

# Extended Block-Lifting-based Lapped Transforms

Taizo Suzuki, *Member, IEEE* and Hiroyuki Kudo, *Member, IEEE*

**Abstract**—We extend an original lapped transform (LT) and use block-lifting factorization to get an extended block-lifting-based LT (XBL-LT). The block-lifting structure maps integer input signals to integer output signals and results in a reversible transform that reduces rounding errors by merging many rounding operations. Although other such block-lifting-based LTs (BL-LTs) have been proposed, they are forcibly constrained by the use of discrete cosine transform (DCT) matrices. In contrast, XBL-LT is DCT-unconstrained and hence also embodies the DCT-constrained form. Furthermore, it has fewer rounding operations by merging the scaling factor with block-lifting coefficients. The both DCT-constrained and unconstrained XBL-LTs perform well at lossy-to-lossless image coding which has scalability from lossless data to lossy data.

**Index Terms**—Block-lifting structure, lapped transform (LT), lossy-to-lossless image coding

## I. INTRODUCTION

LAPPED transforms (LTs) [1] are popular subband transforms that can be used as substitutes for the discrete cosine transform (DCT) [2] used in image/video compression (image coding). Although almost all of the JPEG and H.26x series [3–5], image coding standards, use DCTs for their good energy compaction, DCT-based image codings generate unpleasant artifacts, i.e., blocking artifacts, at low bit rates due to their ignoring the continuity of the blocks. LT-based image coding solves that problem by using a processing that works over the blocks.

The lifting structure [6] is a very important technology to achieve a lossless mode in subband transform-based image coding. It maps integer input signals to integer output signals; i.e., it is an integer-to-integer transform. The  $4 \times 8$  lifting-based LT (L-LT) [7] in JPEG XR [8], the newest image coding standard, is a time-domain LT (TDLT) [9] with simple scaling factors and lifting structures. In spite of its simple structure, it performs well at lossy-to-lossless image coding, which has scalability from lossless to lossy data. The block-lifting structure [10] is a class of lifting structures and results in a reversible transform that reduces rounding errors by merging many rounding operations. Inspired by the L-LT in JPEG XR, we have proposed block-lifting-based LTs (BL-LTs) [11], [12] that perform well with larger block size than those of the L-LT in JPEG XR. However, they are forcibly DCT-constrained and degrade coding performance at high bit rates.

Here, we extend an original LT and use block-lifting factorization to get an extended BL-LT (XBL-LT). The XBL-LT is DCT-unconstrained, unlike the BL-LTs presented in [11], [12], and hence also embodies the DCT-constrained form. Furthermore, more rounding operations than the methods

described in our previous studies are removed by merging the scaling factor with block-lifting coefficients. As a result, the both DCT-constrained and unconstrained XBL-LTs perform well at lossy-to-lossless image coding.

**Notation:** The italic letter  $M$  ( $M = 2^n$ ,  $n \in \mathbb{N}$ ) denotes the block size. Boldface letters  $\mathbf{I}_m$ ,  $\mathbf{J}_m$ ,  $\mathbf{0}$ , and  $\mathbf{D}_m$  denote an  $m \times m$  identity matrix, an  $m \times m$  reversal matrix, a null matrix, and an  $m \times m$  diagonal matrix with alternating  $\pm 1$  entries (i.e.,  $\text{diag}\{1, -1, 1, -1, \dots\}$ ), respectively. The superscripts  $T$  and  $-1$  respectively mean the transpose and inverse of a matrix.

## II. REVIEW AND DEFINITION

### A. Lapped Transform (LT)

In accordance with [11], [12], let  $\mathbf{E}(z)$  be a polyphase matrix of an  $M \times 2M$  LT with a scaling factor  $s$  derived from the L-LT in JPEG XR [7]:

$$\mathbf{E}(z) = \mathbf{P} \begin{bmatrix} \mathbf{I}_N & \mathbf{0} \\ \mathbf{0} & \mathbf{S}_N^{\text{IV}} \mathbf{C}_N^{\text{III}} \end{bmatrix} \Gamma(z) \begin{bmatrix} \mathbf{C}_N^{\text{II}} & \mathbf{0} \\ \mathbf{0} & \mathbf{C}_N^{\text{IV}} \mathbf{J}_N \end{bmatrix} \widetilde{\mathbf{S}} \mathbf{W} \mathbf{J}_M, \quad (1)$$

