FOUR-CHANNEL LIFTING-HOUSEHOLDER-BASED HADAMARD TRANSFORM

Taizo Suzuki

Faculty of Engineering, Information and Systems, University of Tsukuba
Tsukuba, Ibaraki, 305-8573 Japan
Email: taizo@cs.tsukuba.ac.jp

ABSTRACT

This study presents a class of four-channel multiplierless reversible Hadamard transforms (HTs), called lifting-Householder-based HT (LiftH^{2}T), for various signal processing and communication applications. This class is obtained by using a lifting-Householder factorization of a particular 4×4 symmetric orthogonal matrix and its application to a four-channel HT. In spite of the simple multiplierless structure with only nine adders, one shifter, and three process steps, it basically outperforms the integer HT (IntHT) and lifting-based HT (LiftHT) in the JPEG XR standard at lossy-to-lossless image coding, thanks to its considering the dynamic range and having less rounding error.

Index Terms**—** Hadamard transform (HT), Householder factorization, lifting factorization, lossy-to-lossless image coding.

1. INTRODUCTION

The Hadamard transform (HT) [1] is one of the most practical transforms for signal processing and communication applications such as image compression (coding), image watermarking, face recognition, motion estimation, multicarrier CDMA, and multiband and ultra-wideband OFDM [2]. It is often part of popular transforms, e.g., DCT [3], used in image coding standards such as JPEG [4], H.264/AVC [5], and H.265/HEVC [6]. It has very low computational complexity because it only uses adders (and subtracters) and has no multipliers, i.e., ± 1; when the dynamic range is ignored, it is called an integer HT (IntHT). Note however that the dynamic range should be implemented in the synthesis part for ensuring reversibility, and it is unsuitable for image coding, especially lossless mode.

Sweldens presented a lifting structure [7], which is an identity matrix with one single nonzero off-diagonal element and a rounding operation. Since the structure maps integer input signals to integer output signals, it achieves a reversible transform even if the lifting coefficient is any value. Also, when the coefficient is a dyadic value $n/2^b$ ($b, n \in \mathbb{N}$), it can be implemented by using only adders and shifters [8]. Additionally, thanks to its considering the dynamic range, it can produce a normalized HT, unlike IntHT. The four-channel lifting-based HT (LiftHT) [9], which is part of the lapped transform (LT) of the JPEG XR [10], is a simple reversible transform that consists of only ten adders, one shifter, and seven process steps. The process step means a set of steps that can be processed in parallel. Since many process steps delay signals, fewer process steps are desired [11]. We have incorporated the LiftHT in a lifting-based LT for effective lossy-to-lossless image coding [12].

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

This study presents a class of four-channel multiplierless reversible HTs, called lifting-Householder-based HT (LiftH^{2}T), for various signal processing and communication applications. This class is obtained by using a lifting-Householder factorization of a particular 4×4 symmetric orthogonal matrix and its application to a four-channel HT. It has only nine adders, one shifter, and three process steps. LiftH^{2}T basically outperforms IntHT and LiftHT in the JPEG XR at lossy-to-lossless image coding in spite of having one fewer adder and four fewer process steps than LiftHT, thanks to its considering the dynamic range and having less rounding error.

Notation: $\mathbf{I}_N$, $\otimes$, and $\mathbf{P}_4$ respectively denote an $N \times N$ ($N \in \mathbb{N}$, $N \neq 0$) identity matrix, the Kronecker product, and a 4×4 permutation matrix as follows:

$$\mathbf{P}_4 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{1}$$

2. REVIEW AND DEFINITION

2.1. Hadamard Transforms (HTs)

An $M \times M$ ($M = 2^m$, $m \in \mathbb{N}, m \neq 0$) HT matrix $\boldsymbol{\mathcal{H}}_M$ is expressed as follows [1]:

$$\boldsymbol{\mathcal{H}}_M = \boldsymbol{\mathcal{H}}_2 \otimes \boldsymbol{\mathcal{H}}_{\frac{M}{2}} = \frac{1}{\sqrt{2}} \begin{bmatrix} \boldsymbol{\mathcal{H}}_{\frac{M}{2}} & \boldsymbol{\mathcal{H}}_{\frac{M}{2}} \\ \boldsymbol{\mathcal{H}}_{\frac{M}{2}} & -\boldsymbol{\mathcal{H}}_{\frac{M}{2}} \end{bmatrix}, \tag{2}$$

where

$$\boldsymbol{\mathcal{H}}_1 = 1. \tag{3}$$

The HT in Eq. (2) is a normalized HT, and the four-channel case $\boldsymbol{\mathcal{H}}_4$ can be simplified as follows (see the top-left of Fig. 1):

$$\begin{aligned} \boldsymbol{\mathcal{H}}_4 &= \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \\ &= \frac{1}{2} \mathbf{P}_4 \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix} \mathbf{P}_4 \begin{bmatrix} 1 & 1 & 0 & 0 \\ 1 & -1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & -1 \end{bmatrix}, \end{aligned} \tag{4}$$

where the scaling $1/2$s can be replaced by one-bit shifters.

