Avoidance of Singular Point in Integer Orthonormal Transform for Lossless Coding

Masahiro Iwahashi, *Senior Member, IEEE*
Masanori Ogawa, *non member,*
and Hitoshi Kiya, *Senior Member, IEEE*

Abstract - In this correspondence, the singular point (SP) problem peculiar to an integer orthonormal transform is discussed, and compatibility of the integer transform with the corresponding real-valued transform is improved. To avoid the SP problem, we introduce two-point permutations of order and sign of signals. Since it satisfies the commutative law, it becomes possible to reduce computational cost to find the best combination of the permutations which minimizes errors due to rounding of signals inside the integer transform.

Index Terms— error, KLT, reversible, coding, color

I. INTRODUCTION

Over the past few decades, a considerable number of studies have been made on the discrete cosine transform (DCT) to develop efficient compression algorithms [1-2]. They have been widely utilized to compress massive data amount of digital audio, image and video signals for digital communications and storages. Ever since the adoption of the DCT to the international standards JPEG and MPEG, numerous studies have been made on constructing 'lossy' compression of digital wave forms [2]. These algorithms can control quality of a reconstructed signal in exchange for compression rate. However, it is inevitable to have some distortions in the reconstructed signal.

For those who wish to have a waveform exactly the same as the original signal, 'lossless' coding is preferable. The JPEG and the JPEG-LS utilized the DPCM [3] and the edge-directed adaptive prediction [4], respectively. The JPEG 2000 adopted the 5/3 integer discrete wavelet transform (DWT) [5]. However, these are not compatible with the conventional DCT widely used for lossy coding. In this correspondence, we improve compatibility of an integer transform designed for lossless coding with the real valued transform utilized for lossy coding.

Recently, various types of integer DCTs have been proposed [6]. Benefitting from its ladder structure [7], which is essentially the same as the lifting structure excluding delays [8,9], the integer DCT guarantees lossless reconstruction of signals even though signal values are rounded into integers inside the transform. However, little attention has been given to the point that some coefficient values of the integer transform in this structure have a singular point (SP). In our previous report in [10], we have pointed out that a coefficient value becomes huge and it magnifies rounding errors. It degrades coding efficiency and compatibility with the corresponding real valued transform. This is the SP problem of an integer transform.

In this correspondence, we discuss how to minimize the SP problem in an integer orthonormal transform. We deal with the Karhunen-Loève transform (KLT) for color images as an example [11,12]. When it is applied to neighboring pixels in each of color component of an image, it can be replaced by the DCT since it is an asymptotic approximation of the KLT to the auto-regressive model [1]. In this case, the SP problem is latent as the basis functions are fixed. On the contrary, when the KLT is applied between the color components, the SP problem is revealed. This is because correlations between the components vary depending on each of input images.

To cope with the SP problem, permutations of 'order' of signals were introduced to an integer DCT in [13], and an integer KLT of RGB color components in [10], respectively. However, permutations of 'sign' of signals were not introduced. Permutations of both of order and sign were introduced to integer color transforms in [14]. These transforms convert RGB to other color spaces. Note that these permutations do not generate any errors.

In [14], it was confirmed that the existing method could ease the SP problem. However, it requires to evaluate rounding errors of all the possible combinations of the permutations. This is because rotation angles in a KLT are dependent to each other and a three-point rotation is not commutative. In spite of its time consuming procedure, it remained unknown where an SP was moved due to a permutation.

In this correspondence, we introduce two-point permutations of order and sign. Utilizing its composition property, we make it clear that an SP can be shifted by one of {0, $\pi/2$, π or $-\pi/2$} radian by a two-point permutation. Furthermore, complexity of searching the best combination of the permutations is dramatically reduced. This is because the rotation angles are mutually independent and a rotation in a two-dimensional plane satisfies the commutative low. In addition, our method requires calculation of distance between an angle and the SP, instead of evaluating variance of the errors.

In our simulation, benefit of introducing the permutation is firstly demonstrated for coding and compatibility, comparing to the case without any permutation. Secondly, computational cost for finding the best permutation is compared to the existing method in [14]. Finally, we conclude that the SP problem can be avoided at reduced computational cost by the proposed method.

II. INTEGER KLT AND ITS SINGULAR POINT

A. Factorization of a KLT into 2×2 Matrices

A three-point KLT converts a set of input values **x** into a set of output values **y** as

$$\mathbf{y} = \mathbf{K}_3^T \mathbf{x} \tag{1}$$

where

$$\mathbf{y} = [y_i, \quad y_j, \quad y_k]^T, \qquad \mathbf{x} = [x_i, \quad x_j, \quad x_k]^T .$$

A matrix $\mathbf{K}_3$ is determined as eigenvectors of the covariance matrix $\mathbf{R_X}$ defined as

$$\mathbf{R_X} = E[\mathbf{x}\mathbf{x}^T], \tag{2}$$

where $E[\ ]$ denotes ensemble average of each component. The KLT converts $\mathbf{R_X}$ into

$$\begin{aligned}\mathbf{R_Y} &= E[\mathbf{y}\mathbf{y}^T] = \mathbf{K}_3^T E[\mathbf{x}\mathbf{x}^T]\mathbf{K}_3 = \mathbf{K}_3^T \mathbf{R_X} \mathbf{K}_3 \\ &= diag[\lambda_i, \quad \lambda_j, \quad \lambda_k],\end{aligned} \tag{3}$$

where $[\lambda_i, \lambda_j, \lambda_k]$ denote eigen-values. It decorrelates the input values $[x_i, x_j, x_k]$ using their covariance matrix $\mathbf{R_X}$.

