Reversible Image Watermarking Based on Integer-to-Integer Wavelet Transform

Sunil Lee, *Student Member, IEEE*, Chang D. Yoo, *Member, IEEE*, and Ton Kalker, *Fellow, IEEE*

***Abstract*—This paper proposes a high capacity reversible image watermarking scheme based on integer-to-integer wavelet transforms. The proposed scheme divides an input image into nonoverlapping blocks and embeds a watermark into the high-frequency wavelet coefficients of each block. The conditions to avoid both underflow and overflow in the spatial domain are derived for an arbitrary wavelet and block size. The payload to be embedded includes not only messages but also side information used to reconstruct the exact original image. To minimize the mean-squared distortion between the original and the watermarked images given a payload, the watermark is adaptively embedded into the image. The experimental results show that the proposed scheme achieves higher embedding capacity while maintaining distortion at a lower level than the existing reversible watermarking schemes.**

***Index Terms*—Adaptive embedding, image watermarking, integer-to-integer wavelet transform, reversible watermarking.**

I. Introduction

In digital watermarking, an imperceptible signal, referred to as a watermark, is embedded into multimedia data for various purposes, such as copyright protection, fingerprinting, authentication, etc. The embedding of the watermark usually introduces irreversible distortion, although it may be quite small, in the original data. For applications where the availability of original data are essential, irreversible degradation of the original data is not acceptable, and incurred distortions need to be removed. Examples of such applications include multimedia archives, military image processing, and medical image processing for electronic patient records (EPRs) [1]. In multimedia archives, a content provider may not want the original content to be distorted even though the distortion is imperceptible to most users, and it may be too costly in terms of storage space to store both the original and the watermarked versions. In military image processing, images are gathered at a very high cost and are usually subjected to further processing steps, such as extreme magnification. Any distortion may hinder accurate analysis of the image. In medical image processing, any modification to the original image may affect a doctor's diagnosis and lead to legal problems. Any complications that can occur when using a conventional watermarking scheme in the applications listed before can be resolved by using the reversible (lossless, invertible, erasable, etc.) watermarking scheme. Although the embedding distortion is inevitable even in reversible watermarking, it can be removed, and the original data can be reconstructed from the watermarked version. Another advantage of the reversible watermarking is that access to the original content can be controlled. In a conventional watermarking scenario, no one has access to the original content since the distortion due to the embedding of the watermark is not erasable. When the watermark is embedded in a reversible manner, an authorized person can access the original content by erasing the watermark while the watermarked content is available to everyone.

When the original content can be recovered from the watermarked content, the watermarking scheme is said to have the reversibility (invertibility) property. Note that the reversibility property can also be obtained by using standard (cryptographic) scrambling algorithms. However, the cryptographic approach completely obliterates any semantic understanding, which is not the case in reversible watermarking.

Recently, several reversible watermarking schemes have been proposed [2]–[22]. The concept of a reversible watermark was first introduced by Mintzer *et al.* [2]. The watermark that they embedded into an image was completely visible but could be removed since it was embedded in a reversible manner. Fridrich *et al.* extracted a vector which represented specific characteristics of pixel groups, compressed it without any loss, and embedded the watermark bits by appending it to the compressed vector [3]. Tian applied an integer Haar wavelet transform to an image and embedded the watermark into high-frequency coefficients by difference expansion (DE) [4]. Alattar extended Tian's scheme and applied the DE to triplets [5] and quads [6] of adjacent pixels for reversible embedding. He also proposed a reversible watermarking scheme using the DE of a generalized integer transform [7]. Kamstra and Heijmans improved Fridrich's [3] and Tian's methods [4] by sorting least-significant bits (LSBs) or pairs of pixels to be watermarked with respect to the heuristically obtained values [8]. The sorting locally improves the coding efficiency of the lossless compression, so that the overall performance is improved. Veen *et al.* applied the companding technique to reversibly embed a large amount of data into an audio signal [9]. Leest *et al.* applied Veen's method to image watermarking [10]. Vleeschouwer *et al.* proposed lossless watermarking based on a circular interpretation of bijective transformations [11]. Celik *et al.* generalized a well-known LSB-substitution technique and achieved

Manuscript received October 18, 2006; revised June 18, 2007. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Ingemar Cox.

S. Lee and C. D. Yoo are with the Division of Electrical Engineering, School of Electrical Engineering and Computer Science (EECS), Korea Advanced Institute of Science and Technology, Daejeon, 305-701, Korea (e-mail: sunillee@kaist.ac.kr: cdyoo@ee.kaist.ac.kr).

T. Kalker is with the Hewlett-Packard Labs, Multimedia Communications and Networking Department, Palo Alto, CA 94304 USA (e-mail: ton.kalker@ieee.org).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TIFS.2007.905146

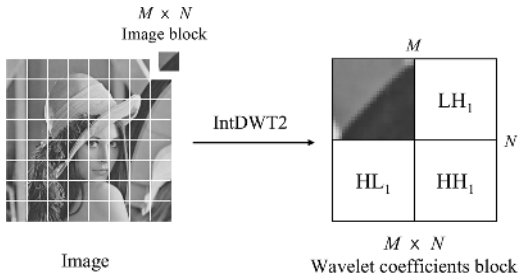

Fig. 1. Watermark is embedded into the high-frequency wavelet subbands HL_1, LH_1, and HH_1, which are the high–low, low–high, and high–high frequency subbands of each $M \times N$ image block. $\mathrm{IntDWT2}(\cdot)$ represents an 2-D integer-to-integer wavelet transform.

high capacity by using a prediction-based conditional entropy coder [12], [13]. Yang *et al.* proposed a reversible watermarking scheme based on an integer discrete cosine transform (DCT) transform [14]. Xuan *et al.* reversibly embedded the watermark bits into the middle- and high-frequency integer wavelet coefficients [15]–[17]. Coltuc *et al.* proposed a reversible watermarking scheme based on an integer transform for pairs of pixels [18] and generalized it for groups of an arbitrary number of pixels [19]. Kalker and Willems derived a theoretical bound on the embedding capacity for reversible data hiding [20]–[22].

In this paper, a novel reversible watermarking scheme with high embedding capacity for digital images is proposed. As shown in Fig. 1, an input image is divided into nonoverlapping blocks, and the watermark is embedded into the high-frequency wavelet coefficients of each block. To achieve the reversibility, invertible integer-to-integer wavelet transforms [23]–[25] are used, and the conditions to avoid underflow and overflow in the spatial domain are derived for arbitrary wavelets and block sizes. The watermark payload includes not only messages but also side information required to reconstruct the original image at the decoder. Contrary to most of the existing reversible watermarking schemes, the proposed scheme does not require lossless compression with high complexity, such as JBIG2 [26], since the embedding capacity is induced not from the lossless compression but from the statistical characteristic of the high-frequency wavelet coefficients. Furthermore, the block-based embedding makes the size of the side information that needs to be embedded small in proportion to the total embedding capacity. This advantage enables the proposed scheme to provide higher embedding capacity while maintaining distortion at a lower level than the existing schemes. Note that the reversible watermarking schemes proposed by Coltuc *et al.* [18], [19] do not require lossless compression either. However, the amount of side information in their methods is much larger than that of the proposed scheme. The proposed scheme also exploits adaptive embedding to improve the perceptual quality of the watermarked image. Rather than embedding watermark randomly or arbitrarily, the proposed method adaptively embeds watermark to give the highest quality for a given embedding capacity.

This paper is organized as follows. In Section II, invertible integer-to-integer wavelet transforms are briefly introduced. Sections III and IV, respectively, present the embedding and the decoding parts of the proposed reversible image watermarking scheme. Section V provides various experimental results. Finally, Section VI concludes the paper.