Although IntHT $\widetilde{\boldsymbol{\mathcal{H}}}_M = \sqrt{M} \boldsymbol{\mathcal{H}}_M$ is constructed from only adders, i.e., ± 1, it is unsuitable for image coding, especially lossless mode, because it ignores the dynamic range. The four-channel

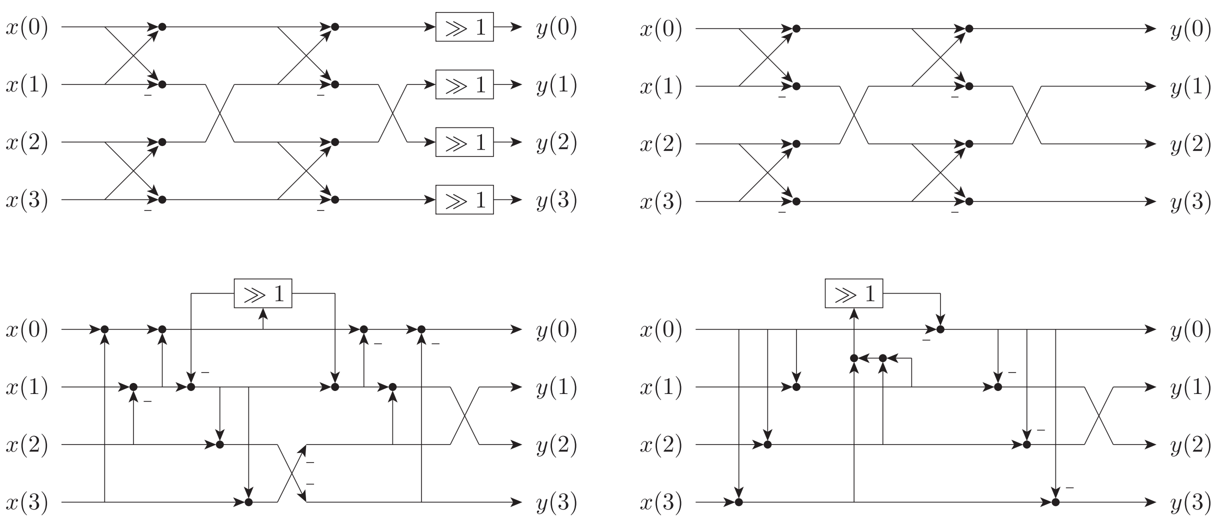

Fig. 1. Four-channel HTs (black circles and ($\gg 1$)s mean adders and one-bit shifters, respectively): (top) normalized HT and IntHT, (bottom) LiftHT and LiftH^{2}T.

case $\widetilde{\boldsymbol{\mathcal{H}}}_4 = 2\boldsymbol{\mathcal{H}}_4$ is shown at the top-right of Fig. 1. It has eight adders and two process steps.

On the other hand, a lifting structure [7] can achieve multiplierless reversible HTs and at the same time allow for the dynamic range. The resulting four-channel LiftHT $\boldsymbol{\mathcal{H}}_4$, which is part of the LT of the JPEG XR [10], is expressed as (see the bottom-left of Fig. 1),

$$
\begin{aligned}
\boldsymbol{\mathcal{H}}_4 = \mathbf{P}_4 &\begin{bmatrix} 1 & 0 & 0 & -1 \\ 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \\
&\cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ \frac{1}{2} & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & -1 \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 \end{bmatrix} \\
&\cdot \begin{bmatrix} 1 & 0 & 0 & 0 \\ -\frac{1}{2} & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.
\end{aligned}
\tag{5}
$$

It has ten adders, one shifter, and seven process steps.