The 3×3 matrix $\mathbf{K}_3$ in Eq.(1) can be factorized into a product of 2×2 matrices $\mathbf{G}(\varphi_i)$, $\mathbf{G}(\varphi_j)$ and $\mathbf{G}(\varphi_k)$ where

$$\mathbf{G}(\varphi) = \begin{bmatrix} \cos\varphi & -\sin\varphi \\ \sin\varphi & \cos\varphi \end{bmatrix}. \tag{4}$$

Fig.1(a) illustrates an example of the factorization of $\mathbf{K}_3$. In this case, the rotation angles $\{\varphi_i, \varphi_j, \varphi_k\}$ are related to elements of $\mathbf{K}_3$ as

Manuscript received Aug. *, 2011; revised Oct. *, 2011 and Dec. *, 2011. Published March *, 2012.

Masahiro Iwahashi is with Nagaoka University of Technology, Nagaoka, Niigata, 940-2188 Japan (e-mail: iwahashi@vos.nagaokaut.ac.jp).
Hitoshi Kiya is with Tokyo Metropolitan University, Hino, Tokyo, 191-0065, Japan.

$$\boldsymbol{\varphi} = [\varphi_i, \quad \varphi_j, \quad \varphi_k] = \left[\arctan\frac{(k)_{2,3}}{(k)_{3,3}}, \quad \arcsin(k)_{1,3}, \quad \arctan\frac{(k)_{1,2}}{(k)_{1,1}}\right] \qquad (5)$$

where $(k)_{n,m}$ denotes an element of $\mathbf{K}_3$ at n-th row and m-th column.

In this correspondence, we assume $\mathbf{x}$ to be a set of RGB color components of a color image signal. The component set $\mathbf{x}$ is decorrelated by the KLT and each component of $\mathbf{y}$ is fed into an integer transform such as the integer DCT or the 5/3 integer DWT. Since correlations between R, G and B vary depending on each input image, a set of the rotation angles $\boldsymbol{\varphi}$ in Eq.(5) is assumed to be attached to a compressed data as an overhead.

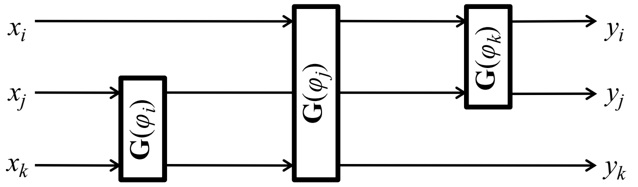


Fig.1 A 3×3 KLT matrix can be factorized into 2×2 rotation matrices $\mathbf{G}(\varphi_i)$, $\mathbf{G}(\varphi_j)$ and $\mathbf{G}(\varphi_k)$.

B. Singular Point Problem

Fig.2 illustrates a pair of integer transforms in the ladder structure [7]. The rotation $\mathbf{G}(\varphi)$ in Eq.(4) can be implemented in this form. Output signals in the figure are calculated as

$$\begin{bmatrix} x'_2 \\ y_1 \\ y_2 \end{bmatrix} = \begin{bmatrix} x_2 + R[f_1 x_1] \\ x_1 + R[f_2 x'_2] \\ x'_2 + R[f_3 y_1] \end{bmatrix}, \quad \begin{bmatrix} y'_2 \\ w_1 \\ w_2 \end{bmatrix} = \begin{bmatrix} y_2 - R[f_3 y_1] \\ y_1 - R[f_2 y'_2] \\ y'_2 - R[f_1 w_1] \end{bmatrix} \qquad (6)$$

where $R[\cdot]$ denotes a rounded value of an argument to the nearest integer, *i.e.* $R[f_1 x_1] := \lfloor f_1 x_1 + 1/2 \rfloor$. Note that it generates rounding errors. In this structure, its output $[w_1 \ w_2]$ is exactly the same as its integer input $[x_1 \ x_2]$ in spite of the rounding since the rounding errors are totally cancelled between input of the forward transform $\mathbf{F}(\theta)$ and output of the backward transform $\mathbf{F}^{-1}(\theta)$.

