

Reversible Resampling of Integer Signals

Pengwei Hao, *Member, IEEE*

Abstract—Except some extremely special cases, signal resampling was generally considered to be irreversible because of strong attenuation of high frequencies after interpolation. In this paper, we prove that signal resampling based on polynomial interpolation can be reversible even for integer signals, i.e., the original signal can be reconstructed losslessly from the resampled data. By using matrix factorization, we also propose a reversible method for uniform shifted resampling and uniform scaled and shifted resampling. The new factorization yields three elementary integer-reversible matrices. The method is actually a new way to compute linear transforms and a lossless integer implementation of linear transforms with the factor matrices. It can be applied to integer signals by in-place integer-reversible computation, which needs no auxiliary memory to keep the original sample data for the transformation during the process or for “undo” recovery after the process. Some examples of low-order resampling solutions are also presented in this paper and our experiments show that the resampling error relative to the original signal is comparable to that of the traditional irreversible resampling.

Index Terms—Factorial polynomials, integer-to-integer transforms, PLUS factorization of matrices, resampling, Stirling numbers.

I. INTRODUCTION

RESAMPLING is a calculation of a discrete set of points of a signal using another discrete set of known points of the signal. Signal resampling is generally implemented by two processes: interpolation of the given discrete signal to a continuous signal and then sampling the interpolated signal at the new coordinate points.

Signal resampling is used for a number of different purposes in signal processing, especially in image processing, see [4], [5], [6], [9], and [11]. Signal format conversion, sample rate conversion, sample filling/decimation, alignment, rectification, and warping are all basic operations that involve resampling. Resampling is employed in image processing and computer graphics for better display, finer registration, less-distortion rotation, and more realistic texture mapping.

By reversible resampling, we mean that the original discrete samples of a signal can be losslessly recovered from the resampled discrete samples of the signal by re-resampling. Reversible

resampling can be applied to integer signals by in-place integer-reversible computation, which needs no auxiliary memory to keep the original sample data for the transformation during the process or for “undo” recovery after the process.

For signal interpolation, a signal can be exactly reconstructed from its samples if the signal is bandlimited and the sampling rate is above the Nyquist frequency. Unfortunately, the conditions cannot always be satisfied for digitized signals and are generally not true for digital images. For signal resampling, it was generally considered that the original samples could not be losslessly recovered from the resampled data except in some extremely special cases, because strong attenuation of high frequencies occurs after interpolation [9, p. 412], [10].

The main contribution of this paper is twofold. First, we prove that signal resampling based on polynomial interpolation can be reversible even for integer signals, i.e., the original signal can be reconstructed losslessly from the resampled data. Second, by using matrix factorization, we propose an elegant method to realize the reversible resampling. The method is actually a lossless integer-reversible implementation of linear transforms.

For integer-reversible transformation, Daubechies and Sweldens [12] firstly tried to factor polyphase matrices of wavelets into lifting steps, which can be directly used to realize integer transformation. Discrete cosine and Fourier transforms were also computed for integer-reversible transformation with the lifting scheme by Chen [13], Liang and Tran [14] and Orintara *et al.* [15]. Overlapping rounding transform was developed by Jung and Prost [16] as a reversible transform with a pair of rounding operations, but later Adams and Kossentini [17] proved that it is equivalent to the lifting scheme. For generic linear transforms, Hao and Shi [3] proposed a matrix factorization of the *elementary reversible matrices* with *integer units* on the diagonal to realize lossless integer-to-integer transformation. They proved that any n -dimensional nonsingular transform matrix could be factored into a product of not more than $n + 1$ single-row elementary reversible matrices (SERM) or three triangular elementary reversible matrices (TERM). The factorization has also been shown to be one case in the PLUS factorization family [2]. Differently, Plonka [18] chose scaling factors to expand the ranges of linear transforms so that the transform range redundancy can be traded for integer reversibility. Srinivasan [19] recently proposed modulo transforms to employ Pythagorean triples to produce reversible, normalized, scale-free matrices and then to cascade them to approximate orthonormal linear transforms. This paper uses a matrix factorization method similar to the PLUS factorization to reconcile resampling to reversibility.