II. Invertible Integer-to-Integer Wavelet Transforms

Conventional wavelet transform is not applicable to the reversible watermarking scheme since it does not guarantee the reversibility. For example, suppose that an image block consisting of integer-valued pixels is transformed into a wavelet domain using a floating-point wavelet transform. If the values of the wavelet coefficients are changed during watermark embedding, the corresponding watermarked image block is no longer guaranteed to have integer values. Any truncation of the floating-point values of the pixels may result in a loss of information and may ultimately lead to the failure of the reversible watermarking systems, that is, the original image cannot be reconstructed from the watermarked image. Furthermore, the conventional wavelet transform is, in practice, implemented as a floating-point transform followed by a truncation or rounding since it is impossible to represent transform coefficients in their full accuracy: information can potentially be lost through forward and inverse transforms. To avoid this problem, an invertible integer-to-integer wavelet transform based on lifting is used in the proposed scheme [23]–[25]. It maps integers to integers and does not cause any loss of information through forward and inverse transforms.

According to [23], every wavelet or subband transform associated with finite length filters can be obtained as the Lazy wavelet followed by a finite number of primal and dual lifting steps and scaling (the Lazy wavelet essentially splits the signal into its even and odd indexed samples). By combining the lifting constructions with rounding off in a reversible way, a wavelet transform that maps integers to integers can be obtained. For example, the integer-to-integer wavelet transform that approximates Le Gall 5/3 filters is given by

$$\begin{aligned} d_{1,n} &= s_{0,2n+1} - \left\lfloor \frac{1}{2}(s_{0,2n} + s_{0,2n+2}) + \frac{1}{2} \right\rfloor, \\ s_{1,n} &= s_{0,2n} + \left\lfloor \frac{1}{4}(d_{1,n-1} + d_{1,n}) + \frac{1}{2} \right\rfloor \end{aligned} \tag{1}$$

where $s_{j,n}$ and $d_{j,n}$ are the nth low-frequency and high-frequency wavelet coefficients at the jth level, respectively [23]. When $j = 0$, $s_{0,n}$ represents the nth pixel value itself. The function $\lfloor x \rfloor$ rounds x to the nearest integer toward minus infinity. To make transforms nonexpansive, a symmetric extension compatible with invertible integer-to-integer wavelet transforms [27] is used.

III. Prediction of Underflow and Overflow

In the proposed scheme, a watermark is embedded into the wavelet coefficients using either the LSB-substitution or the bit-shifting. With the LSB-substitution technique, the watermark is embedded by replacing the LSB of the selected wavelet coefficient with the watermark bit as follows:

$$c^w = 2 \cdot \left\lfloor \frac{c}{2} \right\rfloor + w \tag{2}$$

where c, c^w, and w are the original and the watermarked wavelet coefficients and the watermark bit, respectively.

With the bit-shifting or, specifically, p-bit-shifting technique, the original wavelet coefficient c is multiplied by 2^p where p is a positive integer and a watermark w is embedded into its p LSBs as follows:

$$c^w = 2^p \cdot c + w \tag{3}$$

where $w = 2^0 \cdot w_0 + 2^1 \cdot w_1 + \cdots + 2^{p-1} \cdot w_{p-1}$ and $\{w_0, w_1, \ldots, w_{p-1}\}$ are a set of p watermark bits. Later, we will find out that the value of p is adaptively determined for each bit-shiftable block so that the perceptual distortion of the watermarked image is minimized.

When the watermark is embedded in the wavelet domain using either LSB-substitution or bit-shifting, underflow or overflow can occur in the spatial domain. That is, the pixel values obtained from the watermarked wavelet coefficients can either be smaller than the minimum pixel value $s_{\min}$ (e.g., $s_{\min} = 0$ for 8-bit grayscale image) or be greater than the maximum value $s_{\max}$ (e.g., $s_{\max} = 255$ for an 8-bit grayscale image). Since the reversibility is lost when underflow or overflow occurs, it must be predicted prior to the watermark embedding. In this paper, underflow and overflow are predicted by identifying the LSB-changeable and bit-shiftable image blocks. These blocks can be defined as follows.

Definition 1: An image block is said to be LSB-changeable when a watermark bitstream can be embedded into the LSBs of its high-frequency wavelet coefficients using the LSB-substitution without any underflow or overflow in the spatial domain.

Definition 2: An image block is said to be bit-shiftable or, specifically, p-bit-shiftable, when a watermark bitstream can be embedded into its high-frequency wavelet coefficients using the bit-shifting by p bits without any underflow or overflow in the spatial domain.

The LSB-changeability and the bit-shiftability are not exclusive of each other. Therefore, a block can be both LSB-changeable and bit-shiftable. In Sections III-A and B, the conditions to identify LSB-changeable and bit-shiftable blocks are derived for arbitrary wavelets and block sizes.

A. Derivation of Condition to Avoid Underflow and Overflow

In this subsection, the condition to avoid underflow and overflow is derived. The block-wise embedding using wavelet transforms and the notations that will be henceforth used are described in Fig. 2. First, an $M \times N$ pixel block $\mathbf{S}$ is transformed into a block of $M \times N$ wavelet coefficients $\mathbf{C}$ using the 2-D nonexpansive integer-to-integer wavelet transform $\mathrm{IntDWT2}(\cdot)$. Next, a block of modified wavelet coefficients $\mathbf{C}_M$ is obtained either by setting the LSBs of the chosen wavelet coefficients to zero or by applying bit-shifting to the chosen coefficients in $\mathbf{C}$. The modified pixel block $\mathbf{S}_M$ is obtained by applying the 2-D inverse floating-point $(\mathrm{fDWT2}^{-1}(\cdot))$ wavelet transform to $\mathbf{C}_M$. By adding a watermark bit block $\mathbf{W}$ to $\mathbf{C}_M$, a block of watermarked wavelet coefficients $\mathbf{C}_W$ is obtained. Then, $\mathbf{S}_{WF}$ and $\mathbf{S}_{WI}$ are obtained by applying $\mathrm{fDWT2}^{-1}(\cdot)$ and $\mathrm{IntDWT2}^{-1}(\cdot)$ to $\mathbf{C}_W$, respectively. The embedding error $\mathbf{E}_W$ is obtained by applying $\mathrm{fDWT2}^{-1}(\cdot)$ to $\mathbf{W}$. Henceforth,

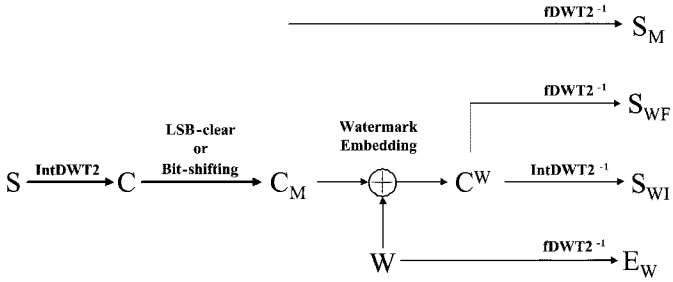

Fig. 2. Forward and inverse wavelet transform and watermark embedding.

$\mathbf{A}(m,n)$ represents an element of the block $\mathbf{A}$ in the mth row and the nth column.

