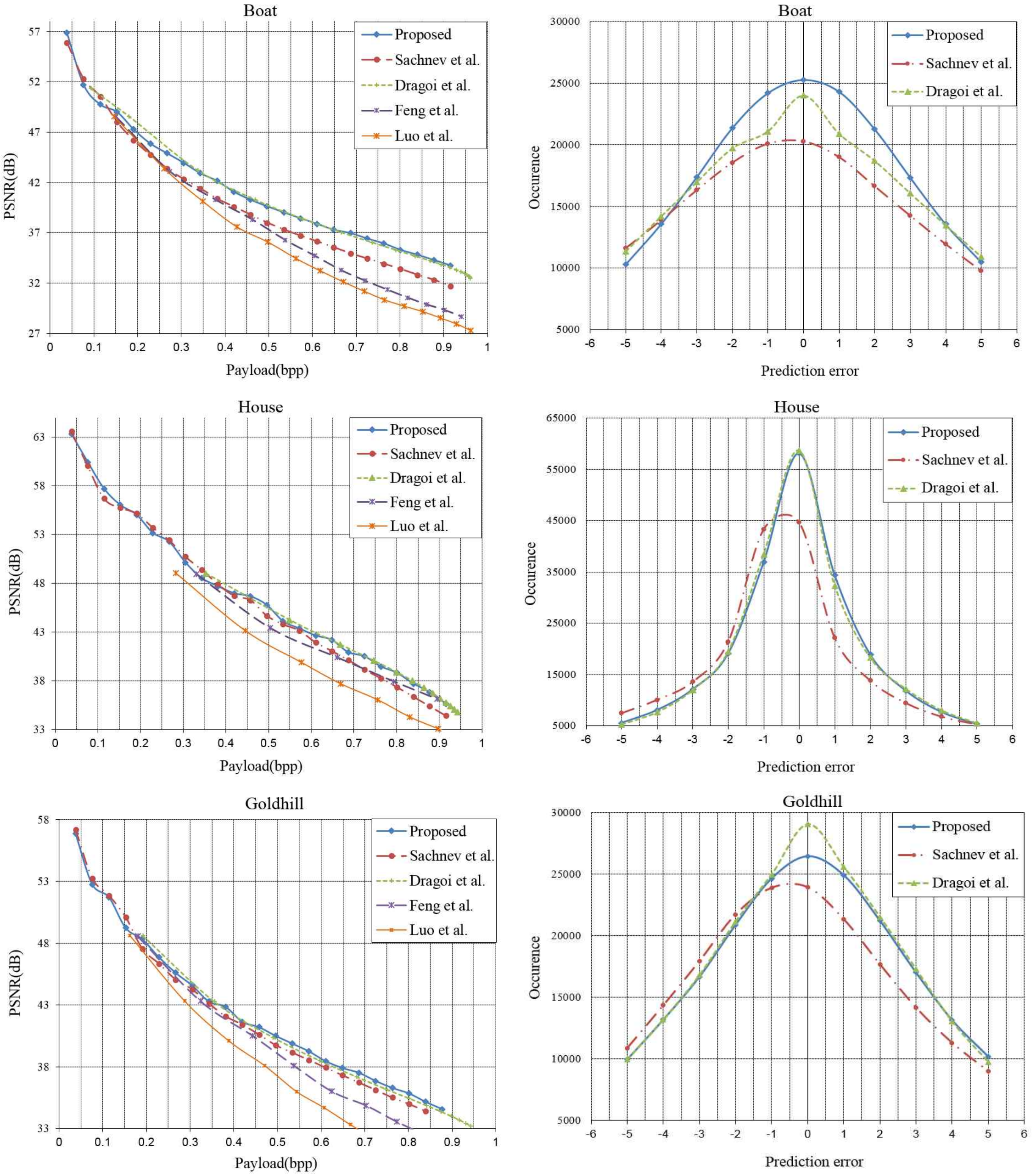


Fig. 11. PSNR and prediction-error histogram (in 10,000 bits) for comparison of *Boat, House*, and *Goldhill*

dramatically compared to that of Sachnev et al.'s method [1].