Neglecting the rounding errors, Eq.(6) is described as a real-valued transform as

$$\begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = \mathbf{F}(\theta) \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \mathbf{F}^{-1}(\theta) \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}, \qquad (7)$$

where

$$\mathbf{F}(\theta) = \begin{bmatrix} 1 & 0 \\ f_3 & 1 \end{bmatrix} \begin{bmatrix} 1 & f_2 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ f_1 & 1 \end{bmatrix}. \qquad (8)$$

When each of the rotations $\mathbf{G}$ in Fig.1 is replaced by the integer transform $\mathbf{F}$ in Fig.2, comparing Eq.(4) and Eq.(8), $\mathbf{G}(\varphi) = \mathbf{F}(\theta)$ implies

$$[f_1 \quad f_2 \quad f_3] = \left[\tan\frac{\varphi}{2}, \quad -\sin\varphi, \quad \tan\frac{\varphi}{2}\right]. \qquad (9)$$

The equation above indicates that the absolute value of f_1 and f_3 are close to infinity when φ is close to π [rad]. This is the singular point (SP) problem we are discussing in this correspondence.

As described above, the rotation angle φ is determined by correlations of an input signal. When it is occasionally close to π [rad], a coefficient value becomes extremely huge, and therefore it magnifies the rounding error. It degrades lossless coding efficiency of the integer transform and also compatibility with the real-valued transform. We confirm it in **IV.A.** and **IV.B.**.

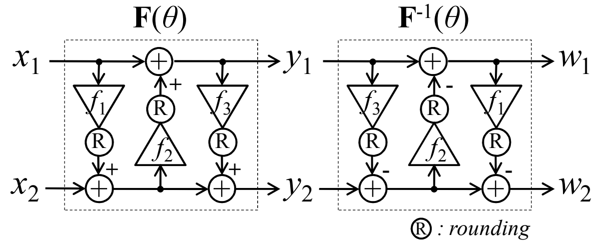


Fig.2 A pair of 2×2 integer rotation transforms. $\mathbf{F}$ and $\mathbf{F}^{-1}$ denote forward transform and backward transform, respectively

C. Existing Method

Fig.3(a) illustrates an existing approach reported in [14]. Permutations of order and sign of signals are introduced to cope with the SP problem. Signals are permutated by the matrices $\mathbf{Q}_a$ and $\mathbf{Q}_b$ before and after the rotations $\mathbf{G}(\theta_i)$, $\mathbf{G}(\theta_j)$ and $\mathbf{G}(\theta_k)$. Each of them is given as a product of permutations of sign $\mathbf{S}_3$ and permutations of order $\mathbf{P}_3$, namely,

$$\mathbf{Q}_a, \mathbf{Q}_b \in \mathbf{S}_3 \mathbf{P}_3, \qquad (10)$$

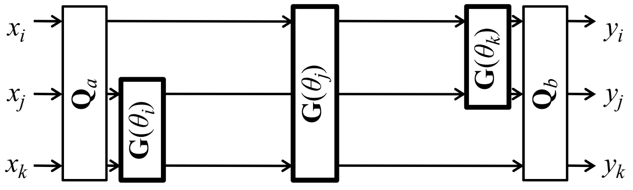
where

$$\mathbf{S}_3 = \begin{bmatrix} (-1)^i & 0 & 0 \\ 0 & (-1)^j & 0 \\ 0 & 0 & (-1)^k \end{bmatrix}, \quad i, j, k \in \{0,1\}, \qquad (11)$$

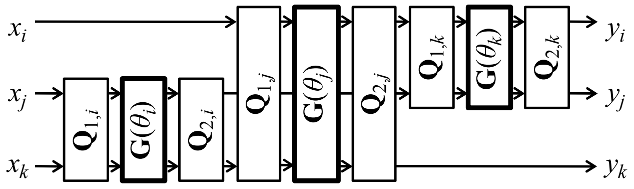
and

$$\mathbf{P}_3 \in \left\{ \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 1 & 0 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 0 & 0 \\ 0 & 0 & 1 \\ 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 0 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix}, \begin{bmatrix} 0 & 0 & 1 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \end{bmatrix} \right\}. \qquad (12)$$

The rotation angles $\boldsymbol{\theta} = [\theta_i, \theta_j, \theta_k]$ in Fig.3(a) are determined under the constrain that Fig.3(a) generates the same output $\mathbf{y}$ as the Fig.1 to the same input $\mathbf{x}$ when the rounding errors are neglected. Especially, when $\mathbf{Q}_a = \mathbf{Q}_b = \mathbf{I}_3$ (the identity matrix), $\boldsymbol{\theta} = \boldsymbol{\varphi}$ holds. Except this case, the rotation angle $\boldsymbol{\theta}$ varies depending on selection of $\mathbf{S}_3$ and $\mathbf{P}_3$.



(a) Existing method



(b) Proposed method

Fig.3 The singular point problem can be avoided by introducing the permutation matrices **Q**.