When a floating-point wavelet transform is used, underflow and overflow that can occur as a result of embedding a watermark in the wavelet domain can be easily predicted by exploiting the linearity of the transform. Then, the watermarked block $\mathbf{S}_{WF}$ in Fig. 2 is given by

$$\begin{aligned}\mathbf{S}_{WF} &= \mathrm{fDWT2}^{-1}(\mathbf{C}_W) \\ &= \mathrm{fDWT2}^{-1}(\mathbf{C}_M + \mathbf{W}) \\ &= \mathrm{fDWT2}^{-1}(\mathbf{C}_M) + \mathrm{fDWT2}^{-1}(\mathbf{W}) \\ &= \mathbf{S}_M + \mathbf{E}_W. \end{aligned} \tag{4}$$

Given an image block $\mathbf{S}$, $\mathbf{S}_M$ in (4) can be straightforwardly determined. Thus, whether underflow or overflow occurs is solely dependent on the error $\mathbf{E}_W$ caused by the embedded watermark $\mathbf{W}$. For an arbitrary watermark bit block $\mathbf{W}$, two error matrices $\mathbf{E}_{WP}$ and $\mathbf{E}_{WN}$, whose elements represent limits of maximum positive and negative errors caused by the embedding can be obtained by exploiting the linearity of the conventional wavelet transform. The matrices $\mathbf{E}_{WP}$ and $\mathbf{E}_{WN}$ are given by

$$\mathbf{E}_{WP} = \sum_{i,j \in (HL_1 \cup LH_1 \cup HH_1)} \frac{1}{2} \left\{ \mathbf{Q}_{ij} + \mathrm{ABS}(\mathbf{Q}_{ij}) \right\}, \tag{5}$$

$$\mathbf{E}_{WN} = \sum_{i,j \in (HL_1 \cup LH_1 \cup HH_1)} \frac{1}{2} \left\{ \mathbf{Q}_{ij} - \mathrm{ABS}(\mathbf{Q}_{ij}) \right\} \tag{6}$$

where $\mathbf{Q}_{ij} = \mathrm{fDWT2}^{-1}(\mathbf{O}_{ij})$ and the matrix $\mathbf{O}_{ij}$ has only one nonzero element of value 1 in the ith row and the jth column. The wavelet subbands HL_1, LH_1, and HH_1 are the high–low, low–high, and high–high frequency subbands of the input image as described in Fig. 1, and the function $\mathrm{ABS}(\mathbf{A})$ takes an absolute value of each element of a matrix $\mathbf{A}$. Since the elements of $\mathbf{E}_W$ satisfy the inequality $\mathbf{E}_{WN}(m,n) \le \mathbf{E}_W(m,n) \le \mathbf{E}_{WP}(m,n)$, neither underflow nor overflow will occur in $\mathbf{S}$ for any watermark block $\mathbf{W}$ if

$$s_{\min} - \mathbf{E}_{WN}(m,n) \le \mathbf{S}_M(m,n) \le s_{\max} - \mathbf{E}_{WP}(m,n) \tag{7}$$

for $0 \le m < M$, $0 \le n < N$.

In the proposed watermarking scheme, integer-to-integer wavelet transforms [23] are used to achieve reversibility. Therefore, the watermarked image block that we obtain is not $\mathbf{S}_{WF}$ but $\mathbf{S}_{WI} = \mathrm{IntDWT2}^{-1}(\mathbf{C}_W)$. Since integer-to-integer wavelet transforms involve the truncations of wavelet coefficients during the lifting steps, roundoff error is inevitable. Let

$\mathbf{E}_R$ be the roundoff error matrix. Then, the watermarked image block $\mathbf{S}_{WI}$ is given by

$$\begin{aligned}\mathbf{S}_{WI} &= \mathrm{IntDWT2}^{-1}(\mathbf{C}_W) \\ &= \mathrm{IntDWT2}^{-1}(\mathbf{C}_M + \mathbf{W}) \\ &= \mathrm{fDWT2}^{-1}(\mathbf{C}_M + \mathbf{W}) + \mathbf{E}_R \\ &= \mathbf{S}_M + \mathbf{E}_W + \mathbf{E}_R. \end{aligned} \tag{8}$$

As represented by $\mathbf{E}_{WP}$ and $\mathbf{E}_{WN}$, roundoff error matrices $\mathbf{E}_{RP}$ and $\mathbf{E}_{RN}$, whose elements represent limits of maximum positive and negative roundoff errors can be defined. Now, an image block $\mathbf{S}$ can be said to be LSB-changeable or bit-shiftable for any watermark block $\mathbf{W}$ if the following inequality is satisfied:

$$\begin{aligned} s_{\min} &- \mathbf{E}_{WN}(m,n) - \mathbf{E}_{RN}(m,n) \\ &\le \mathbf{S}_M(m,n) \\ &\le s_{\max} - \mathbf{E}_{WP}(m,n) - \mathbf{E}_{RP}(m,n) \end{aligned} \tag{9}$$

for $0 \le m < M, 0 \le n < N$.

B. Estimation of Roundoff Error

Now the remaining problem is to estimate the roundoff error in (9). For simplicity, the 1-D case is first analyzed. Let $\mathbf{e}^R = [e_0^R \quad e_1^R \quad \dots \quad e_{N-1}^R]$ be the N-dimensional roundoff error vector. The error vector can be subsampled into even-indexed vector $(\mathbf{e}_0^R)$ and odd-indexed vector $(\mathbf{e}_1^R)$. The relationship between the z-transforms of $\mathbf{e}^R$, $\mathbf{e}_0^R$, and $\mathbf{e}_1^R$ denoted, respectively, as $E^R(z)$, $E_0^R(z)$, and $E_1^R(z)$ are given by

$$E^R(z) = [1 \quad z^{-1}] \begin{bmatrix} E_0^R(z^2) \\ E_1^R(z^2) \end{bmatrix}. \tag{10}$$

Fig. 3 shows the polyphase representation of wavelet transforms. Let (h, g) be a pair of two synthesis filters h (low-pass) and g (high-pass), and $H(z)$ and $G(z)$ be their z-transforms. Given a complementary filter pair (h, g), Laurent polynomial $S_i(z)$ and $T_i(z)$ will always exist for $1 \le i \le m$ and a nonzero constant K so that

$$\begin{aligned} P(z) &= \begin{bmatrix} H_0(z) & G_0(z) \\ H_1(z) & G_1(z) \end{bmatrix} \\ &= \prod_{i=1}^{m} \begin{bmatrix} 1 & S_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ T_i(z) & 1 \end{bmatrix} \begin{bmatrix} K & 0 \\ 0 & \frac{1}{K} \end{bmatrix} \end{aligned} \tag{11}$$

where $P(z)$ is the polyphase synthesis matrix [24], [28], $H(z) = H_0(z^2) + z^{-1}H_1(z^2)$, and $G(z) = G_0(z^2) + z^{-1}G_1(z^2)$. This means that every finite filter wavelet transform can be obtained by starting with the Lazy wavelet followed by m lifting and dual lifting steps followed with a scaling [24]. Using (11), the roundoff errors $E_0^R(z)$ and $E_1^R(z)$ in (10) for $m \ge 2$ can be derived as

$$\begin{aligned} \begin{bmatrix} E_0^R(z) \\ E_1^R(z) \end{bmatrix} &= \begin{bmatrix} 1 & S_1(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} U_{0,1}(z) \\ U_{1,1}(z) \end{bmatrix} \\ &\quad + \sum_{i=2}^{m} \left\{ \prod_{j=1}^{i-1} \begin{bmatrix} 1 & S_j(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ T_j(z) & 1 \end{bmatrix} \right. \\ &\quad \left. \times \begin{bmatrix} 1 & S_i(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} U_{0,i}(z) \\ U_{1,i}(z) \end{bmatrix} \right\} \end{aligned} \tag{12}$$

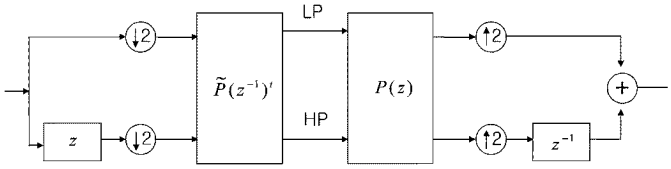

Fig. 3. Polyphase representation of wavelet transform: first subsample into even and odd, then apply the dual polyphase matrix. For the inverse transform: first apply the polyphase matrix and then join even and odd [24].

