LETTER

Reversible Implementations of Irreversible Component Transforms and Their Comparisons in Image Compression

Junghyeun HWANG[†a)], *Member*, Hisakazu KIKUCHI[†b)], *Fellow*, Shogo MURAMATSU[†], *Senior Member*, Kazuma SHINODA[††], *and* Jaeho SHIN[†††], *Members*

SUMMARY Reversible color component transforms derived by the LU factorization are briefly described. It is possible to obtain an reversible implementation to a given component transform, even if the original transform is irreversible. Some examples are presented and their performances are compared in image compression.
key words: *component transform, reversible transform, image compression, LU factorization*

1. Introduction

Color component transforms are helpful in image compression [1]–[3]. The number of components can exceed three in some applications such as multi-band spectral images and multi-primary color imaging systems [4]. For lossless compression of those multi-component digital images, component transforms should be reversible, even if the computational accuracy is finite.

A general solution to a reversible component transform (RCT) was presented in Ref. [5] as well as round-off error analysis. However the round-off error can reach a comparable level to the excursion of signals being transformed. The amplification of round-off errors can spoil the efficient data compression. Some improvements in approximation accuracy and operation speed for hardware implementations were presented in Ref. [6]. Nevertheless the exhaustive optimization requires a qualified data set and spends a huge number of error evaluations. Their practical applications are still being left for further developments.

The purpose of this work is twofold. The first is to present another reversible implementation that has the fewest parameters at the expense of a modification of the original transformation. The other is to evaluate some practical component transforms derived by the proposed method for color image compression.

Manuscript received November 2, 2011.
Manuscript revised December 9, 2011.
[†]The authors are with the Department of Electrical and Electronic Engineering, Niigata University, Niigata-shi, 950-2181 Japan.
[††]The author is with Tokyo Institute of Technology, Yokohama-shi, 226-8503 Japan, as a JSPS Research Fellow.
[†††]The author is with the Department of Electronic Engineering, Dongguk University, 3-26 Pil-dong, Jung-gu, Seoul 100-715, Korea.
a) E-mail: hwang@telecom0.eng.niigata-u.ac.jp
b) E-mail: kikuchi@eng.niigata-u.ac.jp
DOI: 10.1587/transfun.E95.A.824

2. A Derivation of a Reversible Transform

2.1 The LU Factorization

An n-dimensional vector $\boldsymbol{x}$ is mapped to a different vector $\boldsymbol{y}$ by multiplying an $n \times n$ matrix, $\boldsymbol{A}$. The inverse transformation of $\boldsymbol{y}$ back to $\boldsymbol{x}$ is equivalent to solving a linear equation system, $\boldsymbol{y} = \boldsymbol{Ax}$. The LU factorization is the most effective method to solve a linear equation system [8]. If every leading sub-matrix of $\boldsymbol{A}$ is nonsingular, $\boldsymbol{A}$ has a unique factorization: $\boldsymbol{A} = \boldsymbol{LDU}$, where $\boldsymbol{L}, \boldsymbol{D}$, and $\boldsymbol{U}$ are a unit lower triangular matrix, a diagonal matrix, and a unit upper triangular matrix, respectively. Note that all diagonal entries of $\boldsymbol{L}$ and $\boldsymbol{U}$ are one. The factorization is obtained by the Gaussian forward elimination.

Furthermore, an $n \times n$ triangular matrix can be decomposed into a product of $(n^2 - n)/2$ basic triangular matrices of which at most a single off-diagonal entry is different from zero.

Note that every entry of the n-dimensional input and output vectors is implicitly assumed to be an integer in this work. Once the LDU factorization of a matrix $\boldsymbol{A}$ has been obtained, its forward and inverse transformations share the same computational configuration to guarantee the reversible property in integer-to-integer mapping, as long as all matrix factors have their diagonal entries equal to unity in magnitude.

2.2 Reversible Component Transforms

A simple method for reversible and canonical transformations is presented in this subsection. Assume the forward transform is written by

$$\boldsymbol{y} = \boldsymbol{Ax} = \boldsymbol{LDUx}, \tag{1}$$

where $\boldsymbol{A}$ is nonsingular and its LDU factorization is obtained by the forward elimination. By left-multiplying the inverse of $\boldsymbol{D}$, a new output vector is defined by

$$\tilde{\boldsymbol{y}} = \boldsymbol{D}^{-1}\boldsymbol{y}. \tag{2}$$

It is written as $\tilde{\boldsymbol{y}} = \tilde{\boldsymbol{L}}\boldsymbol{Ux}$, where $\tilde{\boldsymbol{L}} = \boldsymbol{D}^{-1}\boldsymbol{LD}$ is found to be a unit lower triangular matrix of which diagonal entries are unity. As a result, an explicit lifting factorization of a 3×3 matrix is expressed as follows.