2.2. Lifting-Householder Factorization of Orthogonal Matrix

An $M \times M$ Householder matrix $\mathbf{H}_M[\mathbf{u}_k]$ is expressed as follows [13]:

$$\mathbf{H}_M[\mathbf{u}_k] = \mathbf{I}_M - 2\mathbf{u}_k\mathbf{u}_k^T, \tag{6}$$

where

$$\mathbf{u}_k = \begin{bmatrix} u_{k,0} & u_{k,1} & \cdots & u_{k,M-1} \end{bmatrix}^T \tag{7}$$

and $u_{k,l}$ ($l = 0, 1, \cdots, M-1$) is an arbitrary value that satisfies $||\mathbf{u}_k|| = 1$. Also, any Householder matrix is identical to its inverse because it is a symmetric orthogonal matrix, i.e.,

$$(\mathbf{H}_M[\mathbf{u}_k])^{-1} = (\mathbf{H}_M[\mathbf{u}_k])^T = \mathbf{H}_M[\mathbf{u}_k]. \tag{8}$$

Any $M \times M$ orthogonal matrix $\mathbf{X}$ can be always factorized into $(M-1)$ cascading Householder matrices as follows:

$$\mathbf{X} = \mathbf{H}_M[\mathbf{u}_0]\,\mathbf{H}_M[\mathbf{u}_1]\cdots\mathbf{H}_M[\mathbf{u}_{M-2}], \tag{9}$$

where

$$
\begin{aligned}
&\begin{bmatrix} \mathbf{u}_0 & \mathbf{u}_1 & \cdots & \mathbf{u}_{M-2} \end{bmatrix} \\
&= \begin{bmatrix} u_{0,0} & 0 & \cdots & 0 \\ u_{0,1} & u_{1,1} & \ddots & \vdots \\ \vdots & \vdots & \ddots & 0 \\ u_{0,M-2} & u_{1,M-2} & \cdots & u_{M-2,M-2} \\ u_{0,M-1} & u_{1,M-1} & \cdots & u_{M-2,M-1} \end{bmatrix}.
\end{aligned}
\tag{10}
$$

Furthermore, Chen and Amaratunga introduced a lifting factorization of an $M \times M$ Householder matrix [14]:

$$
\begin{aligned}
&\mathbf{H}_M[\mathbf{u}_k] = \\
&\begin{bmatrix} \mathbf{I}_r & \begin{bmatrix} \alpha_{k,0} \\ \vdots \\ \alpha_{k,M-1} \end{bmatrix} & \mathbf{0} \\ \mathbf{0} & & \mathbf{I}_{M-r-1} \end{bmatrix} \begin{bmatrix} \mathbf{I}_r & & \mathbf{0} \\ [\beta_{k,0} & \cdots & \beta_{k,M-1}] \\ \mathbf{0} & & \mathbf{I}_{M-r-1} \end{bmatrix} \\
&\cdot \begin{bmatrix} \mathbf{I}_r & \begin{bmatrix} -\alpha_{k,0} \\ \vdots \\ -\alpha_{k,M-1} \end{bmatrix} & \mathbf{0} \\ \mathbf{0} & & \mathbf{I}_{M-r-1} \end{bmatrix},
\end{aligned}
\tag{11}
$$

where

$$\alpha_{k,l} = \begin{cases} 1 & (l = r) \\ \frac{u_{k,l}}{u_{k,r}} & (\text{otherwise}) \end{cases} \tag{12}$$

$$\beta_{k,l} = \begin{cases} -1 & (l = r) \\ -2u_{k,l}u_{k,r} & (\text{otherwise}) \end{cases}, \tag{13}$$

and r for $u_{k,r} \neq 0$ is selected. It has $3(M-1)$ adders, $3(M-1)$ multipliers, and three process steps. Consequently, an $M \times M$ orthogonal matrix can be factorized into $3(M-1)$ lifting matrices after being factorized into $(M-1)$ cascading Householder matrices.

Fig. 2. Test images: (left-right) 512×512 8-bit grayscale images in [15] (*Barbara*, *Boat*, *Finger*, *Goldhill*, *Lena*, and *Room*) and 1024×1024 8-bit clipped grayscale images in [16] (*Bike*, *Cafe*, *Car*, *Falls*, *Sakura*, and *Woman*).