where $U_{0,i}(z)$ and $U_{1,i}(z)$ are the roundoff errors of low-pass and high-pass bands at the ith lifting step, respectively, and K is assumed to be 1. When $m = 1$, the roundoff errors are simply given as

$$\begin{bmatrix} E_0^R(z) \\ E_1^R(z) \end{bmatrix} = \begin{bmatrix} 1 & S_1(z) \\ 0 & 1 \end{bmatrix} \begin{bmatrix} U_{0,1}(z) \\ U_{1,1}(z) \end{bmatrix}. \tag{13}$$

The elements of the roundoff errors $\mathbf{u}_{0,i}$ and $\mathbf{u}_{1,i}$ have the value between $-(1/2)$ and $(1/2)$. By exploiting this property and the linearity of (12) and (13), the roundoff error vectors $\mathbf{e}^{RP}$ and $\mathbf{e}^{RN}$, whose elements represent limits of maximum positive and negative roundoff errors, can be easily obtained. For example, $\mathbf{e}^{RP}$ and $\mathbf{e}^{RN}$ for the integer-to-integer wavelet transform that approximates Le Gall 5/3 filter is given by

$$\mathbf{e}^{RP} = -\mathbf{e}^{RN} = [0.5 \quad 1.0 \quad 0.5 \quad 1.0 \quad 0.5 \quad 1.0 \quad 0.5 \quad 1.0] \tag{14}$$

when the dimension of the roundoff error vector is $N = 8$.

Since 2-D integer-to-integer wavelet transforms are used in the proposed scheme, the above procedure should be extended to the 2-D case. A 2-D wavelet transform is implemented by applying a 1-D transform to each row of an input matrix and then to each column of the matrix whose rows are the results of the first transform. Therefore, the roundoff error given in (12) or (13) becomes the input to the 1-D integer-to-integer wavelet transform, thus total roundoff error is the sum of the filtered error and the roundoff error that occurs in the second transform. For example, $\mathbf{E}_{RP}$ and $\mathbf{E}_{RN}$ for the integer-to-integer wavelet transform that approximates Le Gall 5/3 filter is given by

$$\begin{aligned} \mathbf{E}_{RP} &= -\mathbf{E}_{RN} \\ &= \begin{bmatrix} 1.25 & 2 & 1.25 & 2 & 1.25 & 2 & 1.25 & 2 \\ 2 & 3 & 2 & 3 & 2 & 3 & 2 & 3 \\ 1.25 & 2 & 1.25 & 2 & 1.25 & 2 & 1.25 & 2 \\ 2 & 3 & 2 & 3 & 2 & 3 & 2 & 3 \\ 1.25 & 2 & 1.25 & 2 & 1.25 & 2 & 1.25 & 2 \\ 2 & 3 & 2 & 3 & 2 & 3 & 2 & 3 \\ 1.25 & 2 & 1.25 & 2 & 1.25 & 2 & 1.25 & 2 \\ 2 & 3 & 2 & 3 & 2 & 3 & 2 & 3 \end{bmatrix}. \end{aligned} \tag{15}$$

IV. Proposed Reversible Embedding Algorithm

Given an $X \times Y$ original image to be watermarked, a set of message bits $\mathcal{B}_m$ to be embedded, the block size $(M \times N)$, the wavelet to be used for forward and inverse transforms, and the maximum number of bits allowed to be shifted $p_{\max}$,

the watermarks are reversibly embedded as explained in Sections IV-A–E.

A. Reservation of LSB-Changeable Blocks for the Location Map

The location map $\mathbf{L}$ is a binary matrix that indicates which blocks are watermarked. It is a part of the side information used by the decoder to retrieve the message bits and to reconstruct the original image. Since one bit is assigned to each block, the size of the location map is $(X/M) \times (Y/N)$.

To extract the watermark, the decoder has to retrieve the location map first. However, if the location map is embedded using bit-shifting, it cannot be retrieved since a bit-shiftable block can turn into a non-bit-shiftable block as a result of the watermark embedding, which means it would need its own location map to be identified. Therefore, the location map should be embedded into blocks that retains its characteristics even after the embedding. In the proposed scheme, the following property is exploited to embed the location map.

1) LSB-Changeable: All LSB-substitution blocks remain LSB-changeable after the watermark is embedded using the LSB-substitution.

Using this property, the location map can be embedded into a set of LSB-changeable blocks with LSB-substitution. Since the LSB-changeability of the chosen blocks is maintained regardless of watermark embedding, the decoder can identify the selected blocks and retrieve the location map.

The embedder first finds a number of LSB-changeable blocks with enough capacity to hold the location map in a predefined order. Depending on the application, the selection of the LSB-changeable blocks can be based on a secret key shared with the decoder. Since the size of the location map is $(X/M) \times (Y/N)$ and the maximum number of bits that can be embedded into one block is $(3MN/4)$, the minimum number of LSB-changeable blocks necessary for the embedding of the location map is $\lceil (XY/MN) / (3MN/4) \rceil$ where the function $\lceil x \rceil$ rounds x to the nearest integer toward plus infinity. The chosen blocks are reserved for the embedding of the location map, and not used for other purposes.

B. Distribution of Payload for Adaptive Embedding

Even though the original image can be exactly reconstructed from the watermarked image, the quality of the watermarked image is still important. In this paper, the peak signal-to-noise ratio (PSNR) is used as a measure for quantifying the quality of an image. Therefore, the watermark needs to be embedded such that the mean-squared distortion between the original and the watermarked image is minimized. In order to achieve this goal, the bit-shiftable blocks in which the watermark embedding does not introduce severe distortion need to be found and used for the embedding. Since it is not known which specific watermark bits are embedded into each bit-shiftable block at this moment, the distortion introduced by the watermark embedding is estimated by

$$\mathrm{MSE}(\mathbf{S}, \mathbf{S}_{WI}) \approx \mathrm{MSE}(\mathbf{S}, \mathbf{S}_{BI}) \tag{16}$$

where the function $\mathrm{MSE}(\mathbf{S}_1, \mathbf{S}_2)$ calculates the mean-square error between two image blocks $\mathbf{S}_1$ and $\mathbf{S}_2$. The original and watermarked image blocks $\mathbf{S}$ and $\mathbf{S}_{WI}$ are as shown in Fig. 2. The bit-shifted pixel block $\mathbf{S}_{BI}$ is obtained by applying bit-shifting to the chosen wavelet coefficients of the original wavelet coefficients block $\mathbf{C}$, and then performing 2-D inverse integer-to-integer wavelet transform to it. This approximation is reasonable since the distortion introduced by the bit-shifting is much higher than the distortion introduced by the watermark embedding in most blocks. The detailed procedure to distribute the given payload so that the mean-squared distortion is minimized will be described below. Here, $\mathbf{S}_{ij}$ denotes the block in the ith row and the jth column where $0 \le i < (X/M)$ and $0 \le i < (Y/N)$.