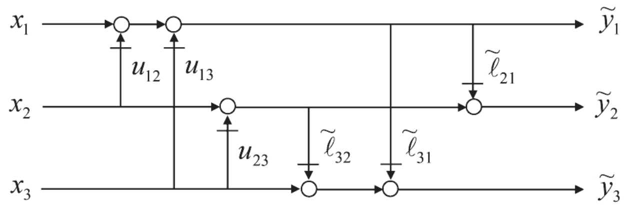

Fig. 1 Modified transform in the lifting form. A summing node is drawn in a circle. A short bar on a lifting step (vertical arrow) denotes a rounding operation. A multiplication factor is written by a lifting step.

$$\tilde{\boldsymbol{L}}\boldsymbol{U} = \tilde{\boldsymbol{L}}_{21}(\tilde{\boldsymbol{L}}_{31}\tilde{\boldsymbol{L}}_{32})\boldsymbol{U}_{23}(\boldsymbol{U}_{13}\boldsymbol{U}_{12}), \tag{3}$$

where just a single off-diagonal entry is not zero in all unit triangular matrix factors in the right-hand side. The suffix of every matrix factor indicates the location of the non-zero entry. The non-zero entry of $\tilde{\boldsymbol{L}}_{ij}$ is given by

$$\tilde{\ell}_{ij} = \ell_{ij}\frac{d_j}{d_i}, \tag{4}$$

where ℓ_{ij} denotes the (i, j) entry of $\boldsymbol{L_{ij}}$ and d_i is the ith diagonal entry of $\boldsymbol{D}$. Since the factorization of Eq. (3) is systematic, it also applies to an $n \times n$ matrix. The number of the basic triangular matrix factors is at most equal to $n^2 - n$.

The reversible implementation shown in Fig. 1 is hence canonical in the sense that the minimum number of independent parameters suffices for the implementation of a transformation. To reduce round-off errors, some rounding operations are combined into a single operation, if a set of successive steps goes to an identical summing node. Actual rounding operations take place on the lifting steps that correspond to $2n-2$ factors separated by parentheses in Eq. (3). The reversibility in lifting is illustrated in Appendix.

Since the output has been modified from the original, this implementation can lose the significance in signal analysis. It is yet independent of the reversibility between the original input and the transformed vector, and it is still useful in data compression.

A further insight is gained through the calculation of a covariance matrix. The covariance matrix of the plain transformed vectors, $\boldsymbol{y} = \boldsymbol{A}\boldsymbol{x}$, is given by $\boldsymbol{C}_{yy} = E\left[(\boldsymbol{y} - \boldsymbol{\mu})(\boldsymbol{y} - \boldsymbol{\mu})^{\dagger}\right]$, where $E[\cdot]$ denotes an expectation value, $\boldsymbol{\mu} = E[\boldsymbol{y}]$, and the superscript $\dagger$ stands for the transposed conjugate. The covariance matrix of $\tilde{\boldsymbol{y}}$ is expressed by

$$\boldsymbol{C}_{\tilde{y}\tilde{y}} = \begin{pmatrix} & \vdots & \\ \cdots & \dfrac{c_{ij} - m_i m_j}{d_i d_j} & \cdots \\ & \vdots & \end{pmatrix}, \tag{5}$$

where c_{ij}, m_i, and d_i denote the (i, j) entry of $\boldsymbol{C}_{yy}$, the ith entry of $\boldsymbol{\mu}$, and the ith diagonal entry of $\boldsymbol{D}$, respectively. For example, if $\boldsymbol{A}$ is given by Karhunen-Loeve transform (KLT) [2], [9] with respect to $\boldsymbol{C}_{xx}$ so that $\boldsymbol{C}_{yy}$ has been diagonalized, and if $\boldsymbol{y}$ is a zero-mean vector in the sense of a stochastic process, $\boldsymbol{C}_{\tilde{y}\tilde{y}}$ is still diagonal. In such a case, all components in the modified vector space are kept to be decorrelated. This fact is advantageous for decorrelation problems such as image compression, where all components can be converted into zero-mean signals in advance.