where

$$\begin{aligned} \Gamma(z) &= \mathbf{W} \Lambda(z) \mathbf{W}, \quad \Lambda(z) = \text{diag}\{\mathbf{I}_N, z^{-1} \mathbf{I}_N\} \\ \mathbf{W} &= \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_N & \mathbf{I}_N \\ \mathbf{I}_N & -\mathbf{I}_N \end{bmatrix}, \quad \widetilde{\mathbf{W}} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I}_N & \mathbf{J}_N \\ \mathbf{J}_N & -\mathbf{I}_N \end{bmatrix} \\ \mathbf{S} &= \text{diag}\{s \mathbf{I}_N, s^{-1} \mathbf{I}_N\}. \end{aligned}$$

$\mathbf{C}_M^X$  and  $\mathbf{S}_M^X$  are type- $X$  DCT (DCT- $X$ ) and type- $X$  discrete sine transform (DST- $X$ ) matrices, and the  $(i, j)$ -elements of  $\mathbf{C}_M^{\text{II}}$  and  $\mathbf{C}_M^{\text{IV}}$  are

$$\begin{aligned} [\mathbf{C}_M^{\text{II}}]_{i,j} &= \sqrt{\frac{2}{M}} c_i \cos\left(\frac{i(j+1/2)\pi}{M}\right) \\ [\mathbf{C}_M^{\text{IV}}]_{i,j} &= \sqrt{\frac{2}{M}} \cos\left(\frac{(i+1/2)(j+1/2)\pi}{M}\right), \end{aligned}$$

where  $c_k = 1/\sqrt{2}$  ( $k = 0$ ) or 1 ( $k \neq 0$ ), respectively. The following relationships between matrices can be established:

$$\mathbf{C}_M^{\text{III}} = \left(\mathbf{C}_M^{\text{II}}\right)^{-1} = \left(\mathbf{C}_M^{\text{II}}\right)^T, \quad \mathbf{S}_M^{\text{IV}} = \mathbf{D}_M \mathbf{C}_M^{\text{IV}} \mathbf{J}_M. \quad (2)$$

Here,  $\mathbf{P}$  is an  $M \times M$  permutation matrix. The optimal scaling  $s$  in  $\mathbf{S}$  is empirically determined, e.g.,  $s = 0.8981$  if  $M = 8$  and  $0.9360$  if  $M = 16$ . Since the LT in Eq. (1) with  $s = 1$  is completely equivalent to a lapped orthogonal transform (LOT), we will use the LT in Eq. (1) as a representative expression of LT.

T. Suzuki and H. Kudo are with the Faculty of Engineering, Information and Systems, University of Tsukuba, Tsukuba, Ibaraki, 305-8573 Japan (e-mail: {taizo, kudo}@es.tsukuba.ac.jp).

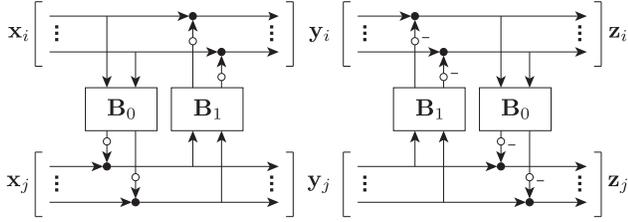


Fig. 1. Block-lifting structure (black and white circles mean adders and rounding operations, respectively).

### B. Block-Lifting Structure

The block-lifting structure [10] (Fig. 1) is a class of lifting structures. The structure can be expressed as follows:

$$\begin{aligned} \mathbf{y}_j &= \mathbf{x}_j + \text{round}(\mathbf{B}_0 \mathbf{x}_i), & \mathbf{y}_i &= \mathbf{x}_i + \text{round}(\mathbf{B}_1 \mathbf{y}_j) \\ \mathbf{z}_i &= \mathbf{y}_i - \text{round}(\mathbf{B}_1 \mathbf{y}_j) = \mathbf{x}_i, & \mathbf{z}_j &= \mathbf{y}_j - \text{round}(\mathbf{B}_0 \mathbf{z}_i) = \mathbf{x}_j, \end{aligned}$$

where  $\mathbf{x}_\times$ ,  $\mathbf{y}_\times$ , and  $\mathbf{z}_\times$  are  $N \times 1$  input/output vector signals,  $\text{round}(\cdot)$  is a rounding operation, and the block-lifting coefficients  $\mathbf{B}_0$  and  $\mathbf{B}_1$  are  $N \times N$  arbitrary matrices. In this case, the matrices and their inverses are expressed as lower and upper block-lifting matrices as follows:

$$\begin{aligned} \mathfrak{L}[\mathbf{B}_0] &\triangleq \begin{bmatrix} \mathbf{I}_N & \mathbf{0} \\ \mathbf{B}_0 & \mathbf{I}_N \end{bmatrix}, & \mathfrak{L}[\mathbf{B}_0]^{-1} &= \mathfrak{L}[-\mathbf{B}_0] \\ \mathfrak{U}[\mathbf{B}_1] &\triangleq \begin{bmatrix} \mathbf{I}_N & \mathbf{B}_1 \\ \mathbf{0} & \mathbf{I}_N \end{bmatrix}, & \mathfrak{U}[\mathbf{B}_1]^{-1} &= \mathfrak{U}[-\mathbf{B}_1]. \end{aligned}$$

Rounding errors generated by the rounding operation in each lifting step degrade coding performance. The block-lifting structure reduces such rounding errors by merging many rounding operations. A special class of block-lifting structure is expressed as [13]

$$\begin{aligned} \begin{bmatrix} \mathbf{M} & \mathbf{0} \\ \mathbf{0} & \mathbf{M}^{-1} \end{bmatrix} &= \mathfrak{L}[\mathbf{M}^{-1}] \mathfrak{U}[-\mathbf{M}] \mathfrak{L}[\mathbf{M}^{-1}] \tilde{\mathbf{J}}_M & (3) \\ &= \hat{\mathbf{J}}_M \mathfrak{L}[-\mathbf{M}] \mathfrak{U}[\mathbf{M}^{-1}] \mathfrak{L}[-\mathbf{M}], & (4) \end{aligned}$$

where

$$\tilde{\mathbf{J}}_M = \begin{bmatrix} \mathbf{0} & \mathbf{I}_N \\ -\mathbf{I}_N & \mathbf{0} \end{bmatrix}, \quad \hat{\mathbf{J}}_M = \begin{bmatrix} \mathbf{0} & -\mathbf{I}_N \\ \mathbf{I}_N & \mathbf{0} \end{bmatrix},$$

and  $\mathbf{M}$  is an  $N \times N$  arbitrary nonsingular matrix.

### III. EXTENDED BLOCK-LIFTING-BASED LAPPED TRANSFORM (XBL-LT)

*Theorem:* We can extend the DCT-constrained LT in Eq. (1) to a DCT-unconstrained LT as follows:

$$\mathbf{E}(z) = \mathbf{P} \begin{bmatrix} \mathbf{I}_N & \mathbf{0} \\ \mathbf{0} & \mathbf{D}_N \tilde{\mathbf{V}}^{-1} \mathbf{J}_N \tilde{\mathbf{U}}^{-1} \end{bmatrix} \Gamma(z) \begin{bmatrix} \tilde{\mathbf{U}} & \mathbf{0} \\ \mathbf{0} & \tilde{\mathbf{V}} \mathbf{J}_N \end{bmatrix} \tilde{\mathbf{S}} \tilde{\mathbf{W}} \mathbf{J}_M,$$

where  $\tilde{\mathbf{U}}$  and  $\tilde{\mathbf{V}}$  are  $N \times N$  arbitrary nonsingular matrices. This equation can be simplified as

$$\mathbf{E}(z) = \mathbf{P} \begin{bmatrix} \mathbf{I}_N & \mathbf{0} \\ \mathbf{0} & \mathbf{V}^{-1} \mathbf{J}_N \mathbf{U}^{-1} \end{bmatrix} \Gamma(z) \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{V} \mathbf{J}_N \end{bmatrix} \hat{\mathbf{W}} \mathbf{J}_M, \quad (5)$$

where

$$\mathbf{U} = \sqrt{2s} \tilde{\mathbf{U}}, \quad \mathbf{V} = \frac{1}{\sqrt{2s}} \tilde{\mathbf{V}}, \quad \hat{\mathbf{W}} = \begin{bmatrix} \frac{1}{2} \mathbf{I}_N & \frac{1}{2} \mathbf{J}_N \\ \mathbf{J}_N & -\mathbf{I}_N \end{bmatrix},$$