1) Initialization.
   a) A matrix $\mathbf{P}$ is set to an $(X/M) \times (Y/N)$ zero matrix. The induced capacity R is set to zero.
   b) Among all of the blocks except those reserved for the embedding of the location map, the 1-bit-shiftable blocks are identified using (9). Let $\mathcal{S}$ be the set of the identified bit-shiftable blocks, and $\mathcal{I}$ be the set of its indices such that $\mathcal{I} = \{(i,j) \mid \mathbf{S}_{ij} \in \mathcal{S}\}$..
2) Iteration.
   a) $\{\mathbf{P}(i,j)+1\}$-bit-shifting is applied to all blocks in $\mathcal{S}$, where $(i,j) \in \mathcal{I}$. Then, $(\mathbf{S}_{ij})_{BI}$ is obtained for each block in $\mathcal{S}$ by taking the inverse integer-to-integer wavelet transform to the bit-shifted wavelet coefficients.
   b) The block $\mathbf{S}_{\hat{i}\hat{j}}$ in which the watermark embedding may lead to the lowest mean-squared error among all bit-shiftable blocks is searched by

$$(\hat{i}, \hat{j}) = \arg\min_{(i,j)\in\mathcal{I}} \mathrm{MSE}\{\mathbf{S}_{ij}, (\mathbf{S}_{ij})_{BI}\}. \tag{17}$$

   c) The induced embedding capacity R is incremented by $(3MN/4 - \lceil \log_2(p_{\max}) \rceil)$ if $\mathbf{P}(\hat{i}, \hat{j}) = 0$; otherwise, by $(3MN/4)$. The reserved $\lceil \log_2(p_{\max}) \rceil$ bits are used to embed the element of the matrix $\mathbf{P}$ which indicates how many bits are shifted in the corresponding block.
   d) The value of $\mathbf{P}(\hat{i}, \hat{j})$ is incremented by 1.
   e) If $\mathbf{P}(\hat{i}, \hat{j}) = p_{\max}$ or the block $\mathbf{S}_{\hat{i}\hat{j}}$ is not $\{\mathbf{P}(\hat{i}, \hat{j}) + 1\}$-bit-shiftable, the block $\mathbf{S}_{\hat{i}\hat{j}}$ and its index $(\hat{i}, \hat{j})$ are deleted from $\mathcal{S}$ and $\mathcal{I}$, respectively.
   f) If the induced capacity R satisfies the following inequality, the iteration is terminated

$$R \ge |\mathcal{B}_m| + |\mathcal{B}_o| = |\mathcal{B}_m| + |\mathcal{B}_l| = |\mathcal{B}_m| + \frac{(XY)}{(MN)} \tag{18}$$

   where $\mathcal{B}_l$ is a set of bits in the location map $\mathbf{L}$, $\mathcal{B}_o$ is a set of original LSBs replaced during the embedding of the location map via the LSB-substitution, and $|\cdot|$ is the cardinality of a set. If (18) is not satisfied, go to a) and repeat the iteration.

As a result of the procedure described before, the matrix $\mathbf{P}$ is obtained. It is used to adaptively embed the watermark into the image in a reversible manner.

C. Construction and Embedding of Location Map

The $(X/M) \times (Y/N)$ location map $\mathbf{L}$ is constructed by

$$\mathbf{L}(i,j) = \begin{cases} 1, & \text{if } \mathbf{P}(i,j) > 0 \\ 0, & \text{otherwise} \end{cases} \tag{19}$$

where $0 \le i < (X/M)$ and $0 \le j < (Y/N)$. Then, $(XY)/(MN)$ bits in the location map $\mathbf{L}$ are sequentially embedded into the reserved LSB-changeable blocks by the LSB-substitution. The original LSBs replaced during the embedding of the location map are collected into the set $\mathcal{B}_o$. For reversibility, they are also embedded into the image.

D. Embedding of Message and Remaining Side Information

The remaining payload to be embedded includes the message bits, the original LSBs, and the matrix $\mathbf{P}$, whose elements indicate how many bits are shifted in each block. First, a set of the bit-shiftable blocks $\mathcal{S}_b = \{\mathbf{S}(i,j) \mid \mathbf{P}(i,j) > 0\}$ into which the watermark is embedded is found using the matrix $\mathbf{P}$ that is constructed following the procedures described before. Next, $\mathbf{P}(i,j)$-bit-shifting is applied to each block in $\mathcal{S}_b$. As a result of the bit-shifting $\{\mathbf{P}(i,j) \cdot (3MN/4)\}$ bits are, respectively, available for embedding each block in $\mathcal{S}_b$. For each block in $\mathcal{S}_b$, first $\mathbf{P}(i,j)$ is converted to the binary number of the length $\lceil \log_2(p_{\max}) \rceil$ and embedded. In the remaining $(\mathbf{P}(i,j) \cdot (3MN/4) - \lceil \log_2(p_{\max}) \rceil)$ bits available for embedding in each block belonging to $\mathcal{S}_b$, the message bits and the original LSBs are sequentially embedded.

E. Reversible Watermark Embedding for Color Images

The proposed reversible watermarking scheme can be easily applied to color images. First, given an image and block size, wavelet, and $p_{\max}$, the maximum embedding capacity is estimated for each color channel (e.g., R_r (red channel), R_g (green channel), and R_b (blue channel) bits per pixel (bpp), respectively). Next, the set of the message bits $\mathcal{B}_m$ is divided into three subsets $\mathcal{B}_{mr}$, $\mathcal{B}_{mg}$, and $\mathcal{B}_{mb}$, whose cardinalities are, respectively, given by

$$|\mathcal{B}_{mr}| = \left\lfloor \frac{|\mathcal{B}_m| \cdot R_r}{(R_r + R_g + R_b)} \right\rfloor, \tag{20}$$

$$|\mathcal{B}_{mg}| = \left\lfloor \frac{|\mathcal{B}_m| \cdot R_g}{(R_r + R_g + R_b)} \right\rfloor, \tag{21}$$

$$|\mathcal{B}_{mb}| = |\mathcal{B}_m| - (|\mathcal{B}_{mr}| + |\mathcal{B}_{mg}|) \tag{22}$$

where $\mathcal{B}_{mr} \cap \mathcal{B}_{mg} = \mathcal{B}_{mg} \cap \mathcal{B}_{mb} = \mathcal{B}_{mb} \cap \mathcal{B}_{mr} = \emptyset$ and $\mathcal{B}_{mr} \cup \mathcal{B}_{mg} \cup \mathcal{B}_{mg} = \mathcal{B}_m$. Then, the three subsets of the message bits $\mathcal{B}_{mr}$, $\mathcal{B}_{mg}$, and $\mathcal{B}_{mb}$ are independently embedded into each color channel using the reversible watermarking scheme described before. This procedure can also be applied to the color images with other color coordinates (e.g., YUV [29]), by dividing message bits into subsets and embedding them into each color channel independently.

V. Decoding Algorithm

First, the watermarked image is divided into nonoverlapping blocks with dimension $M \times N$, and each block is transformed using the same wavelet used in the embedding procedure. Next, LSB-changeable blocks are searched in a predefined order. In some applications, the order of the search is based on a secret key shared with the embedder. When the LSB-changeable blocks are found, the location map is retrieved by collecting the LSBs of the high-frequency wavelet coefficients in those blocks. Based on the location map, the blocks into which the watermark is embedded are sequentially searched, and the number of embedded bits is calculated by decoding the value of p in each block. Finally, the payload that includes the original LSBs and the message bits is extracted from the blocks indicated by the location map. Using the original LSBs and the location map, the original image block can be reconstructed exactly. For a color image that is watermarked, the decoding procedure given before is repeated for each color channel.

VI. Experimental Results

In all experiments, 512×512 RGB and grayscale images from the USC–SIPI database [30] were used as test images, and the integer-to-integer wavelet transform based on Le Gall 5/3 filters is used. The embedding capacity and the quality of the watermarked image are represented by bit per pixel (bpp) and PSNR in decibels, respectively. Note that message bits are only counted when calculating the embedding capacity, that is, the embedding capacity obtained in the experiments is $|\mathcal{B}_m|/(XY)$. The message bits are randomly generated using the MATLAB function **rand**(). Considering the quality of the watermarked image, the value of $p_{\max}$ is set to 2, that is, only 2-bit-shifting is allowed.