Compared with Sachnev et al. [1], the proposed method requires a small location-map size. For *Barbara*, the location-map size of Sachnev at al. [1] at the embedding capacity of 230,000 bits is 3,611 bits, while only 35 bits are required for the proposed method; similarly, for *Baboon*, the required location-map sizes are 4,789 and 1,836 bits, respectively (See Table 4). It is obvious that the proposed method requires a smaller location-map size for many images. The exception, however, is *Sailboat* whereby the required number of bits for Sachnev at al. [1] is 301 bits and that for the proposed method is 750 bits. The performance of the proposed method in terms of PSNR vs.

Table 3. Comparison of kurtosis values

	[1]	[19]	Proposed
Barbara	5.946	8.068	7.866
Baboon	3.187	3.536	3.502
Lena	8.525	8.359	7.719
Sailboat	4.899	4.785	5.748
Couple	5.641	6.294	7.395
Pepper	5.198	5.135	5.770
Boat	5.219	4.701	5.951
House	7.038	9.444	9.554
Goldhill	5.515	6.051	5.756
Average	5.685	6.264	6.585

Table 4. Comparison of location-map sizes

Payload (bit)	Baboon			Barbara			Lena			Sailboat			Couple		
	[1]	[19]	Proposed	[1]	[19]	Proposed	[1]	[19]	Proposed	[1]	[19]	Proposed	[1]	[19]	Proposed
190,000	445	192	195	16	0	1	0	0	0	11	2	19	1,576	1403	1,150
210,000	1,140	493	498	329	1	3	0	0	2	65	21	110	2,278	2041	1,690
230,000	4,789	2047	1,836	3,611	28	35	2	0	2	301	154	750	2,773	2809	2,163

Table 5. Comparison of location-map sizes

Payload (bit)	Pepper			Boat			House			Goldhill		
	[1]	[19]	Proposed	[1]	[19]	Proposed	[1]	[19]	Proposed	[1]	[19]	Proposed
180,000	3,091	2,700	2,383	245	84	151	0	0	0	0	0	0
200,000	3,593	3,660	3,038	370	132	243	0	0	0	0	0	0
220,000	4,290	5,641	3,557	653	394	361	6	1	1	4	2	3

embedding capacity is also improved since its predictor is more effective.

The method of Dragoi and Coltuc [19] also requires a small location-map size; for *Barbara* and *Baboon*, for example, 28 and 2047 bits, respectively, are required at the same embedding capacity. Accordingly, their method and the proposed method perform far more effectively than the method of Sachnev et al. [1] regarding a large embedding capacity.

Table 4 and Table 5 show how many location-map bits are reduced regarding the proposed method; here, the creation of the location map is the same as that of Sachnev et al.'s idea [1]. In terms of the over/underflow problem, extremely high or extremely low intensity pixels are where the problem often occurs. A prediction improvement favorably influences the restraining of over/underflow errors.

The prediction-error histograms of all of the test images are shown in Fig. 9, Fig. 10, and Fig. 11. These figures depict the lines of the proposed idea and the idea of [19] as sharper than that of [1]. The prediction method for which P2AR is used is therefore far more accurate than [1] since it works more adaptively with the context pixels, while [1] works only with the fixed pixel values.

Kurtosis [22] is a measure of whether the data are peaked or flat relative to a normal distribution. The kurtosis value of the normal distribution is 3, while that of the Laplace distribution is 6, meaning that the Laplace-distribution peak and tails are sharper and thinner, respectively, than those of the Gaussian distribution. Once the kurtosis value is larger, its prediction method is more accurate. Table 3 shows that the average kurtosis values of the proposed method and [19] are quite larger than that of [1], and that the value of the proposed method is slightly larger than that of [19], reinforcing the relatively higher prediction accuracy of the proposed method compared with the other methods.