The paper is organized as follows. The resampling is formulated as a linear transform in Section II, a reversible shifted resampling method is presented in Section III, and a scaled and shifted resampling method is in Section IV. Section V gives

Manuscript received September 18, 2007; revised September 12, 2008. First published October 31, 2008; current version published January 30, 2009. This work was supported in part by the NSFC of China by Grant 60572043 and by the NKBRPC of China by Grant 2004CB318005. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yonina C. Eldar.

The author is with the Department of Computer Science, Queen Mary, University of London, E1 4NS, U.K., and also with the Center for Information Science, Peking University, Beijing, 100871, China (e-mail: phao@cis.pku.edu.cn; phao@dcs.qmul.ac.uk).

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

Digital Object Identifier 10.1109/TSP.2008.2008243

analysis of the integer resampling error, and some examples and experimental results are given in Section VI. Finally, this paper is concluded in Section VII.

II. RESAMPLING MODELING

Suppose the given discrete samples of a signal at n coordinate points $t_1, t_2, t_3, \dots, t_n$ are $p_1, p_2, p_3, \dots, p_n$, a resampling method is to find new samples at m coordinate points $s_1, s_2, s_3, \dots, s_m$: $p(s_1), p(s_2), p(s_3), \dots, p(s_m)$. We can use vectors to denote the original samples: $\mathbf{p}_n = [p_1, p_2, p_3, \dots, p_n]^T$. Without loss of generality, we assume that the original sampling is uniform, i.e., $t_k = k$ and $p_k = p(k)$ for $k = 1, 2, 3, \dots, n$. We will use such symbols throughout this paper. The signals in this paper are not necessary to be scalar, and actually our method generalizes its validity to multiple component signals or vector signals.

If a polynomial interpolation is employed for resampling, we can use the following function as the interpolated continuous signal from the n samples:

$$p(t) = [1, t, t^2, \dots, t^{n-1}] \cdot \mathbf{A} \cdot \mathbf{p}_n. \quad (1)$$

For signal interpolation, the original samples should be reproducible from the interpolated signal. Hence, the following equations should be satisfied:

$$p(k) = p_k \text{ for } k = 1, 2, 3, \dots, n. \quad (2)$$

Let \mathbf{V} denote the following m -by- n Vandermonde matrix:

$$\mathbf{V}(s_1, s_2, s_3, \dots, s_m) = \begin{bmatrix} s_1^0 & s_1^1 & s_1^2 & \dots & s_1^{n-1} \\ s_2^0 & s_2^1 & s_2^2 & \dots & s_2^{n-1} \\ s_3^0 & s_3^1 & s_3^2 & \dots & s_3^{n-1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ s_m^0 & s_m^1 & s_m^2 & \dots & s_m^{n-1} \end{bmatrix} \quad (3)$$

where the (i, j) th element can be written as

$$\mathbf{V}_{i,j}(s_1, s_2, s_3, \dots, s_m) = s_i^{j-1} \text{ for } i, j = 1, 2, 3, \dots, n. \quad (4)$$

We also use these simpler matrix notations below, $\mathbf{V}(s) = \mathbf{V}(1+s, 2+s, \dots, m+s)$ and $\bar{\mathbf{V}} = \mathbf{V}(0)$ for $m = n$. We may use subscripts for dimensions of matrices, e.g., $\mathbf{V}_{m \times n}(s)$.

With (1)–(2) and $\bar{\mathbf{V}}$, we can build a matrix equation and then we have $\bar{\mathbf{V}} \cdot \mathbf{A} = \mathbf{I}$ or $\mathbf{A} = \bar{\mathbf{V}}^{-1}$.

If we use the interpolated signal of (1) for resampling, the new samples are

$$p(s_k) = [1, s_k, s_k^2, \dots, s_k^{n-1}] \cdot \bar{\mathbf{V}}^{-1} \cdot \mathbf{p}_n \quad (5)$$

where $k = 1, 2, 3, \dots, m$.