A. Performance for Natural Images With Various Block Sizes

The proposed scheme is applied to various natural images F-16 (Airplane), Lena, Barbara, Peppers, Fishing boat, and Baboon (Mandrill). Fig. 4 shows the quality of the watermarked images at various embedding capacities up to 0.50 bpp. In this experiment, the block size was set to 16×16. As shown in Fig. 4, the proposed scheme achieves high embedding capacity with low distortion. However, the capacity-distortion performance depends on the characteristic of each image. High embedding capacity could be achieved with low distortion for images that contain a large amount of low-frequency components (e.g., F-16 and Lena). On the other hand, for images that include a large amount of high-frequency components (e.g., Baboon), much lower embedding capacity was obtained at the same PSNR. Figs. 5 and 6 show the original image and the examples of the watermarked images at various embedding capacities for grayscale Lena and Barbara images, respectively. As shown in the figures, the perceptual quality of the watermarked image is quite good at low and moderate embedding capacities and is acceptable even at very high embedding capacity around 1.0 bpp. Since the proposed reversible scheme embeds the watermark into the high-frequency wavelet coefficients, the high-frequency components of the image are strengthened and the watermarked image seems to be filtered with a sharpening mask. However the sharpening effect introduced by the watermark embedding is not perceptually annoying, even at low PSNRs.

The proposed scheme is also applied to color images. Fig. 7 shows the PSNRs of the watermarked color images at various embedding capacities up to 1.0 bpp. The block size is set to

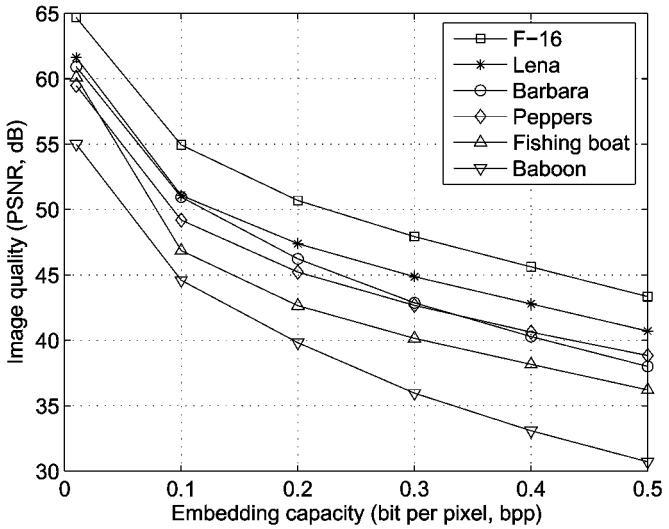

Fig. 4. Comparison of embedding capacity in bpp versus distortion in PSNR for various grayscale images.

Fig. 5. Original and watermarked grayscale Lena images. (a) Original. (b) 44.87 dB with 0.30 bpp. (c) 37.82 dB with 0.65 bpp. (d) 32.92 dB with 1.00 bpp.

Fig. 6. Original and watermarked grayscale Barbara images. (a) Original. (b) 46.23 dB with 0.20 bpp. (c) 38.02 dB with 0.50 bpp. (d) 31.78 dB with 0.80 bpp.

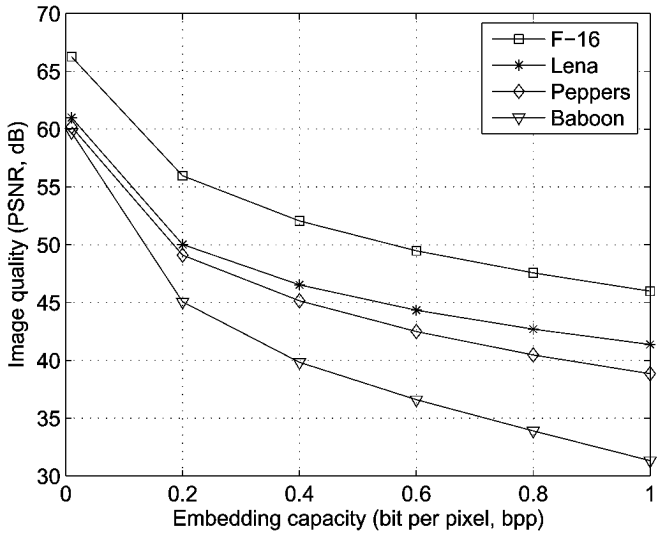

Fig. 7. Comparison of embedding capacity in bpp versus distortion in PSNR for various color images.

16×16 as in the previous experiments. The result in the figure shows that the capacity-distortion performance for color images also depends on the characteristic of each individual image. As in the case of grayscale images, higher embedding capacity can be achieved at the same PSNR when the input image has more low-frequency components (e.g., color F-16). Figs. 8 and 9 show the original image and the examples of the watermarked images at various embedding capacities for colo Lena and Baboon images, respectively. Even though the sharpening effect is observed as in grayscale images, the perceptual quality of the watermarked image is quite good at all embedding capacities.

In the proposed scheme, the block size can be easily adjusted to the specific requirements of an application. Fig. 10 shows how the performance of the proposed scheme varies with different block sizes: 4×4, 8×8, 16×16, and 32×32. As shown in the figure, the performance of the proposed scheme is degraded when the block size is too small (4×4) or too large (32×32). The smaller the block size is, the larger the amount of the side information becomes. For example, 0.1250 bpp is necessary to embed the location map and the original LSBs when the block size is 4×4, while only 0.0078 bpp is required for 16×16 block size. Since 0.1250 bpp is too high of a capacity to hold, the performance degrades when the block size is too small. When the block size is large, the amount of the side information that needs to be embedded is very small. However, the block size that is too large makes the adaptive embedding less useful especially

Fig. 8. Original and watermarked color Lena images. (a) Original. (b) 44.34 dB with 0.60 bpp. (c) 39.62 dB with 1.30 bpp. (d) 34.90 dB with 2.00 bpp.

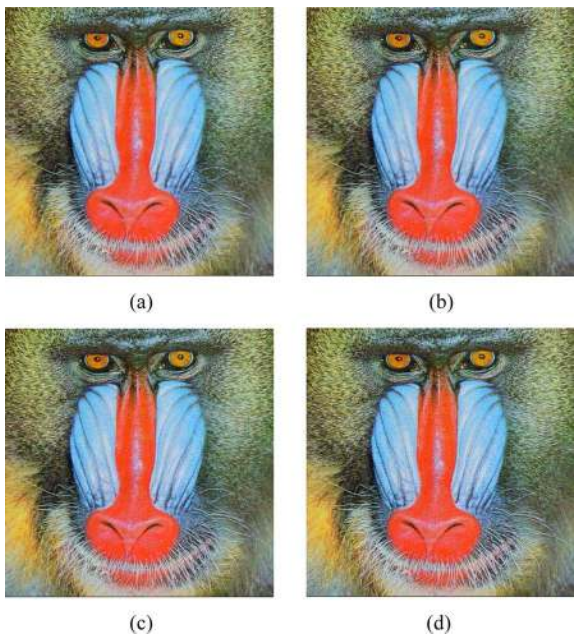

Fig. 9. Original and watermarked color Baboon images. (a) Original. (b) 38.56 dB with 0.50 bpp. (c) 32.42 dB with 1.00 bpp. (d) 27.77 dB with 1.50 bpp.

for high embedding capacity, and this degrades the performance of the proposed scheme.