D. Computational Cost for Optimization

In the existing method, one pair of permutations $\mathbf{S}_3$ and $\mathbf{P}_3$ is determined so that the total amount of variance of the rounding error becomes the minimum. Since $\mathbf{S}_3$ and $\mathbf{P}_3$ have 2^3 and 3! candidates, respectively, the existing method selects the best one from all the $(2^3\ 3!)^2$ /2 = 1,152 combinations (the cases when the determinant is equal to unity are excluded [14]). This is because the rotation angles are dependent to each other and a three-point rotation is not commutative. For example, changing order of x_i and x_j in Fig.3(a) varies the rotation angles as

$$\begin{aligned}&\theta_i = \arctan\left(\frac{\tan\varphi_j}{\cos\varphi_i}\right), \quad \theta_j = \arcsin\left(\sin\varphi_i \cos\varphi_j\right)\\&\theta_k = \arctan\left(\frac{\tan\varphi_i \sin\varphi_j + \tan\varphi_k}{\tan\varphi_i \sin\varphi_j \tan\varphi_k - 1}\right).\end{aligned} \tag{13}$$

It is obvious that changing an angle has an effect on other angles. Therefore, The existing method requires investigating the variance of rounding error for all the combinations of the permutations.

The computational cost for the optimization described above affects on designing time and implementation time of the integer transform. First of all, when the optimization is performed for averaged model of various input images, correlation among the color components of each image is not completely utilized. This is not the case we are discussing. On the contrary, when the optimization is performed for each of input images, we should include the rotation angles $\boldsymbol{\varphi}$ in Eq.(5) into the overhead as described in section **II.A.**. Note that we need not only the angles to include into the overhead, but also the best combination of the permutations.

In this correspondence, we are reducing computational cost of determining the best combination. When the best combination is described in the overhead information, it doesn't reduce implementation time of the transform, but it reduces designing time. On the contrary, when it is not described in the overhead, the decoder should find the best combination in real time. In this case, our approach reduces the time for implementation.

III. PROPOSED METHOD

A. Utilization of Two-Point Permutations

Fig.3(b) illustrates the proposed method. We introduce two-point permutations instead of the three-point ones in the existing method. Note that both of the existing method and the proposed method have 9 rounding operations in total since each of $\mathbf{G}(\theta_q)$ has 3 and the permutation has zero. The 2×2 permutation matrix $\mathbf{Q}_{p,q}$, $p \in \{1, 2\}$, $q \in \{i, j, k\}$ is a product of a permutation of sign $\mathbf{S}_2$, a permutation of order $\mathbf{P}_2$ and the identity matrix $\mathbf{I}_2$ where

$$\mathbf{P}_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, \quad \mathbf{S}_2 = \begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix}, \quad \mathbf{I}_2 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}. \tag{14}$$

Since these matrices have the property:

$$\mathbf{P}_2^2 = \mathbf{I}_2, \quad \mathbf{S}_2^2 = \mathbf{I}_2, \quad (\mathbf{P}_2\mathbf{S}_2)^2 = (-\mathbf{S}_2\mathbf{P}_2)^2 = -\mathbf{I}_2, \tag{15}$$

the permutations in our method have a structure of the dihedral group of order 8. In addition to the Givens rotation in Eq.(4), denoting the Householder reflection as

$$\mathbf{H}(\varphi) = \begin{bmatrix} \cos\varphi & \sin\varphi \\ \sin\varphi & -\cos\varphi \end{bmatrix}, \tag{16}$$

the permutations in Eq.(14) are expressed as

$$\mathbf{P}_2 = \mathbf{H}(\pi/2), \quad \mathbf{S}_2 = \mathbf{H}(\pi), \quad \mathbf{I}_2 = \mathbf{G}(0). \tag{17}$$

Since the Givens rotation **G** and the Householder reflection **H** have the following composition property (*proof is given in appendix*):

$$\begin{cases} \mathbf{G}(\beta)\mathbf{G}(\alpha) = \mathbf{G}(\beta+\alpha) \\ \mathbf{H}(\beta)\mathbf{G}(\alpha) = \mathbf{H}(\beta-\alpha) \\ \mathbf{G}(\beta)\mathbf{H}(\alpha) = \mathbf{H}(\beta+\alpha) \\ \mathbf{H}(\beta)\mathbf{H}(\alpha) = \mathbf{G}(\beta-\alpha) \end{cases}, \tag{18}$$

it became clear that there are only four independent permutations:

$$\begin{cases} \mathbf{I}_2 & = \mathbf{G}(0) \\ \mathbf{P}_2\mathbf{S}_2 & = \mathbf{H}(\pi/2)\mathbf{H}(\pi) \quad = \mathbf{G}(-\pi/2) \\ \mathbf{S}_2\mathbf{P}_2\mathbf{S}_2\mathbf{P}_2 & = \mathbf{G}(\pi/2)\mathbf{G}(\pi/2) = \mathbf{G}(\pi) \\ \mathbf{S}_2\mathbf{P}_2 & = \mathbf{H}(\pi)\mathbf{H}(\pi/2) \quad = \mathbf{G}(\pi/2) \end{cases} \tag{19}$$

which are composed of $\mathbf{P}_2$, $\mathbf{S}_2$ and $\mathbf{I}_2$ in Eq.(14). Fig.4 summarizes these four candidates to be selected as the permutation matrix $\mathbf{Q}_{p,q}$ in Fig.3(b).