3. Reversible Implementations

There exist many component transforms such as YUV, YIQ, and Y_{709}CbCr [10]. Most of them are irreversible. Among them, the reversible implementations of two classes of component transforms are presented in this section as a demonstration of the proposed method. The classes are data-independent and data-dependent.

The reversible implementations of data-independent transforms are presented. For example, the following component transform is referred to as EMN [11], which is an irreversible transform.

$$\begin{pmatrix} M \\ N \\ E \end{pmatrix} = \begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & -1 \\ \frac{1}{3} & \frac{1}{3} & \frac{1}{3} \end{pmatrix} \begin{pmatrix} R \\ B \\ G \end{pmatrix}, \tag{6}$$

where rows have been reordered in order of magnitude of diagonal entries to suppress numerical errors in matrix manipulations. Applying the proposed method in the previous section to Eq. (6), one obtains its modified LU factorization as follows.

$$\begin{aligned} \begin{pmatrix} \tilde{M} \\ \tilde{N} \\ \tilde{E} \end{pmatrix} = \begin{pmatrix} M \\ N \\ E \end{pmatrix} &= \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \frac{1}{3} & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & \frac{1}{3} & 1 \end{pmatrix} \\ &\times \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & -1 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 1 & 0 & -1 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} R \\ B \\ G \end{pmatrix} \end{aligned} \tag{7}$$

Since all matrix factors are basic triangular matrices, the reversible implementation in the lifting form is readily found to be expressed by

$$\left.\begin{aligned} &M \leftarrow R - G, \\ &N \leftarrow B - G, \\ &E \leftarrow G + \left\lfloor \frac{M + N}{3} \right\rfloor. \end{aligned}\right\} \tag{8}$$

The transformed signals of M, N, and E in Eq. (8) are written as R, B, and G, respectively, in Table 1 in order to share the same notation with the other RCTs.

In Table 1, $\lfloor\cdot\rfloor$ stands for truncation into an integer. The arrow, $\leftarrow$, denotes a substitution. R, G, and B initially stand for the red, green and blue component of a pixel, respectively. Since lifting is an in-place operation, the transformed values are obtained at their own positions as they have been located. A set of equations should be computed in the order as they appear. Only the forward transforms are given in the table. The inverse transform is obtained by executing the equations in the reverse order after the polarity inversion between plus and minus. The polarity in issue is the leftmost one in each substitution equation. The truncation

Table 1 Lifting forms obtained for some RCTs.

No.	RCT	Expressions in the Lifting Form
1	EMN	$R \leftarrow R - G$, $B \leftarrow B - G$, $G \leftarrow G + \lfloor (R+B)/3 \rfloor$.
2	DCT	$R \leftarrow R - B$, $B \leftarrow B + \lfloor R/2 \rfloor$, $G \leftarrow G - B$, $B \leftarrow B + \lfloor G/3 \rfloor$.
3	Y_{709}CbCr	$R \leftarrow R - \lfloor 0.092B + 0.908G \rfloor$, $B \leftarrow B - G$, $G \leftarrow G + \lfloor 0.092B + 0.213R \rfloor$, $B \leftarrow B - \lfloor 0.234R \rfloor$.
4	YUV	$R \leftarrow R - \lfloor 0.163B + 0.837G \rfloor$, $B \leftarrow B - G$, $G \leftarrow G + \lfloor 0.163B + 0.299R \rfloor$, $B \leftarrow B - \lfloor 0.357R \rfloor$.
5	YIQ	$R \leftarrow R - \lfloor 0.539B + 0.461G \rfloor$, $B \leftarrow B - G$, $G \leftarrow G + \lfloor 0.275B + 0.299R \rfloor$, $B \leftarrow B + \lfloor 0.514R \rfloor$.
6	GMN	$R \leftarrow R - G$, $B \leftarrow B - G$. G is unchanged.
7	LMN	$R \leftarrow R - G$, $B \leftarrow B - G$, $G \leftarrow G + \lfloor (R+B)/4 \rfloor$.
8	YCoCg	$R \leftarrow R - B$, $B \leftarrow B + \lfloor R/2 \rfloor$, $G \leftarrow G - B$, $B \leftarrow B + \lfloor G/2 \rfloor$.
9	rKLT for *bike*	$R \leftarrow R - \lfloor 0.599B + 0.250G \rfloor$, $B \leftarrow B - \lfloor 1.129G \rfloor$, $G \leftarrow G + \lfloor 0.538B + 0.296R \rfloor$, $B \leftarrow B + \lfloor 0.223R \rfloor$.

can be changed into rounding but there are not significant differences in performance.