3. FOUR-CHANNEL LIFTING-HOUSEHOLDER-BASED HADAMARD TRANSFORM (LIFTH^{2}T)

3.1. Lifting-Householder Factorization of a Particular 4×4 Symmetric Orthogonal Matrix

Let $\mathbf{R}_\theta$ be a rotation matrix with an arbitrary rotation angle θ:

$$\mathbf{R}_\theta = \begin{bmatrix} \cos\theta & \sin\theta \\ \sin\theta & -\cos\theta \end{bmatrix} = \begin{bmatrix} c_\theta & s_\theta \\ s_\theta & -c_\theta \end{bmatrix}. \tag{14}$$

A particular 4×4 symmetric orthogonal matrix $\boldsymbol{\mathcal{S}}_\theta$ with four rotation matrices in Eq. (14) is obtained as follows:

$$\boldsymbol{\mathcal{S}}_\theta = \mathbf{R}_\theta \otimes \mathbf{R}_\theta = \begin{bmatrix} c_\theta^2 & c_\theta s_\theta & c_\theta s_\theta & s_\theta^2 \\ c_\theta s_\theta & -c_\theta^2 & s_\theta^2 & -c_\theta s_\theta \\ c_\theta s_\theta & s_\theta^2 & -c_\theta^2 & -c_\theta s_\theta \\ s_\theta^2 & -c_\theta s_\theta & -c_\theta s_\theta & c_\theta^2 \end{bmatrix}. \tag{15}$$

Since any $M \times M$ orthogonal matrix can be factorized into $(M-1)$ cascading Householder matrices as described in Sec. 2.2, the symmetric orthogonal matrix $\boldsymbol{\mathcal{S}}_\theta$ in Eq. (15) can be also factorized into them. Fortunately, the symmetric orthogonal matrix $\boldsymbol{\mathcal{S}}_\theta$ is easily composed of only a 4×4 permutation matrix $\mathbf{P}_4$ and a 4×4 Householder matrix $\mathbf{H}_4[\mathbf{u}_\theta]$:

$$\boldsymbol{\mathcal{S}}_\theta = \mathbf{P}_4 \mathbf{H}_4 [\mathbf{u}_\theta], \tag{16}$$

where

$$\mathbf{u}_\theta = \frac{1}{\sqrt{2}} \begin{bmatrix} \pm s_\theta & \mp c_\theta & \mp c_\theta & \mp s_\theta \end{bmatrix}^T. \tag{17}$$

This study selects $r = 0$ for simplicity though we can select the other rs. The Householder matrix $\mathbf{H}_4[\mathbf{u}_\theta]$ in Eq. (16) can be factorized into three lifting matrices as follows:

$$\mathbf{H}_4[\mathbf{u}_\theta] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -\alpha_\theta & 1 & 0 & 0 \\ -\alpha_\theta & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & \beta_\theta & \beta_\theta & \gamma_\theta \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ \alpha_\theta & 1 & 0 & 0 \\ \alpha_\theta & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}, \tag{18}$$

where

$$\alpha_\theta = \frac{c_\theta}{s_\theta}, \quad \beta_\theta = c_\theta s_\theta, \quad \text{and} \quad \gamma_\theta = s_\theta^2. \tag{19}$$

It has nine adders, four multipliers, and three process steps. To obtain a multiplierless structure, any lifting coefficient with an irrational (floating-point) value must be approximated to a rational (dyadic) coefficient $n/2^b$. The resulting multiplierless structure will yield fast implementations at the expense of decreasing the performance of the transform.

Table 1. Number of operations and process steps.

	IntHT [1]	LiftHT [9]	LiftH^{2}T Prop.
Adder	**8**	10	9
Shifter	**0**	1	1
Multiplier	**0**	**0**	**0**
Process Step	**2**	7	3
Dynamic Range	Ignoring	**Considering**	**Considering**

3.2. Application to Four-Channel Hadamard Transform

A four-channel LiftH^{2}T can be derived by using the lifting-Householder factorization of a particular 4×4 symmetric orthogonal matrix described in Sec. 3.1. The symmetric orthogonal matrix $\boldsymbol{\mathcal{S}}_\theta$ when $\theta = \pi/4$ in Eq. (15) is a special case, i.e., a four-channel HT $\boldsymbol{\mathcal{H}}_4$:

$$\boldsymbol{\mathcal{H}}_4 = \boldsymbol{\mathcal{S}}_{\frac{\pi}{4}} = \mathbf{P}_4 \mathbf{H}_4[\mathbf{u}_{\frac{\pi}{4}}] = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & -1 & 1 & -1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix}, \tag{20}$$

where

$$\mathbf{u}_{\frac{\pi}{4}} = \frac{1}{2} \begin{bmatrix} \pm 1 & \mp 1 & \mp 1 & \mp 1 \end{bmatrix}^T. \tag{21}$$