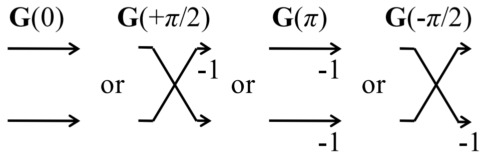


Fig.4 One of four permutations in this figure is selected as the permutation matrices **Q** in the proposed method.

B. Effect of a Permutation on Rotation Angles

Fig.5 illustrates four cases to be selected as $\mathbf{Q}_{2,q}\mathbf{G}(\theta_q)\mathbf{Q}_{1,q}$, $q \in \{i, j, k\}$ in Fig.3(b). These are expressed as

$$\begin{cases} case(1) & \mathbf{G}(\varphi_q) = \mathbf{G}(0)\mathbf{G}(\theta_q) \\ case(2) & \mathbf{G}(\varphi_q) = \mathbf{G}(-\pi/2)\mathbf{G}(\theta_q) \\ case(3) & \mathbf{G}(\varphi_q) = \mathbf{G}(\pi)\mathbf{G}(\theta_q) \\ case(4) & \mathbf{G}(\varphi_q) = \mathbf{G}(\pi/2)\mathbf{G}(\theta_q) \end{cases} \tag{20}$$

for each rotation $\mathbf{G}(\varphi_q)$ in Fig.1. This is equivalent to setting $\mathbf{F}(\theta_q)$ in Fig.2 as $\mathbf{G}(\theta_q)$ in Fig.3(b), and $\mathbf{G}(0)$ as $\mathbf{Q}_{1,q}$ in Fig.3(b). Furthermore, $\mathbf{G}(0)$, $\mathbf{G}(-\pi/2)$, $\mathbf{G}(\pi)$ and $\mathbf{G}(\pi/2)$ is selected as the permutation $\mathbf{Q}_{2,q}$ in case(1), (2), (3) and (4), respectively. Namely, there are only four candidates in Eq.(20) for each rotation $\mathbf{G}(\varphi_q)$ in Fig.1.

Utilizing the property in Eq.(18), each of the cases in Eq.(20) gives the angle as follows

$$\begin{cases} case(1) & \varphi_q = \theta_q \\ case(2) & \varphi_q = \theta_q - \pi/2 \\ case(3) & \varphi_q = \theta_q - \pi \\ case(4) & \varphi_q = \theta_q + \pi/2 \end{cases} . \tag{21}$$

This means that the SP can be shifted by one of {0, $\pi/2$, π or $-\pi/2$} by one of the two-point permutations. Note that there exists an SP at $\theta=\pi$. As a result of our theoretical analysis, it became clear where the SP is moved by the two-point permutation.

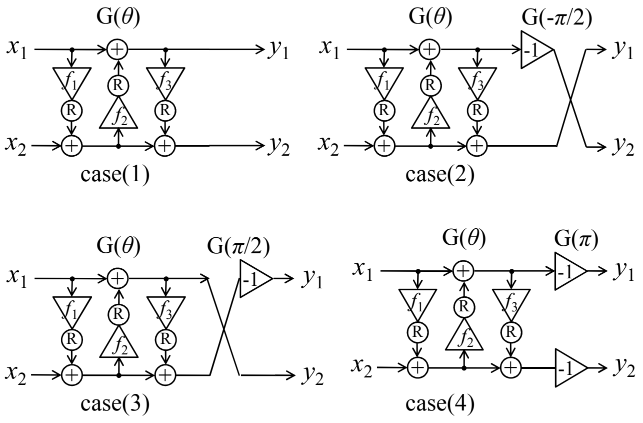


Fig.5 Four candidates for the rotation $\mathbf{G}(\varphi_q)$ in Fig.1.

C. Effect of a Permutation on Error Variance

Denoting the error as e_m generated by the rounding just after the multiplier f_m, $m \in \{1,2,3\}$ in $\mathbf{F}(\theta)$ in Fig.2, its output $[y_1, y_2]$ contains the errors:

$$\begin{bmatrix} e_{y1} \\ e_{y2} \end{bmatrix} = \begin{bmatrix} -\sin\theta & 1 & 0 \\ \cos\theta & \tan\dfrac{\theta}{2} & 1 \end{bmatrix} \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} . \tag{22}$$

Variance of these errors are calculated as

$$\begin{bmatrix} \sigma_{y1}^2 \\ \sigma_{y2}^2 \end{bmatrix} = \frac{1}{12} \begin{bmatrix} 1+\sin^2\theta \\ 1+\cos^2\theta + \tan^2\dfrac{\theta}{2} \end{bmatrix} \tag{23}$$

and therefore their total amount becomes

$$\sigma_{sgl}^2(\theta) = \sigma_{y1}^2 + \sigma_{y2}^2 = \frac{1}{12}\left(3+\tan^2\frac{\theta}{2}\right). \tag{24}$$

Fig.6 illustrates this error variance theoretically calculated as a bold line, and experimental one as a cross. It is theoretically indicated that the error variance has an SP at $\varphi=\pi$ in case (1). It is shifted to $\varphi=\pi/2$, π and $3\pi/2(=-\pi/2)$ in case (2), (3) and (4), respectively.