The other irreversible component transforms are also implemented in reversible ones in the same way. They include 3-point discrete cosine transform (DCT in short), Y_{709}CbCr, YUV, and YIQ. Their explicit lifting forms are given in Table 1.

Well-known RCTs of GMN [1], [11], LMN [11] (JPEG2000 RCT [2]) and YCoCg [12] in H.264AVC/ FRExt are again found by the proposed method as shown in Table 1.

If one wishes to apply KLT as a data-dependent transform, it is possible to find its reversible implementation. After the KLT is computed from the covariance matrix of a given image, say *bike* which is one of ISO/IEC SCID standard images [13], its reversible KLT (rKLT in short) is obtained in the lifting form as listed in Table 1.

4. Performance Evaluations of Representative Component Transforms

Some RCTs obtained by the proposed method are evaluated by means of coding gain and bit rates (bits per pixels) in the point of view of color image compression.

The test images are comprised of ISO/IEC SCID standard images [13] and Kodak set of 24 color images [14]. *scid3h* is a synthesized image by concatenating a half-sized *bike, cafe,* and *woman*. *kodaksetH* is made of the 24 images subsampled into a half. Synthetic images are prepared to test the effectiveness of obtained RCTs, even if possible image processing has been applied to an original image to change the statistical characteristics.

4.1 Coding Gain

The coding gain in transform coding is defined by a ratio of the average mean square distortion generated by the plain waveform coding against that generated by a transform coding at the identical bit rate [9], [15], [16]. It is calculated, if the transform and the covariance matrix of a target signal are given. It gives a theoretical figure of merit of a transform in the sense of decorrelation capability. Needless to mention, since KLT is optimum in the sense of decorrelation among components, the highest coding gain is obtained by rKLT as found in Table 2. However a higher coding gain does not always lead to an efficient lossless compression, which will be demonstrated in the next subsection.

Among the others, DCT performs well for most of images. The result agrees with the fact that DCT is a good approximation of KLT [9].

EMN outperforms the others on average because of the following two major reasons. At first, as seen in Eq. (6), the luma component expressed by E is formed by a plain average of three color components as same as in 3-point DCT. Secondly, the color components are well decorrelated in two chrominance representations; they are formed by a difference between a middle-wavelength light signal of green and an adjacent band signal of red or blue.

It is a slight surprise to see that YCoCg is inferior to EMN and LMN in spite of the latest standard. While YCbCr is the most popular in image/video compression for a long time, the coding gain of its RCT is lower than EMN, LMN, DCT, and YCoCg. RCTs derived from classical YUV and YIQ are poorer than DCT.

4.2 Bit Rates

Actual bit rates depend on both RCTs and compression algorithms. Hence two different types of codecs for lossless image compression are experimented for reliable evaluations. JPEG-LS [1] is selected as a predictive coding in the picture domain. The other is scalable coding in the wavelet-transform domain. LBLC [17] is one of them and is competitive with JPEG2000 in lossy-to-lossless compression.

Table 3 shows a comparison in bit rate saving that is defined by the difference between two bit rates obtained by RCT-free compression and compression after RCT. Values in the column of RGB show the actual bit rates of compressed images obtained by RCT-free compression. The bit rate saving gained by those RCTs is about 0.6 and 0.7 bpp/component in picture-domain coding and transform-domain coding, respectively.

EMN is slightly superior to both LMN and YCoCg. It outperforms the others on average because of its high

Table 2 Coding gain of several RCTs derived from irreversible component transforms. YCbCr stands for Y_{709}CbCr. LMN is identical to JPEG2000 RCT. Values in boldface are the highest except for rKLT.

Image	Coding gain in dB								
	rKLT	DCT	YCbCr	YUV	YIQ	GMN	EMN	LMN	YCoCg
bike	3.473	**1.923**	1.726	1.785	1.604	1.119	1.669	1.665	1.854
cafe	3.435	**1.976**	1.766	1.481	1.826	0.929	1.405	1.405	1.910
woman	6.465	3.319	3.975	3.461	3.063	4.390	**4.883**	4.874	3.246
scid3h	3.686	**2.216**	1.654	1.486	2.000	1.535	2.082	2.066	2.134
kodaksetH	4.644	**2.955**	2.494	2.378	2.621	2.249	2.943	2.883	2.830
Average	4.340	2.477	2.323	2.118	2.222	2.044	**2.596**	2.578	2.394

Table 3 Bit rate saving against the RCT-free compression. JLS [1] is a sample codec operating in the picture-domain. LBLC [17] is a sample codec operating in the transform-domain. The column of RGB lists the bit rates obtained by the RCT-free compression. Values in boldface show the best among all.