by using Eq. (2) and skipping the sign inversion matrix  $\mathbf{D}_N$ . The scaling matrix  $\mathbf{S}$  is being embedded in  $\mathbf{U}$  and  $\mathbf{V}$ . When  $\tilde{\mathbf{U}} = \mathbf{C}_N^H$  and  $\tilde{\mathbf{V}} = \mathbf{C}_N^V$ , the LT in Eq. (5) is completely equivalent to the LT in Eq. (1) except that the signs are different. Then, we factorize the LT in Eq. (5) into block-lifting structures as

$$\begin{aligned} \mathbf{E}(z) &= \mathbf{P} \mathfrak{L}[\mathbf{B}_4] \mathfrak{U}[\mathbf{B}_3] \Lambda(z) \mathfrak{U}[\mathbf{B}_3] \mathfrak{L}[\mathbf{B}_2] \mathfrak{U}[\mathbf{B}_1] \mathfrak{L}[\mathbf{B}_0] \\ &\cdot \mathfrak{U} \left[ -\frac{1}{2} \mathbf{J}_N \right] \mathfrak{L}[\mathbf{J}_N] \tilde{\mathbf{J}}_M, \end{aligned} \quad (6)$$

wherein each matrix is defined as

$$\begin{aligned} \mathbf{B}_0 &= -\mathbf{V}^{-1}, \quad \mathbf{B}_1 = \mathbf{V}, \quad \mathbf{B}_2 = \mathbf{B}_0 + \mathbf{B}_4 \\ \mathbf{B}_3 &= -\frac{1}{2} \mathbf{U} \mathbf{J}_N \mathbf{V}, \quad \mathbf{B}_4 = \mathbf{V}^{-1} \mathbf{J}_N \mathbf{U}^{-1}. \end{aligned}$$

Actually, the block-lifting matrices  $\mathfrak{U}[\mathbf{B}_3]$  on both sides of the delay matrix  $\Lambda(z)$  are collectively implemented, as shown in Fig. 2.

*Proof:* We can perform an easy matrix manipulation as follows:

$$\begin{aligned} \begin{bmatrix} \mathbf{M}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_1 \end{bmatrix} \begin{bmatrix} \mathbf{I}_N & \mathbf{N}_0 \\ \mathbf{N}_1 & \mathbf{I}_N \end{bmatrix} &= \begin{bmatrix} \mathbf{I}_N & \mathbf{M}_0 \mathbf{N}_0 \mathbf{M}_1^{-1} \\ \mathbf{M}_1 \mathbf{N}_1 \mathbf{M}_0^{-1} & \mathbf{I}_N \end{bmatrix} \begin{bmatrix} \mathbf{M}_0 & \mathbf{0} \\ \mathbf{0} & \mathbf{M}_1 \end{bmatrix}, \end{aligned}$$

where  $\mathbf{M}_\times$  and  $\mathbf{N}_\times$  are an  $N \times N$  arbitrary nonsingular matrix and  $N \times N$  arbitrary matrix, respectively. First,  $\Gamma(z)$  in Eq. (5) can easily be represented by

$$\Gamma(z) = \mathfrak{L}[\mathbf{I}_N] \mathfrak{U} \left[ -\frac{1}{2} \mathbf{I}_N \right] \Lambda(z) \mathfrak{U} \left[ \frac{1}{2} \mathbf{I}_N \right] \mathfrak{L}[-\mathbf{I}_N].$$

Next,  $\mathbf{U}^{-1}$  in Eq. (5) is moved to the right side of  $\Gamma(z)$  and simplified as

$$\begin{aligned} \Psi(z) &\triangleq \begin{bmatrix} \mathbf{I}_N & \mathbf{0} \\ \mathbf{0} & \mathbf{U}^{-1} \end{bmatrix} \Gamma(z) \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_N \end{bmatrix} \\ &= \mathfrak{L}[\mathbf{U}^{-1}] \mathfrak{U} \left[ -\frac{1}{2} \mathbf{U} \right] \Lambda(z) \mathfrak{U} \left[ \frac{1}{2} \mathbf{U} \right] \mathfrak{L}[-\mathbf{U}^{-1}] \begin{bmatrix} \mathbf{U} & \mathbf{0} \\ \mathbf{0} & \mathbf{U}^{-1} \end{bmatrix} \\ &= \mathfrak{L}[\mathbf{U}^{-1}] \mathfrak{U} \left[ -\frac{1}{2} \mathbf{U} \right] \Lambda(z) \mathfrak{U} \left[ -\frac{1}{2} \mathbf{U} \right] \mathfrak{L}[\mathbf{U}^{-1}] \tilde{\mathbf{J}}_M, \end{aligned}$$

because the block diagonal matrix  $\text{diag}\{\mathbf{U}, \mathbf{U}^{-1}\}$  in  $\Psi(z)$  can be factorized into block-lifting structures as in Eq. (3). Then,  $\mathbf{V}^{-1} \mathbf{J}_N$  in Eq. (5) is moved to the right side of  $\Psi(z)$  and