B. Comparison of Performance With Other Schemes in the Literature

The performance of the proposed scheme is compared with the existing reversible schemes. Fig. 11 shows the comparison of the embedding capacity in bpp versus distortion in PSNR of the proposed scheme with that of the existing reversible schemes for the grayscale Lena image. As shown in the figure, the RS scheme proposed by Fridrich *et al.* [3] has low embedding capacity (≤ 0.15 bpp) compared to the others. The integer DWT-based reversible embedding scheme proposed by Xuan *et al.* [15] does not provide an option for tradeoff between the embedding capacity and the perceptual quality, that is, for a given host image, the achievable capacity and image quality is fixed and cannot be adjusted. In the other schemes, for example, the difference-expansion (DE) schemes proposed by Tian [4] and its variation [5], the integer DCT-based scheme proposed by Yang *et al.* [14], and the generalized LSB (G-LSB) scheme proposed by Celik *et al.* [12], the tradeoff between capacity and image quality is possible and relatively high embedding capacity can be achieved. However, as shown in the figure, the proposed scheme achieves higher embedding capacity with lower distortion than the other schemes. Note that the performance of the scheme proposed by Xuan *et al.* [17] is comparable to that of the proposed scheme for the grayscale Lena image. However they did not mention whether the book keeping information of histogram modification and the replaced original wavelet coefficients were included in the payload. The performance of the scheme in [17] depends on how well this information can be losslessly compressed. Fig. 12 shows the results for the grayscale Barbara image. In this experiment, the performance of the proposed scheme is compared with only competitive schemes. As shown in the figure, the proposed scheme achieves higher embedding capacity at all PSNRs than the other schemes. The evaluation results in Figs. 11 and 12 show that the proposed scheme outperforms the existing reversible schemes.

VII. Conclusion

In this paper, a reversible image watermarking scheme based on integer-to-integer wavelet transforms is proposed. In the proposed scheme, an input image is divided into a number of blocks, and a watermark is embedded into the high-frequency wavelet coefficients of each block by LSB-substitution or bit-shifting. The original image can be exactly reconstructed at the decoder since the side information for achieving the reversibility is also embedded in the image while avoiding the underflow and overflow. Experimental results show that the proposed scheme achieves higher embedding capacity at lower distortion than other existing reversible schemes. Further work is to obtain the theoretical bound of the embedding capacity for the reversible image watermarking using the results of Kalker and Willems [20]–[22], and to propose an improved reversible watermarking scheme whose embedding capacity is close to the theoretical bound.

References

[1] G. Coatrieux, H. Maitre, B. Sankur, Y. Rolland, and R. Collorec, "Relevance of watermarking in medical images," in *Proc. Workshop of the Int. Telemedical Information Soc., IEEE EMBS Int. Conf. Information Technology Applications in Biomedicine*, Nov. 2000, pp. 250–255.

[2] F. Mintzer, J. Lotspiech, and N. Morimoto, "Safeguarding digital library contents and users: Digital watermarking," *D-Lib Mag.*, Dec. 1997.

[3] J. Fridrich, M. Golijan, and R. Du, "Lossless data embedding for all image formats," presented at the SPIE, Security and Watermarking of Multimedia Contents, San Jose, CA, 2002.

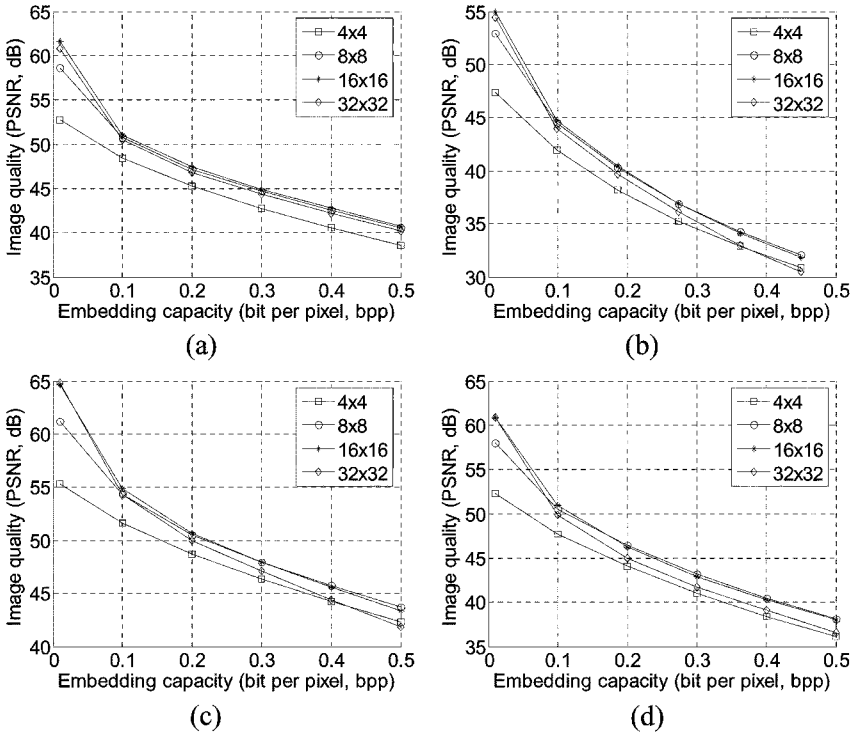

Fig. 10. Comparison of embedding capacity in bpp versus distortion in PSNR for grayscale images with various block sizes, 4×4, 8×8, 16×16, and 32×32: (a) `Lena`, (b) `Baboon`, (c) `F-16` (`Airplane`), and (d) `Barbara`.

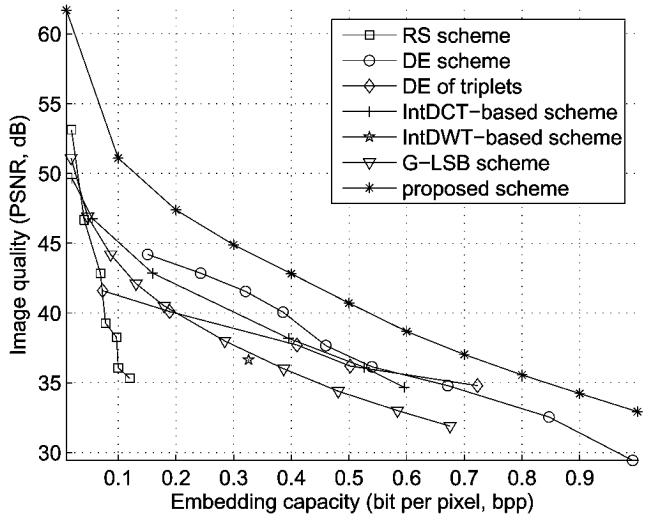

Fig. 11. Comparison of embedding capacity in bpp versus distortion in PSNR with existing reversible schemes—RS [3], DE [4], DE of triplets [5], integer DCT(IntDCT)-based [14], integer DWT(IntDWT)-based [15], and generalized-LSB [12] schemes. The test image is the grayscale `Lena`.

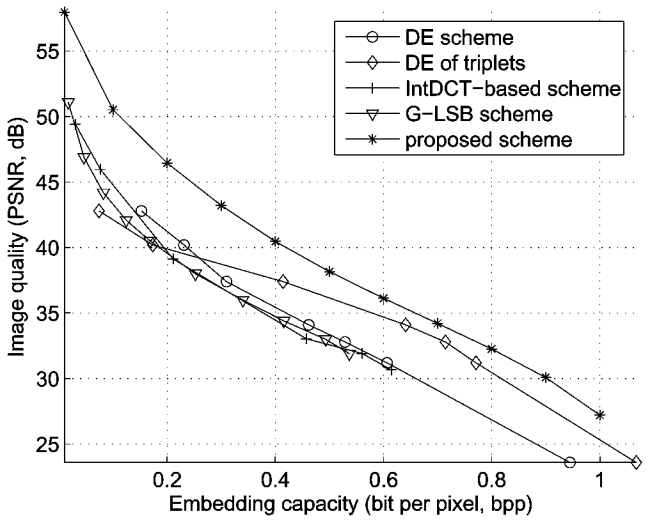

Fig. 12. Comparison of embedding capacity in bpp versus distortion in PSNR with existing reversible schemes—DE [4], DE of triplets [5], integer DCT(IntDCT)-based [14], and generalized-LSB [12] schemes. The test image is the grayscale `Barbara`.

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

[5] A. M. Alattar, "Reversible watermark using difference expansion of triplets," in *Proc. IEEE ICIP*, Barcelona, Spain, Sep. 2003, vol. 1, pp. 501–504.