The following multiplierless lifting structure can be easily derived without any approximation (see the bottom-right of Fig. 1).

$$\mathbf{H}_4[\mathbf{u}_{\frac{\pi}{4}}] = \begin{bmatrix} 1 & 0 & 0 & 0 \\ -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ -1 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -1 & \frac{1}{2} & \frac{1}{2} & \frac{1}{2} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}, \tag{22}$$

where

$$\alpha_{\frac{\pi}{4}} = 1 \quad \text{and} \quad \beta_{\frac{\pi}{4}} = \gamma_{\frac{\pi}{4}} = \frac{1}{2}. \tag{23}$$

It has only nine adders, one shifter, and three process steps.

4. EXPERIMENTAL RESULTS

This study compared the four-channel LiftH^{2}T with IntHT and LiftHT.

4.1. Number of Operations and Process Steps

Table 1 shows the number of operations and process steps. IntHT has the fewest operations and process steps, though it ignores the dynamic range, unlike LiftHT and LiftH^{2}T. Moreover, LiftH^{2}T has one fewer adder and four fewer process steps than LiftHT. Since the processing usually uses many HTs, LiftH^{2}T is efficient even if it only has one fewer operation or one fewer process step in each HT. The next subsection will prove that it is efficient.

Table 3. Lossy image coding results (PSNR [dB]).

Test Images	Bitrate [bpp]	IntHT [1]	LiftHT [9]	LiftH^{2}T Prop.	Test Images	Bitrate [bpp]	IntHT [1]	LiftHT [9]	LiftH^{2}T Prop.
	0.25	25.504	26.696	**26.720**		0.25	23.656	24.231	**24.255**
Barbara	0.50	29.372	30.422	**30.435**	*Bike*	0.50	27.683	29.256	**29.280**
	1.00	34.823	35.111	**35.136**		1.00	32.452	33.849	**33.871**
	0.25	26.824	27.576	**27.605**		0.25	19.058	20.195	**20.211**
Boat	0.50	30.400	30.857	**30.892**	*Cafe*	0.50	22.294	23.353	**23.360**
	1.00	34.183	34.350	**34.371**		1.00	26.463	27.517	**27.523**
	0.25	22.048	22.893	**22.901**		0.25	32.527	33.560	**33.628**
Finger	0.50	24.743	25.485	**25.493**	*Car*	0.50	36.444	37.218	**37.238**
	1.00	28.713	29.007	**29.045**		1.00	40.817	40.837	**40.854**
	0.25	28.360	28.710	**28.856**		0.25	30.238	31.372	**31.401**
Goldhill	0.50	30.860	31.177	**31.184**	*Falls*	0.50	34.063	35.174	**35.183**
	1.00	34.177	34.447	**34.458**		1.00	38.725	39.831	**39.846**
	0.25	30.287	31.557	**31.572**		0.25	27.839	28.220	**28.251**
Lena	0.50	34.046	35.035	**35.066**	*Sakura*	0.50	32.415	32.290	**32.335**
	1.00	37.513	38.355	**38.380**		1.00	37.395	38.573	**38.621**
	0.25	27.300	28.147	**28.203**		0.25	28.638	29.106	**29.165**
Room	0.50	31.922	32.338	**32.366**	*Woman*	0.50	31.532	32.028	**32.068**
	1.00	37.006	38.528	**38.537**		1.00	35.489	35.707	**35.723**

Table 2. Lossless image coding results (LBR [bpp]).

Test Images	IntHT [1]	LiftHT [9]	LiftH^{2}T Prop.
Barbara	8.071	**4.823**	4.824
Boat	8.421	5.140	**5.138**
Finger	9.027	5.723	**5.721**
Goldhill	8.354	5.073	**5.067**
Lena	7.875	4.619	**4.615**
Room	7.539	4.362	**4.360**
Bike	8.402	5.129	**5.126**
Cafe	9.304	6.002	**6.001**
Car	7.136	4.003	**4.002**
Falls	7.142	4.059	**4.049**
Sakura	7.514	4.341	**4.336**
Woman	8.023	**4.767**	4.768

Fig. 3. Particular area of an image *Lena* reconstructed with different HTs when the bit rate is 0.25 bpp: (left-right) IntHT, LiftHT in the JPEG XR, and LiftH^{2}T.