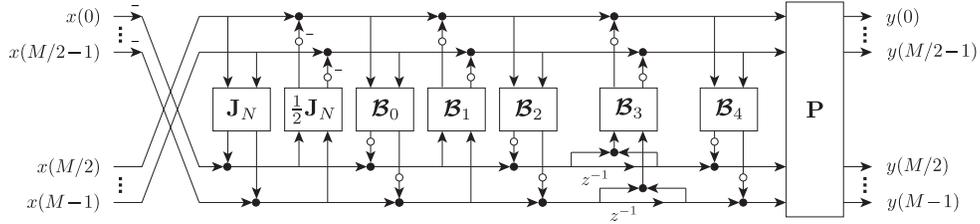


Fig. 2. XBL-LT (black and white circles mean adders and rounding operations, respectively).

TABLE I  
CODING GAINS OF XBL-LTs.

Block Size	DCT-Constrained	DCT-Unconstrained
8	9.4475	<b>9.4538</b>
16	9.8455	<b>9.8621</b>

TABLE II  
NUMBERS OF ROUNDING OPERATIONS OF BL-LTs.

Block Size	BL-LT [11]	BL-LT [12]	XBL-LT
8	48	28	<b>24</b>
16	96	56	<b>48</b>
$M$	$6M$	$7M/2$	<b><math>3M</math></b>

simplified as

$$\begin{aligned}
 \Omega(z) &\triangleq \begin{bmatrix} \mathbf{I}_N & \mathbf{0} \\ \mathbf{0} & \mathbf{v}^{-1}\mathbf{J}_N \end{bmatrix} \Psi(z) \begin{bmatrix} \mathbf{I}_N & \mathbf{0} \\ \mathbf{0} & \mathbf{v}\mathbf{J}_N \end{bmatrix} \\
 &= \mathfrak{L}[\mathbf{W}^{-1}] \mathfrak{U} \left[ -\frac{1}{2}\mathbf{W} \right] \Lambda(z) \mathfrak{U} \left[ -\frac{1}{2}\mathbf{W} \right] \\
 &\quad \cdot \mathfrak{L}[\mathbf{W}^{-1}] \tilde{\mathbf{J}}_M \begin{bmatrix} \mathbf{v}^{-1}\mathbf{J}_N & \mathbf{0} \\ \mathbf{0} & \mathbf{v}\mathbf{J}_N \end{bmatrix} \\
 &= \mathfrak{L}[\mathbf{W}^{-1}] \mathfrak{U} \left[ -\frac{1}{2}\mathbf{W} \right] \Lambda(z) \mathfrak{U} \left[ -\frac{1}{2}\mathbf{W} \right] \\
 &\quad \cdot \mathfrak{L}[\mathbf{W}^{-1} - \mathbf{v}^{-1}] \mathfrak{U}[\mathbf{v}] \mathfrak{L}[-\mathbf{v}^{-1}] \begin{bmatrix} \mathbf{J}_N & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_N \end{bmatrix},
 \end{aligned}$$

where  $\mathbf{W} = \mathbf{U}\mathbf{J}_N\mathbf{V}$ , because the block diagonal matrix  $\text{diag}\{\mathbf{v}^{-1}, \mathbf{v}\}$  in  $\Omega(z)$  can also be factorized into block-lifting structures as in Eq. (4). Finally, the residual part  $\widehat{\mathbf{W}}\mathbf{J}_M$  in Eq. (5) and  $\text{diag}\{\mathbf{J}_N, \mathbf{J}_N\}$  in  $\Omega(z)$  are collectively factorized as

$$\begin{bmatrix} \mathbf{J}_N & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_N \end{bmatrix} \widehat{\mathbf{W}}\mathbf{J}_M = \mathfrak{U} \left[ -\frac{1}{2}\mathbf{J}_N \right] \mathfrak{L}[\mathbf{J}_N] \tilde{\mathbf{J}}_M.$$