Putting all the samples of (5) together, we can find a matrix equation between the original samples and the new samples

$$\begin{bmatrix} p(s_1) \\ p(s_2) \\ p(s_3) \\ \vdots \\ p(s_m) \end{bmatrix} = \mathbf{V}(s_1, s_2, s_3, \dots, s_m) \cdot \bar{\mathbf{V}}^{-1} \cdot \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ \vdots \\ p_n \end{bmatrix}. \quad (6)$$

Let $\mathbf{R}(s_1, s_2, s_3, \dots, s_m)$ be the $m \times n$ transform matrix for m -point resampling from n -point discrete signal, we have

$$\mathbf{R}(s_1, s_2, s_3, \dots, s_m) = \mathbf{V}(s_1, s_2, s_3, \dots, s_m) \cdot \bar{\mathbf{V}}^{-1}. \quad (7)$$

Therefore, the resampling problem based on the above assumptions is actually a linear transform, and the existence of the reversible resampling solution is whether the linear transform is invertible.

Generally, if $m < n$, the equation system is under-determined, and we have to append $n - m$ more rows to the matrix to make it invertible, which implies that the resampled data is not sufficient for the recovery of the original samples. The method to remedy this is to keep enough original samples for reversibility or to increase the resampling rate.

For $m = n$, the resampling matrix \mathbf{R} is a square matrix. It is invertible or $\det(\mathbf{R}) \neq 0$ if all the resampling points are different from each other, and it is integer-reversible if $|\det(\mathbf{R})| = 1$ [3]. A reversible shifted resampling method for this case is presented in the next section.

If $m > n$, the resampling equation is overdetermined. Because the resampling matrix is a product of a Vandermonde matrix and the inverse of another Vandermonde matrix, if all the resampling points are different from each other, we can just use n samples to recover the original. However, the problem for integer signals is its integer reversibility, which needs $|\det(\mathbf{R})| \geq 1$ for the reduced resampling matrix \mathbf{R} . In this case, the resampling method is a bit complicated. In this paper, the scaled and shifted resampling is investigated, and we use $n + 1$ or more resampled points to recover the n original integer samples.

III. REVERSIBLE SHIFTED RESAMPLING

Shifted resampling has a number of applications, such as registration of images with only subpixel shifts [20], [21], rotation of images by shears [10]. If a shifted resampling can be reversible, it needs no auxiliary memory to keep the original image for the transformation during the process or for “undo” recovery after the process.

A simple shifted resampling model is to shift a digitized signal by a fractional delay or a digital image by a subpixel offset and then to resample the signal at the same rate as the original, i.e., $m = n, s_k = k + s$, where s is a real number for the shift, $-1/2 < s \leq 1/2$. Such a shifted-resampling problem can be formulated as to find $p(k + s)$ from $p(k) = p_k$ for $k = 1, 2, 3, \dots, n$, and the reversibility can be achieved by recovering all the original points $p(k) = p_k$ from $p(k + s)$ by re-resampling with a shift of $-s$.

Let $\binom{m}{n}$ denote the binomial coefficient “ m choose n ” and $\mathbf{T}(s)$ denote an upper triangular matrix as a function of s as follows:

$$\mathbf{T}(s) = \begin{bmatrix} \binom{0}{0} & \binom{1}{0}s & \binom{2}{0}s^2 & \dots & \binom{n-1}{0}s^{n-1} \\ 0 & \binom{1}{1} & \binom{2}{1}s & \dots & \binom{n-1}{1}s^{n-2} \\ 0 & 0 & \binom{2}{2} & \dots & \binom{n-1}{2}s^{n-3} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \binom{n-1}{n-1} \end{bmatrix} \quad (8)$$

where the (i, j) th element can be written as

$$T_{i,j}(s) = \begin{cases} 0 & (i > j) \\ \binom{j-1}{i-1} s^{j-i} & (i \leq j) \end{cases} \quad \text{for } i, j = 1, 2, 3, \dots, n. \quad (9)$$