[6] A. M. Alattar, "Reversible watermark using difference expansion of quads," in *Proc. ICASSP*, 2004, vol. 3, pp. 377–380.

[7] A. M. Alattar, "Reversible watermark using the difference expansion of a generalized integer transform," *IEEE Trans. Image Process.*, vol. 13, no. 8, pp. 1147–1156, Aug. 2004.

[8] L. Kamstra and H. J. A. M. Heijmans, "Reversible data embedding into images using wavelet techniques and sorting," *IEEE Trans. Image Process.*, vol. 14, no. 12, pp. 2082–2090, Dec. 2005.

[9] M. Veen, F. Bruekers, A. Leest, and S. Cavin, "High capacity reversible watermarking for audio," in *Proc. SPIE, Security, Steganography, and Watermarking of Multimedia Contents*, Jan. 2003, vol. 5020, pp. 1–11.

[10] A. Leest, M. Veen, and F. Bruekers, "Reversible watermarking for images," presented at the SPIE, Security, Steganography, and Watermarking of Multimedia Contents, San Jose, CA, Jan. 2004.

[11] C. De Vleeschouwer, J. F. Delaigle, and J. F. B. Macq, "Circular interpretation of bijective transformations in lossless watermarking for media asset management," *IEEE Trans. Multimedia*, vol. 5, no. 1, pp. 97–105, Mar. 2003.
[12] M. U. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Reversible data hiding," in *Proc. IEEE ICIP*, Rochester, NY, Sep. 2002, vol. 2, pp. 157–160.
[13] M. Celik, G. Sharma, A. M. Tekalp, and E. Saber, "Localized lossless authentication watermark (LAW)," *Proc. SPIE, Security and Watermarking of Multimedia Contents*, vol. 5020, no. 70, pp. 689–698, Jan. 2003.
[14] B. Yang, M. Schmucker, W. Funk, C. Busch, and S. Sun, "Integer DCT-based reversible watermarking for images using companding technique," in *Proc. SPIE, Security, Steganography, and Watermarking of Multimedia Contents*, San Jose, CA, Jan. 2004, pp. 405–415.
[15] G. Xuan, J. Chen, J. Zhu, Y. Q. Shi, Z. Ni, and W. Su, "Lossless data hiding based on integer wavelet transform," in *Proc. MMSP*, St. Thomas, Virgin Islands, Dec. 2002, pp. 312–315.
[16] G. Xuan, C. Yang, Y. Q. Shi, and Z. Ni, "High capacity lossless data hiding algorithms," in *Proc. ISCAS*, Vancouver, BC, Canada, May 2004.
[17] G. Xuan, Y. Q. Shi, Q. Yao, Z. Ni, C. Yang, J. Gao, and P. Chai, "Lossless data hiding using histogram shifting method based on integer wavelets," presented at the LNCS, Int. Workshop Digital Watermarking, Jeju Island, Korea, Nov. 2006.
[18] D. Coltuc and A. Tremeau, "Simple reversible watermarking schemes," in *Proc. SPIE, Security, Steganography, and Watermarking of Multimedia Contents*, San Jose, CA, Jan. 2005, pp. 561–568.
[19] D. Coltuc and J.-M. Chassery, "Simple reversible watermarking schemes: further results," in *Proc. SPIE, Security, Steganography, and Watermarking of Multimedia Contents*, San Jose, CA, Jan. 2006, pp. 739–746.
[20] T. Kalker and F. Willems, "Capacity bounds and constructions for reversible data-hiding," in *Proc. Int. Conf. Digital Signal Processing*, Jun. 2002, vol. 1, pp. 71–76.
[21] T. Kalker and F. Willems, "Capacity bounds and constructions for reversible data-hiding," presented at the SPIE Conf. Security and Watermarking of Multimedia Content V, Santa Clara, CA, Jan. 2003.
[22] F. Willems and T. Kalker, "Coding theorems for reversible embedding," presented at the DIMACS, Piscataway, NJ, Mar. 2003.
[23] A. R. Calderbank, I. Daubechies, W. Sweldens, and B. L. Yeo, "Wavelet transforms that map integers to integers," *Appl. Comput. Harmonics Anal.*, vol. 5, no. 3, pp. 332–369, 1998.
[24] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Appl.*, vol. 4, no. 3, pp. 245–267, 1998.
[25] W. Sweldens, "The lifting scheme: a construction of second generation wavelets," *SIAM J. Math. Anal.*, vol. 29, no. 2, pp. 511–546, Mar. 1998.
[26] P. G. Howard, F. Kossentini, B. Martins, S. Forchhammer, and W. J. Rucklidge, "The emerging JBIG2 standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 8, no. 7, pp. 838–848, Nov. 1998.
[27] M. D. Adams and R. K. Ward, "Symmetric-extension-compatible reversible integer-to-integer wavelet transforms," *IEEE Trans. Signal Process.*, vol. 51, no. 10, pp. 2624–2636, Oct. 2003.
[28] M. Vetterli and J. Kovacevic, *Wavelets and Subband Coding*. Englewood Cliffs, NJ: Prentice-Hall, 1995.
[29] R. C. Gonzalez and E. W. Richard, *Digital Image Processing*, 2nd ed. Englewood Cliffs, NJ: Prentice-Hall, 2002.
[30] USC-SIPI image database [Online]. Available: http://sipi.usc.edu/services/database/Database.html.

Sunil Lee (S'02) received the B.S. degree in electrical engineering from Yonsei University, Seoul, Korea, in 2001 and the M.S. degree in electrical engineering and computer science from the Korea Advanced Institute of Science and Technology (KAIST), Daejeon, Korea, in 2002, where he is currently pursuing the Ph.D. degree.

He joined the Multimedia Processing Laboratory at KAIST in 2001. His research interests include multimedia retrieval, multimedia security, digital watermarking, and multirate signal processing.

Mr. Lee is a student member of the Acoustical Society of Korea (ASK) and the Institute of Electrical Engineers of Korea (IEEK).

Chang D. Yoo (S'92–M'96) received the B.S. degree in engineering and applied science from the California Institute of Technology, Pasadena, in 1986; the M.S. degree in electrical engineering from Cornell University, Ithaca, NY, in 1988; and the Ph.D. degree in electrical engineering from the Massachusetts Institute of Technology (MIT), Cambridge, in 1996.

From 1997 to 1999, he was a Senior Researcher with Korea Telecom, Seoul, Korea. He joined the Department of Electrical Engineering with the Korea Advanced Institute of Science and Technology in 1999. From 2005 to 2006, he was with the Research Laboratory of Electronics at MIT. His research interests are in the application of machine learning and digital signal processing in multimedia.

Prof. Yoo is a member of Tau Beta Pi and Sigma Xi.

Ton Kalker (M'93–SM'99–F'01) received the Ph.D. degree from the University of Leiden, Leiden, The Netherlands, in 1986.

He is a Senior Researcher with the Multimedia Communications and Networking Department, Hewlett-Packard Laboratories, Palo Alto, CA, and a Faculty Member with the Technical University of Eindhoven, Eindhoven, The Netherlands. His primary responsibility is to support Hewlett-Packard in the area of media security, including interoperability of digital rights management, watermarking, and robust hashing and advanced encryption technologies. From 1985 to 1994, he was a member of the staff at Philips Research, Eindhoven, where he developed currently widely deployed watermarking and fingerprinting solutions. From 1983 to 1985, he was a Faculty Member of the University of Delft, Delft, The Netherlands.

Dr. Kalker participates in the Coral Consortium on DRM interoperability as one of the leading architects, in the DLNA Copy Protection Subcommittee, and in the OMA DRM Working Group. Ton is a Fellow of the IEEE for his contributions to practical applications of watermarking, in particular, watermarking for DVD–video copy protection. He cofounded IEEE TRANSACTIONS ON INFORMATION FORENSICS AND SECURITY.