□

#### IV. EXPERIMENTAL RESULTS

##### A. Coding Gain, Frequency Response, and Number of Rounding Operations

We designed  $8 \times 16$  and  $16 \times 32$  XBL-LTs by optimizing the coding gain (CG) [14]

$$\text{CG [dB]} = 10 \log_{10} \frac{\sigma_x^2}{\prod_{k=0}^{M-1} \sigma_{x_k}^2 \|f_k\|^2},$$

where  $\sigma_x^2$  is the variance of the input signal,  $\sigma_{x_k}^2$  is the variance of the  $k$ -th subbands and  $\|f_k\|^2$  is the norm of the  $k$ -th synthesis filter. To simplify the design in DCT-unconstrained

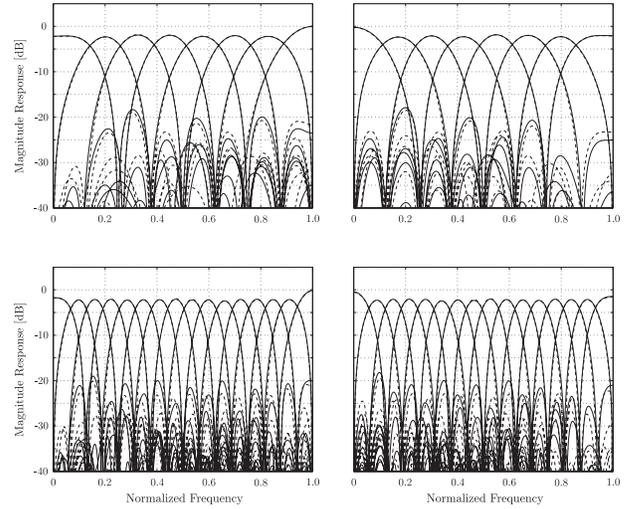


Fig. 3. Frequency responses of analysis and synthesis parts of XBL-LTs (dashed and solid lines indicate DCT-constrained and unconstrained cases, respectively): (top)  $8 \times 16$  XBL-LT, (bottom)  $16 \times 32$  XBL-LT.

case, we set  $\tilde{\mathbf{U}}$  and  $\tilde{\mathbf{V}}$  in Eq. (6) as  $N \times N$  arbitrary unitary matrices, where  $\tilde{\mathbf{U}}$  is designed such that it has structural one-degree regularity [10] to achieve good image coding. Table I compares of the CGs[dB] of the XBL-LTs in the DCT-constrained and unconstrained cases. In addition, Fig. 3 shows the frequency responses of the analysis and synthesis parts of the XBL-LTs for the same cases as in Table I. In Table I and Fig. 3, the DCT-unconstrained case had slightly better results than the DCT-constrained case. Table II shows the numbers of rounding operations of BL-LTs. It is clear that the XBL-LTs have fewer rounding operations than those of conventional BL-LTs.

##### B. Lossy-to-Lossless Image Coding

For convenience regarding the number of pages, lossy-to-lossless image coding was implemented in only the  $M = 8$  case. We used L-LTs in [7], [9], [11], [12] as the conventional L-LTs. The L-LT in [7] is the  $4 \times 8$  L-LT for JPEG XR. The L-LT in [9] is the  $8 \times 16$  TDLT with the pre-filtering part indicated by Cfg. 5 in Table V in [9] and the DCT part indicated by [15]. The L-LTs in [11], [12] are the DCT-constrained BL-LTs. After the images were transformed by the L-LTs and periodic extension at the boundaries, the transformed coefficients were rearranged from the subband mode to the multiresolution mode similar to the wavelet transform. They were encoded with a common wavelet-based zerotree