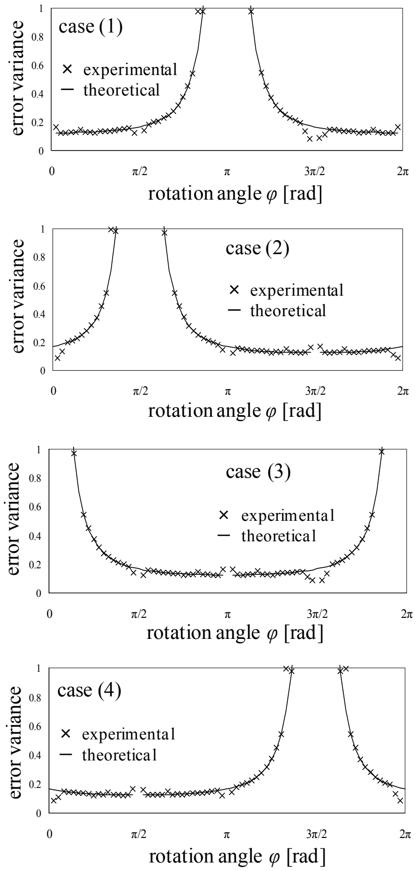


Fig.6 Variance of the rounding errors versus the angle φ_q of the rotation $\mathbf{G}(\varphi_q)$ in Fig.1.

It should be noted that the error variance is a monotonic function with respect to distance between the SP and the angle. Therefore, we can select the best case in Eq.(20) according to the distance. If a given angle φ_q exists in the interval $[-\pi/4, \pi/4)$, case (1) is selected as the optimum form for $\mathbf{G}(\varphi_q)$ in Fig.1, so that the distance becomes the maximum. Similarly case (2), case (3) and case (4) is selected for $\varphi_q \in [5\pi/4, 7\pi/4)$, $[3\pi/4, 5\pi/4)$ and $[\pi/4, 3\pi/4)$, respectively.

D. Computational Cost of the Two Methods

As described in section **II**, the existing method requires 1,152 times calculation of the error variance to determine the best case. On the contrary, in the proposed method, one of four candidates in Fig.5 is determined for each of the three rotations in Fig.1. It is performed independently since a rotation in the two-dimensional plane satisfies the commutative low. It amounts to (4 × 3) =12 times calculation of the distance.

This discussion can be extended to a general case for a transform with order N. According to [15], it is known that an $N \times N$ orthonormal matrix $\mathbf{K}_N$ can be factorized into a product of $N(N-1)/2$ matrices $\mathbf{G}(\varphi_{i,j})$ of order 2 as

$$\mathbf{K}_N = \prod_{i=1}^{N-1} \prod_{j=i+1}^{N} \mathbf{G}(\varphi_{i,j}) . \tag{25}$$

Therefore, computational cost of the searching procedure in the proposed method becomes $4 \times N(N\text{-}1)/2 = 2N(N\text{-}1)$. It dramatically reduces the computational cost of the existing method which is given as $N!\ 2^{2N}/2$. We confirm it in **IV.C.**.

IV. SIMULATION RESULTS

In the simulation below, benefit of introducing the permutation for lossless coding and compatibility is firstly demonstrated comparing to the case without any permutation. Secondly, computational cost of searching the best permutation is compared to the existing method in [14]. Finally, we conclude that the SP problem can be eased off by the proposed method at reduced computational cost.

A. Effectiveness of Permutations on Compatibility

Fig.7 compares the proposed method to the existing method with the total amount of the rounding errors in reconstructed signals. In this experiment, a forward transform of the integer KLT with the permutations is applied to an input image, and reconstructed by the backward transform of the real-valued transform. The error comes from difference between the integer transform and the real-valued transform. This means the compatibility. The peak signal to noise ratio (PSNR) is used as the measure.

When the best combination of the permutations is selected, both of the two methods attain approximately 54 [dB]. The existing method is slightly better than the proposed method. However, these are almost the same. The point we should pay attention is the fact that the PSNR drops to approximately 40 [dB] at the maximum in the worst case when the permutation is not applied. As a result, it was confirmed that introduction of the permutations is effective in both of the existing method and the proposed method to avoid the SP problem.