It is, however, clear that the actual performance of the proposed method cannot be explained using a single parameter such as the kurtosis value. The performances of the reversible data hiding methods cannot be explained clearly by using either the location-map sizes in Table 4 and Table 5, the histograms of the prediction errors in Fig. 9, Fig. 10, and Fig. 11, or the kurtosis values in Table 3, since many unknown parameters affect algorithmic performances. One thing that is clear is the quite favorable functionality of the proposed method that is due to the successful predictor proposed in this paper.

5. Conclusion

In this paper, we propose an enhanced predictor that is based on a combination of PAR and a rhombus-shaped two-stage embedding scheme; furthermore, the use of the embedding filter according to the variance around the TP produces an effect that is the same as that from the exploitation of the sorting. Our predictor utilizes six critical full-context SPs through the pixels in the TS, enabling an identification of the shape of the region around the TP and the proper coefficients; therefore, a set of pixels located in a highly variative region of an image is predicted more effectively when the proposed scheme, rather than Sachnev et al.'s method, is used [1]. Due to this full-context SP utilization, the number of prediction errors decreases and the size of the location map decreases greatly; therefore, the tendency of the proposed method significantly improves the embedding capacity, especially regarding highly variative images. The experiment results demonstrate that the results of the proposed method are more favorable than those of the other methods.

For the proposed method, we utilize the fact that a combination of causal pixels, cross pixels, and a partial group of dot pixels, or dot pixels and a partial group of cross pixels, at the decoding phase can be used for prediction. For the original rhombus-prediction method, only the cross pixels or the dot pixels were used for prediction because the researchers were unaware that the cross-utilization of pixels is possible; in this paper, the scope of causal pixels is widened. In addition, the PAR method becomes adaptive in this paper whereby adaptability is explained as follows: among M candidates of the TS pixels, we can use only $||S||$ candidates adaptively without the existence of side information between the encoder and the decoder by utilizing a threshold parameter τ.

The complexity of the proposed method, however, may be an issue. For each pixel, approximately one of the pseudo-inverse matrix computations of Eq. (4) is carried out, whereby the pseudo-matrix multiplication requires $||S||^2 \cdot N$ operations; therefore, the total number of operations is approximately $||S||^2 \cdot N \cdot n$, where n is the number of total pixels. Typically, in this paper, $n \geq 256^2$, but $N = 6$ and $||S|| \geq 80$. For a small image with $n = 256^2$, $||S||^2 \cdot N \geq 80^2 \cdot 6 = 38{,}400$, which is approximately equal to n; however, when n is large, $||S||^2 \cdot N$ is relatively negligible. In conclusion, the complexity of the proposed method is between $O(n)$ and $O(n^2)$; moreover, the computation speed of the proposed method is unsurprisingly slow since the corresponding computational burden is relatively large.

Acknowledgements

This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MEST) (NRF-2015R1A2A2A01004587).

References

[1] V. Sachnev, H. J. Kim, J. Nam, S. Suresh, and Y. Q. Shi, "Reversible watermarking algorithm using sorting and prediction," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 19, no. 7, pp. 989-999, Jul. 2009.

[2] J. Tian, "Reversible data embedding using a difference expansion," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 8, pp. 890-896, Aug. 2003.

[3] A.M. Alattar, "Reversible watermark using difference expansion of triplets," in *IEEE Int. Conf. Image Process.*, Catalonia, Spain, 2003, vol. 1, pp. 501-504.

[4] A. M. Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE Trans. Image Process.*, vol. 13, pp. 1147-1156, 2004.

[5] D. M. Thodi, and J. J. Rodriguez, "Expansion embedding techniques for reversible watermarking," *IEEE Trans. Image Process*, vol. 16, no. 3, pp. 721-730, 2007.

[6] M. J. Weinberger, G. Seroussi, and G. Sapiro, "The LOCO-I lossless image compression algorithm: Principles and standardization into JPEG-LS," *IEEE Trans. Image Process.*, vol. 9, no. 8, pp. 1309-1324, Aug. 2000.

[7] S. U. Kang, H. J. Hwang, and H. J. Kim, "Reversible watermark using an accurate predictor and sorter based on payload balancing," *ETRI J.*, vol. 34, no. 3, pp. 410-420, Jun. 2012.