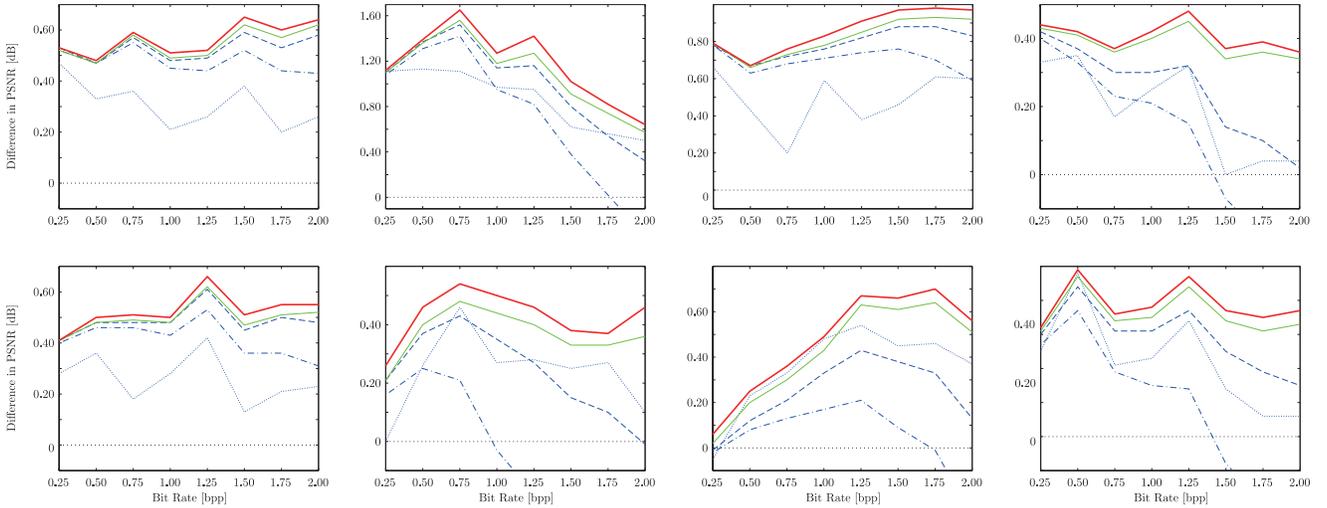


Fig. 4. Results of lossy image coding (blue dotted, blue chained, blue dashed, green solid, and red bold solid lines indicate  $8 \times 16$  L-LT in [9],  $8 \times 16$  BL-LT in [11],  $8 \times 16$  BL-LT in [12],  $8 \times 16$  DCT-constrained XBL-LT, and  $8 \times 16$  DCT-unconstrained XBL-LT, respectively): (top) *Baboon*, *Barbara*, *Finger*, and *Goldhill*, (bottom) *Grass*, *Lena*, *Pepper*, and *Tank*.

TABLE III

RESULTS OF LOSSLESS IMAGE CODING (LBR [BPP],  $(\alpha)$  AND  $(\beta)$  MEAN DCT-CONSTRAINED AND DCT-UNCONSTRAINED CASES, RESPECTIVELY.)

Test Images	L-LTs		BL-LTs		XBL-LTs	
	[7]	[9]	[11]	[12]	$(\alpha)$	$(\beta)$
<i>Baboon</i>	6.23	6.26	6.21	6.21	6.21	<b>6.20</b>
<i>Barbara</i>	4.96	4.90	4.90	4.86	4.84	<b>4.83</b>
<i>Boat</i>	5.20	5.15	5.16	5.14	5.13	<b>5.12</b>
<i>Elaine</i>	5.27	5.24	5.26	<b>5.23</b>	<b>5.23</b>	<b>5.23</b>
<i>Finger</i>	5.89	5.92	5.84	<b>5.82</b>	<b>5.82</b>	<b>5.82</b>
<i>Finger2</i>	5.62	5.64	5.55	5.53	5.52	<b>5.51</b>
<i>Goldhill</i>	5.12	5.17	5.15	5.12	<b>5.11</b>	<b>5.11</b>
<i>Grass</i>	6.09	6.14	6.08	6.07	6.07	<b>6.06</b>
<i>Lena</i>	4.63	4.64	4.66	4.62	4.61	<b>4.60</b>
<i>Pepper</i>	5.00	4.93	4.96	4.93	4.92	<b>4.91</b>
<i>Room</i>	4.47	4.51	4.55	4.48	4.46	<b>4.44</b>
<i>Sakura</i>	5.94	6.03	5.95	5.93	5.92	<b>5.90</b>
<i>Station</i>	5.16	5.25	5.21	5.16	5.15	<b>5.14</b>
<i>Tank</i>	5.16	5.19	5.18	5.16	<b>5.15</b>	<b>5.15</b>

coder (SPIHT) [16]. The XBL-LTs were compared with the conventional L-LTs by using the lossless bit rate (LBR) and peak-to-noise ratio (PSNR) in lossy-to-lossless image coding:

$$\text{LBR [bpp]} = \frac{\text{Total number of bits [bit]}}{\text{Total number of pixels [pixel]}}$$

$$\text{PSNR [dB]} = 10 \log_{10} \left( \frac{255^2}{\text{MSE}} \right).$$

Table III and Fig. 4 show the results of lossless and lossy image coding. The vertical axes in Fig. 4 are the differences in PSNRs relative to the  $4 \times 8$  L-LT in [7] for JPEG XR. It is clear that the XBL-LTs outperformed the conventional L-LTs in lossy-to-lossless image coding. In particular, although lifting-based transforms tended to degrade coding performance at high bit rates, the XBL-LTs preserved coding performance even at such high bit rates.

## V. CONCLUSION

By extending an original LT and using block-lifting factorization, we developed an XBL-LT with fewer rounding

operations. It is DCT-unconstrained and hence can be DCT-constrained as well; i.e., it can be considered to be a more general structure than other BL-LTs. Although we constrained the design by using unitary matrices in this paper, the DCT-unconstrained structures have the potential to achieve better coding.

## ACKNOWLEDGMENT

This work was supported by JSPS Grant-in-Aid for Young Scientists (B), Grant Number 25820152.

## REFERENCES

- [1] H. S. Malvar, *Signal Processing with Lapped Transforms*, Norwood, MA: Artech House, 1992.
- [2] K. R. Rao and P. Yip, *Discrete Cosine Transform Algorithms*, Academic Press, 1990.
- [3] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Trans. Consum. Electr.*, vol. 38, no. 1, pp. xviii–xxxiv, Feb. 1992.
- [4] T. Wiegand, G. J. Sullivan, G. Bjøntegaard, and A. Luthra, "Overview of the H.264/AVC video coding standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 13, no. 7, pp. 560–576, July 2003.
- [5] G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.
- [6] W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," *Appl. Comput. Harmon. Anal.*, vol. 3, no. 2, pp. 186–200, Apr. 1996.
- [7] C. Tu, S. Srinivasan, G. J. Sullivan, S. Regunathan, and H. S. Malvar, "Low-complexity hierarchical lapped transform for lossy-to-lossless image coding in JPEG XR/HD Photo," in *Proc. of SPIE*, San Diego, CA, Aug. 2008, vol. 7073, pp. 1–12.
- [8] F. Dufaux, G. J. Sullivan, and T. Ebrahimi, "The JPEG XR image coding standard," *IEEE Signal Process. Mag.*, vol. 26, no. 6, pp. 195–199, 204, Nov. 2009.
- [9] T. D. Tran, J. Liang, and C. Tu, "Lapped transform via time-domain pre- and post-filtering," *IEEE Trans. Signal Process.*, vol. 6, no. 6, pp. 1557–1571, June 2003.
- [10] T. Suzuki, M. Ikehara, and T. Q. Nguyen, "Generalized block-lifting factorization of  $M$ -channel biorthogonal filter banks for lossy-to-lossless image coding," *IEEE Trans. Image Process.*, vol. 21, no. 7, pp. 3220–3228, July 2012.
- [11] T. Suzuki and M. Ikehara, "Integer fast lapped transforms based on direct-lifting of DCTs for lossy-to-lossless image coding," *EURASIP J. Image. Video Process.*, vol. 2013, no. 65, pp. 1–9, Dec. 2013.

- [12] T. Suzuki and H. Kudo, "Integer fast lapped biorthogonal transform via applications of DCT matrices and dyadic-valued factors for lifting coefficient blocks," in *Proc. of ICIP'13*, Merboulne, Australia, Sep. 2013, pp. 800–804.
- [13] T. Suzuki and M. Ikehara, "Integer DCT based on direct-lifting of DCT-IDCT for lossless-to-lossy image coding," *IEEE Trans. Image Process.*, vol. 19, no. 11, pp. 2958–2965, Nov. 2010.
- [14] G. Strang and T. Nguyen, *Wavelets and Filter Banks*, Wellesley-Cambridge Press, 1996.
- [15] S. Fukuma, K. Ohyama, M. Iwahashi, and N. Kambayashi, "Lossless 8-point fast discrete cosine transform using lossless Hadamard transform," Tech. Rep. of IEICE, DSP99-103, Oct. 1999.
- [16] A. Said and W. A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 6, no. 3, pp. 243–250, June 1996.