B. Effectiveness of Permutations on Lossless Coding

Fig.8 compares the proposed method to the existing method with the bit rate per pixel per component. Color components [R, G, B] are decorrelated by the integer KLT first, and then compressed with the 5/3 integer DWT and the EBCOT defined by the JPEG 2000 [5].

In case of the existing method, the bit rate averaged over all the nine kinds of images is 4.14 [bit] at the best, and 4.47 at the worst. For the proposed method, those are 4.14 [bit] and 4.38 [bit] respectively. As a result of this experiment, no significant difference was observed between the two methods. However, it was observed that introducing the permutations saves the bit rate by approximately 0.3 [bit] at the maximum per pixel per component. Necessity of avoiding the SP problem was indicated.

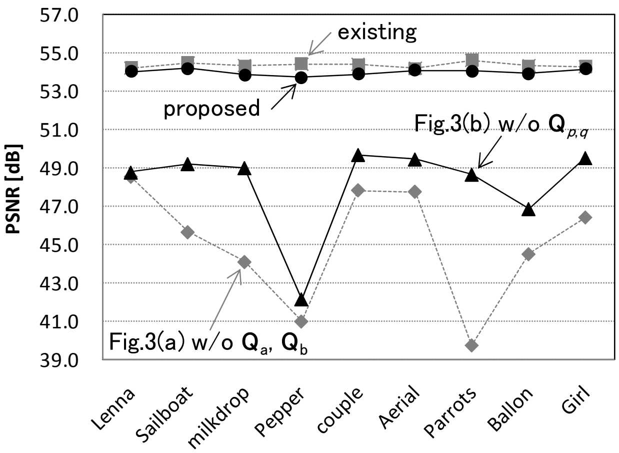


Fig.7 Effectiveness of permutations on compatibility

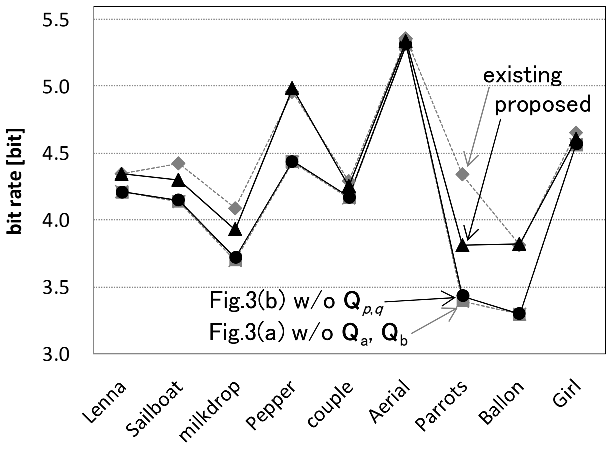


Fig.8 Effectiveness of permutations on lossless coding.

C. Comparison of Computational Cost

Table I compares the existing method in [14] and the proposed method in respect of the execution time to find the best combination of the permutations, measured on the hardware platform of 'Pentium D' with 'MATLAB 2006a' with 'Windows XP'. The table clearly indicates superiority of the proposed method to the existing method in computational cost. It was reduced to approximately 1/570.

Fig.9 illustrates the computational cost to find the best combination versus the order N of an orthonormal transform, estimated in section **III.*D*.**. Finally, it was confirmed that the proposed method dramatically reduces the computational cost to find the best combination of the permutations. It enables to speed up design and implementation of an integer transform which adaptively utilizes correlations between color components of each input image signals.

TABLE I Execution time to find the best combination [sec].

	Lenna	Sailboat	milkdrop	Pepper	couple
Existing	313.89	310.41	306.08	309.02	317.79
Proposed	0.56	0.54	0.53	0.53	0.55
	Aerial	Parrots	Ballon	Girl	Average
Existing	326.68	311.04	307.83	317.68	**313.38**
Proposed	0.57	0.54	0.57	0.55	**0.55**

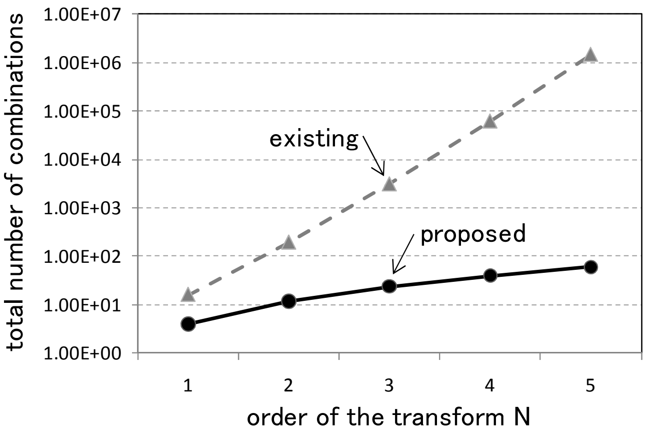


Fig.9 The total number of combinations to be investigated to find the best combination of permutations.