Image	Codec	RGB	Bit rate saving in bpp (bits per pixel) per component							
			rKLT	DCT	YCbCr	YIQ	GMN	EMN	LMN	YCoCg
bike	JLS	4.885	0.399	0.404	0.416	0.440	**0.448**	0.436	0.433	0.403
	LBLC	4.956	0.562	0.593	0.591	**0.621**	0.581	0.607	0.602	0.592
cafe	JLS	5.457	0.194	0.227	0.201	0.210	0.205	**0.238**	0.224	0.212
	LBLC	5.665	0.421	**0.461**	0.409	0.436	0.386	0.457	0.438	0.442
woman	JLS	5.103	0.557	0.616	0.602	0.616	0.594	**0.637**	0.630	0.610
	LBLC	4.951	0.602	0.664	0.638	0.653	0.626	**0.676**	0.668	0.656
scid3h	JLS	5.241	0.607	0.620	0.583	0.617	0.599	**0.626**	0.620	0.616
	LBLC	5.194	0.692	0.703	0.667	0.703	0.679	**0.717**	0.710	0.697
kodaksetH	JLS	5.179	1.025	1.061	1.110	1.058	**1.118**	1.112	1.111	1.060
	LBLC	5.162	1.072	1.097	1.120	1.093	1.124	**1.132**	1.131	1.096
Average	JLS	5.173	0.556	0.586	0.582	0.588	0.593	**0.610**	0.604	0.580
	LBLC	5.186	0.670	0.703	0.685	0.702	0.679	**0.718**	0.710	0.696

coding gain and the following two reasons. At first, only a single round-off operation is involved with the reversible EMN and this fact results in a lower level of round-off errors. Secondly, the round-off error is generated at the destination node and is neither amplified nor diffused by traveling through any other lifting steps [18] unlike the case of DCT and YCoCg.

rKLT is poor in lossless compression in spite of its highest coding gain, because there are four round-off error sources as seen in Table 1, and the round-off errors propagate through lifting steps.

GMN in predictive coding performs better for a few images in lossless compression, because it generates no round-off error as seen in the expressions in Table 1 and the prediction of pixel values is sensitive to the statistical distribution of less significant bits of pixel values.

Since three rounding operations are involved with Y_{709}CbCr, YUV, and YIQ, and since the round-off errors are amplified through a few rear lifting steps, their performances are poorer in lossless compression.

In consideration of two evaluations by coding gain and bit rates, it is evident that EMN is a good choice for both lossless compression and lossy-to-lossless compression of color images.

5. Conclusions

An arbitrary linear transformation has been implemented as a reversible transform. Reversible implementations of some irreversible component transforms were obtained and compared in coding gain and bit rates. Among some experimented RCTs, EMN was found to outperform the others in color image compression.

Acknowledgments

The authors thank anonymous reviewers for their comments and suggestions to improve the readability and to add an illustration of the reversibility in lifting.

References

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

[2] D.S. Taubman and M.W. Marcellin, JPEG2000, Kluwer Academic, Boston, 2002.

[3] M. Barni, ed., Document and Image Compression, CRC Press, Boca Raton, 2006.

[4] M. Yamaguchi, H. Haneishi, and N. Ohyama, "Beyond red-green-blue (RGB): Spectrum-based color imaging technology," J. Imaging Sci. and Tech., vol.52, no.1, pp.010201-1–15, Jan./Feb. 2008.

[5] P. Hao and Q. Shi, "Matrix factorizations for reversible integer mapping," IEEE Trans. Signal Process., vol.49, no.10, pp.2314–2324, Oct. 2001.

[6] S.C. Pei and J.J. Ding, "Reversible integer color transform," IEEE Trans. Image Process., vol.16, no.6, pp.1686–1691, June 2007.

[7] W. Sweldens, "The lifting scheme: A custom-design construction of biorthogonal wavelets," J. Appl. Comput. Harmonic Analysis, vol.3, pp.186–200, 1996.