4.2. Lossy-to-Lossless Image Coding

This subsection compares the results of different HTs in lossy-to-lossless image coding based on the LT of the JPEG XR and a simple quadtree-based embedded image coder EZW-IP [17] by using the following lossless bitrate (LBR) [bpp] and peak signal-to-noise ratio (PSNR) [dB] in lossy-to-lossless image coding:

$$\text{LBR [bpp]} = \frac{\text{Total number of bits [bit]}}{\text{Total number of pixels [pixel]}} \tag{24}$$

$$\text{PSNR [dB]} = 10\log_{10}\left(\frac{255^2}{\text{MSE}}\right), \tag{25}$$

where MSE is the mean squared error. A two-level decomposition for the LT of the JPEG XR was employed, where only the core transform (reversible DCT) was used in the second stage in accordance with the specification. Six 512×512 8-bit grayscale images in [15] and six 1024×1024 8-bit clipped grayscale images in [16] were selected for this experiment (see Fig. 2).

Table 2, Table 3, and Fig. 3 show the results of the lossless and lossy image coding. LiftH^{2}T basically outperformed IntHT and LiftHT although the differences between it and LiftHT were trivial.

5. CONCLUSION

This study presented a four-channel LiftH^{2}T having only nine adders, one shifter, and three process steps, that was obtained by using a lifting-Householder factorization of a particular 4×4 symmetric orthogonal matrix and its application to a four-channel HT. Since the processing usually uses many HTs, LiftH^{2}T is efficient even if it only has one fewer operation or one fewer process step in each HT. Moreover, LiftH^{2}T basically outperformed IntHT and LiftHT in the JPEG XR at lossy-to-lossless image coding in spite of having one fewer adder and four fewer process steps than LiftHT, because it considers the dynamic range and has less rounding error. It can be used in not only the LT of the JPEG XR but also various signal processing and communication applications.

6. REFERENCES

[1] B. J. Fino and V. R. Algazi, "Unified matrix treatment of the fast Walsh-Hadamard transform," *IEEE Trans. Computers*, vol. C-25, no. 11, pp. 1142–1146, Nov. 1976.

[2] S. Kyochi and Y. Tanaka, "General factorization of conjugate-symmetric Hadamard transforms," *IEEE Trans. Signal Process.*, vol. 62, no. 13, pp. 3379–3392, July 2014.

[3] K. R. Rao and P. Yip, *Discrete Cosine Transform Algorithms*, Academic Press, 1990.

[4] G. K. Wallace, "The JPEG still picture compression standard," *IEEE Trans. Consum. Electr.*, vol. 38, no. 1, pp. xviii–xxxiv, Feb. 1992.

[5] 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.

[6] 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.

[7] 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.

[8] T. Suzuki, S. Kyochi, Y. Tanaka, M. Ikehara, and H. Aso, "Multiplierless lifting based FFT via fast Hartley transform," in *Proc. of ICASSP'13*, Vancouver, Canada, May 2013, pp. 5603–5607.

[9] 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.

[10] 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.

[11] M. Iwahashi, T. Orachon, and H. Kiya, "Three dimensional discrete wavelet transform with deduced number of lifting steps," in *Proc. of ICIP'13*, Melbourne, Australia, Sep. 2013, pp. 1651–1654.

[12] T. Suzuki and M. Ikehara, "Simplified DCT-lifting-based reversible lapped transforms using parallel processing of two same type lapped transforms," in *Proc. of ICIP'14*, Paris, France, Oct. 2014, pp. 5586–5590.

[13] P. P. Vaidyanathan, *Multirate Systems and Filter Banks*, Englewood Cliffs, NJ: Prentice Hall, 1992.

[14] Y. J. Chen and K. S. Amaratunga, "M-channel lifting factorization of perfect reconstruction filter banks and reversible M-band wavelet transforms," *IEEE Trans. Circuits Syst. II*, vol. 50, no. 12, pp. 963–976, Dec. 2003.

[15] "The USC-SIPI Image Database," *Univ. Southern California, Signal and Image Processing Institute [Online]*, Available: http://sipi.usc.edu/database/.

[16] "JPEG core experiment for the evaluation of JPEG XR image coding," *EPFL, Multimedia Signal Processing Group [Online]*, Available: http://mmspg.epfl.ch/iqa.

[17] Z. Liu and L. J. Karam, "An efficient embedded zerotree wavelet image codec based on intraband partitioning," in *Proc. of ICIP'00*, Vancouver, British Columbia, Canada, Sep. 2000.