[8] X. Li and M. T. Orchard, "Edge-directed prediction for lossless compression of natural images," *IEEE Trans. Image Process.*, vol. 10, no. 6, pp. 813-817, Jun. 2001.

[9] L.-J. Kau and Y.-P. Lin, "Adaptive lossless image coding using least squares optimization with edge-look-ahead," *IEEE Trans. Circuits Syst.*, vol. 52, no. 11, pp. 751-755, Nov. 2005.

[10] X. Wu and N. Memon, "Context-based, adaptive, lossless image coding," *IEEE Trans. Commun.*, vol. 45, no. 4, pp. 437-444, Apr.1997.

[11] X. Wu and K. Barthel, "Piesewise 2D autoregression for predictive image coding," in *Int. Conf. Image Process.*, vol. 3, Oct. 1998, pp. 901-905.

[12] M. H. Hansen and B. Yu, "Model selection and the principle of minimum description length," *J. Amer. Statist. Assoc.*, vol. 96, no. 6, pp. 746-774, 2001.

[13] X. Wu, G. Zhai, X. Yang, and W. Zhang, "Adaptive sequential prediction of multidimensional signals with applications to lossless image coding," *IEEE Trans. Image Process.*, vol. 20, no. 1, pp. 36-42, Jan. 2011.

[14] M. Chen, Z. Chen, X. Zeng, and Z. Xiong, "Model order selection in reversible image watermarking," *IEEE J. Sel. Top. Signal Process.*, vol. 4, no. 3, pp. 592-604, 2010.

[15] H. J. Hwang, H. J. Kim, V. Sachnev, and S. H. Joo, "Reversible watermarking method using optimal histogram pair shifting based on prediction and sorting," *KSII, Trans. Internet Inform. Syst.*, vol. 4, no. 4, pp. 655-670, Aug. 2010.

[16] G. Feng, Z. Qian, and N. Dai, "Reversible watermarking via extreme learning machine prediction," *Neurocomputing*, vol. 82 no. 1, pp. 62-68, Apr. 2012.

[17] L. Luo, Z. Chen, M. Chen, X. Zeng, and Z. Xiong, "Reversible image watermarking using interpolation technique," *IEEE Trans. Inf. Forensics Secur.*, vol. 5, no. 1, pp. 187-193, Dec.2010.

[18] B. Ou, X. Li, Y. Zhao, R. Ni, and Y. Q. Shi, "Pairwise prediction-error expansion for efficient reversible data hiding, *IEEE Trans. Image Process.*, vol. 22, no. 12, pp. 36-42, Dec. 2013

[19] I. C. Dragoi and D. Coltuc, "Local-prediction-based difference expansion reversible watermarking," *IEEE Trans. Image Process.,* vol. 23, no. 4, pp. 1779-1790, 2014.

[20] J. Wen, J. Lei, and Y. Wan, "Adaptive reversible data hiding through autoregression," In *Int. Conf. Inform. Science Tech.*, pp. 831-838. 2012.

[21] X. Qu, S. Kim, and H. J. Kim, "Reversible watermarking based on compensation," *J. Electr. Eng. Technol.*, vol. 10, no. 1, pp. 422-428, 2015.

[22] R. A. Groeneveld, and M. Glen, "Measuring skewness and kurtosis." *The Statistician*, pp. 391-399, 1984

Byeong Yong Lee He received a B.S. degree in mathematics from Konkuk University, Korea in 2006. He received an M.S degree in mathematics from Korea University, Korea, in 2010. He received a Ph.D. degree at the Graduate School of Information Security, Korea University. His research interests are reversible watermarking, divisible loads, and steganography

Hee Joon Hwang He received a B.S. degree from the Electric and Electronic Engineering department and an M.S. degree from the Graduate School of Information Management and Security at Korea University, Seoul, Korea, in 2008. He joined the Graduate School of Information Security, Korea University, Seoul, Korea, in 2007 where he is currently pursuing Ph.D. His research interests include multimedia security, reversible and robust watermarking, and steganography.

Hyoung Joong Kim He is currently with the Graduate School of Information Security, Korea University, Korea.