[8] G. Strang, Linear Algebra and Its Applications, 2nd ed., Harcourt Brace Jovanovich Publishers, Orlando, 1980.
[9] N.S. Jayant and P. Noll, Digital Coding of Waveforms, Prentice-Hall, New Jersey, 1984.
[10] C. Poynton, Digital Video and HDTV, Morgan Kaufmann, San Francisco, 2003.
[11] N. Liu, Fractal Imaging, Academic Press, New York, 1997.
[12] H. Malvar and G. Sullivan, "YCoCg-R: A color space with RGB reversibility and low dynamic range," ISO/IEC MPEG & ITU-T VCEG, JVT-I014r3, Trondheim, July 2003.
[13] Graphic Technology -Press Digital Data Exchange- Standard Colour Image Data (SCID), Japanese Standard Association, Tokyo, 1995.
[14] formerly available at ftp://ftp.ipl.rpi.edu/stills/kodak/color
[15] J. Woods, Subband Coding, Kluwer Academic Press, Boston, 1990.
[16] J. Katto, "Performance evaluation of subband coding and optimization of its filter coefficients," Proc. SPIE (VCIP 1991), vol.1605, pp.95–106, 1991.
[17] K. Shinoda, H. Kikuchi, and S. Muramatsu, "A lossless-by-lossy coding for scalable lossless image compression," IEICE Trans. Fundamentals, vol.E91-A, no.11, pp.3356–3364, Nov. 2008.
[18] H. Kikuchi, J. Hwang, S. Muramatsu, and J. Shin, "Reversible component transforms by the LU factorization," 28th Picture Coding Symp., pp.238–241, Nagoya, Dec. 2010.

Appendix: Reversibility in Lifting

The following trivial identity suggests an algebraic aspect in a lifting computation why it is reversible.

$$\begin{pmatrix}1 & 0\\ -c & 1\end{pmatrix}\begin{pmatrix}1 & 0\\ c & 1\end{pmatrix} = \begin{pmatrix}1 & 0\\ 0 & 1\end{pmatrix} \tag{A$\cdot$1}$$

The inversion of a unit triangular matrix is identical to itself, if the sign of the off-diagonal entry is flipped. Hence an LDU factorization gives a conceptual lifting configuration, if all diagonal entries are unity in magnitude.

Lifting yet offers another advantage; any sort of mapping is allowed on a lifting step. In a lifting step described by the equation

$$\begin{pmatrix}y_1\\ y_2\end{pmatrix} = \begin{pmatrix}1 & 0\\ c & 1\end{pmatrix}\begin{pmatrix}x_1\\ x_2\end{pmatrix}, \tag{A$\cdot$2}$$

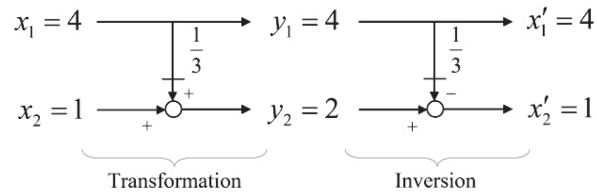

Fig. A· 1 An illustration of the reversibility in a lifting system.

cx_1 is added to x_2 to compute y_2. It is worth to note that cx_1 can be generalized to any function with respect to x_1. Even if it may be a nonlinear operation such as truncation and look-up table mapping, the reversibility is ensured.

A lifting system consists of a pair of processes: transformation and inversion. They share the same algebraic operations except for the polarity inversion on a pair of lifting steps each of which exists at the corresponding location in transformation or inversion as suggested in the second and first matrix factor in Eq. (A· 1), respectively. Imagine the simplest case shown Fig. A· 1: a two-dimensional one-step lifting system such as Eq. (A· 1), where $c = 1/3$. Assume the input vector, $(x_1, x_2) = (4, 1)$, is applied to a lifting step which is given by

$$y_2 \leftarrow x_2 + \lfloor x_1/3 \rfloor = 1 + \lfloor 4/3 \rfloor. \tag{A$\cdot$3}$$

Note that $y_1 \leftarrow x_1$ in the lifting system. One obtains $y_1 = 4$ and $y_2 = 2$. In the inversion system, $x'_1 \leftarrow y_1$ and the lifting step is written as

$$x'_2 \leftarrow y_2 - \lfloor y_1/3 \rfloor = 2 - \lfloor 4/3 \rfloor. \tag{A$\cdot$4}$$

The original signal is hence recovered as $(x'_1, x'_2) = (4, 1)$. Since a lifting step in inversion reproduces the same quantity that has been created on the corresponding lifting step in transformation, and since the quantity is added in transformation and is subtracted in inversion, the original quantity is recovered. In this way, integer-valued inputs are exactly recovered in a lifting system, no matter what nonlinear operations may be involved with lifting steps.