V. CONCLUSIONS

In this correspondence, we discussed the singular point (SP) problem of an integer KLT for decorrelation of RGB color components of image signals. We introduced two-point permutations of order and sign of signals to avoid the SP problem. Analyzing an effect of the permutation on a rotation angle, computational cost of finding the best combination of permutations is dramatically reduced. It contributes to speed up design and implementation of an integer transform adaptive to each input image signals.

Results in this correspondence are limited to three-color components. It is necessary to expand the proposed method to more than three in the future.

VI. REFERENCES

[1] N. S. Jayant, Peter Noll, “Digital coding of waveforms, Principles and applications to speech and video,” Prentice Hall, Signal Processing series, 1984.

[2] ISO/IEC 11172-1:1993, “Information technology -- Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s,” 1993.

[3] JTC1/ SC29, "Information technology -- Digital compression and coding of continuous-tone still images: Requirements and guidelines", ISO/ IEC 10918-1, 1994.

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

[5] D. S. Taubman, M. W. Marcellin, “JPEG 2000 - Image compression fundamentals, standards and practice,” Kluwer Academic Publishers, 2002.

[6] Vladimir Britanak, P.C.Yip, K.R.Rao, “Discrete Cosine and Sine Transforms: General Properties, Fast Algorithms and Integer Approximations”, Academic, Oct.2006.

[7] F. A. M. L. Bruelers, A. W. M. van den Enden, "New networks for perfect inversion and perfect reconstruction," IEEE J-SAC, vol.10, no.1, pp.130-137, Jan.1992.

[8] W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," Technical report 1994:7, Industrial Mathematics Initiative, Department of Mathematics, University of South Carolina, 1994.

[9] H. Kiya, M. Yae, M. Iwahashi, "Linear phase two channel filter bank allowing perfect reconstruction," IEEE Proc. ISCAS, no.2, pp.951-954, May 1992.

[10] M.Iwahashi, K.Oguni, "Three Dimensional Integer Rotation Transform and Improvement of its Compatibility," IEEE International Symposium on Circuits and Systems (ISCAS), pp.2205 - 2208, May 2009.

[11] Y. Kim, J.Jeong, S. Kim, Y. Choe, "Color Plane Ordering to Minimize Mutual Information Loss in RGB Video Coding," International Workshop on Advanced Image Technology (IWAIT), No.S12, pp.155-159, Jan. 2010.

[12] H. Hashim, R.A. Rahman, R. Jarmin, Nasir Taib Mohd, "A Study on RGB Color Extraction of Psoriasis Lesion using Principle Component Analysis (PCA)," International Symposium on Image and Signal Processing and Analysis (ISPA), pp.288 - 292, Sept. 2007.

[13] M. Iwahashi, D. K. Dang, M. Ohnishi, S. Chokchaitam, “A New Structure of Integer DCT Least Sensitive To Finite Word Length Expression of Multipliers”, IEEE International Conference on Image Processing (ICIP), no.II, pp.269-272, Sept. 2005.

[14] Soo-Chang Pei, Jian-Jiun Ding, "Reversible Integer Color Transform," IEEE Trans. Image Processing, Vol.16, Issue 6, pp. 1686- 1691, June 2007.

[15] M. D. Adams, F. Kossentini, R. K. Ward, “Generalized S Transform”, IEEE Transactions on signal processing, Vol.50, No.11, Nov, 2002

VII. APPENDIX

Proof of Eq.(18): Denoting variables as

$$z = x + iy, \quad \bar{z} = x - iy, \quad u_\theta = \cos\theta + i\sin\theta$$

and Eq.(4) and (16) as

$$z \xrightarrow{G(\theta)} u_\theta z, \quad z \xrightarrow{H(\theta)} u_\theta \bar{z},$$

Eq.(18) are given as follows.

$$\begin{cases} \mathbf{G}(\beta)\mathbf{G}(\alpha) \Rightarrow u_\beta(u_\alpha z) = u_\beta u_\alpha z = u_{\beta+\alpha} z \Rightarrow \mathbf{G}(\beta+\alpha) \\ \mathbf{H}(\beta)\mathbf{G}(\alpha) \Rightarrow u_\beta\overline{(u_\alpha z)} = u_\beta \bar{u}_\alpha \bar{z} = u_{\beta-\alpha}\bar{z} \Rightarrow \mathbf{H}(\beta-\alpha) \\ \mathbf{G}(\beta)\mathbf{H}(\alpha) \Rightarrow u_\beta(u_\alpha \bar{z}) = u_\beta u_\alpha \bar{z} = u_{\beta+\alpha}\bar{z} \Rightarrow \mathbf{H}(\beta+\alpha) \\ \mathbf{H}(\beta)\mathbf{H}(\alpha) \Rightarrow u_\beta\overline{(u_\alpha \bar{z})} = u_\beta \bar{u}_\alpha z = u_{\beta-\alpha} z \Rightarrow \mathbf{G}(\beta-\alpha) \end{cases}$$

Q.E.D.