$T(s)$ is actually unit upper triangular, $T(0) = I$ and $\det(T(s)) = 1$. With $T(s)$ as in (9), we have

$$[1, t + s, (t + s)^2, \dots, (t + s)^{n-1}] \\ = [1, t, t^2, \dots, t^{n-1}] \cdot T(s). \quad (10)$$

For matrices in (3) and (8), it is easy to see that the following equation exists:

$$V(s) = \bar{V} \cdot T(s). \quad (11)$$

Based on the above (5)–(7) and (11), we can formulate the shifted-resampling problem as

$$p(s) = V(s) \cdot \bar{V}^{-1} \cdot p_n = R(s) \cdot p_n \quad (12)$$

where $p(s) = [p(1+s), p(2+s), p(3+s), \dots, p(n+s)]^T$, $R(s) = V(s) \cdot \bar{V}^{-1}$.

It is equivalent to interpolation with piecewise basis functions

$$p(s) = \sum_{j=1}^n R_j(s) \cdot p_j \quad \text{or} \quad p(i+s) = \sum_{j=1}^n R_{i,j}(s) \cdot p_j \quad (13)$$

where $R_j(s)$ is the j th column of $R(s)$ representing the j th piecewise basis function for the interpolation, and $R_{i,j}(s)$ is the i th row and j th column element of $R(s)$ representing the i th piece of j th piecewise basis function.

The above (12)–(13) tell that an n -point-to- n -point shift-resampling by using polynomial interpolation can be implemented by a linear transform, and the resampling transform matrix $R(s)$ is a function of shift s . Then we have

$$R(s) = V(s) \cdot \bar{V}^{-1} = \bar{V} \cdot T(s) \cdot \bar{V}^{-1} \quad (14)$$

and the determinant of the transform matrix is

$$\det(R(s)) = \det(\bar{V}) \cdot \det(T(s)) \cdot \det(\bar{V})^{-1} = 1. \quad (15)$$

In [3], we have proved that a matrix can be factorized into three TERMS if the determinant of the matrix is absolutely 1. A TERM can directly transform an integer vector into another integer vector, and the transform can be losslessly reversed.

Although a few customizable factorization forms for reversible transformation with $R(s)$ can be found by using the methods presented in [2] and [3], we have also derived another elegant and simple factorization of three TERMS as follows.

Suppose the LDU factorization of the Vandermonde matrix \bar{V} is $\bar{V} = L \cdot D \cdot \bar{U}$, and use $U(s)$ to denote the combination of three upper triangular matrices and two diagonal matrices

$$U(s) = D \cdot \bar{U} \cdot T(s) \cdot \bar{U}^{-1} \cdot D^{-1}. \quad (16)$$

Then, we have a new factorization of three TERMS for (14)

$$R(s) = L \cdot D \cdot \bar{U} \cdot T(s) \cdot \bar{U}^{-1} \cdot D^{-1} \cdot L^{-1} \\ = L \cdot U(s) \cdot L^{-1} \quad (17)$$

where L is a unit lower triangular matrix and a lower TERM (thus L^{-1} is also a lower TERM) and $U(s)$ is actually a unit upper triangular matrix and an upper TERM. To detail and clarify the representation, we present the matrices by elements

$$L_{i,j} = \begin{cases} 0 & (i < j) \\ \binom{i-1}{j-1} & (i \geq j) \end{cases} \quad (18)$$

$$D_{i,j} = (i-1)! \delta_{i,j} \quad (19)$$

$$U_{i,j}(s) = \begin{cases} 0 & (i > j) \\ \frac{s^{j-i}}{(j-i)!} & (i \leq j) \end{cases} \quad (20)$$

where $x^{\underline{m}} = x(x-1)(x-2)\cdots(x-m+1)$ are the falling factorial polynomials, $x^{\underline{0}} = 1$, and $0! = 1$. For $-1 < s \leq 1$, we have $|U_{i,j}(s)| < 1$ if $i < j$.

Amazingly, L is the elegant and famous *Yanghui's Triangle* (called Pascal's Triangle in the West and the staircase of Mount Meru in India) [22], [23], [24]. Its inverse is also simple

$$L_{i,j}^{-1} = \begin{cases} 0 & (i < j) \\ (-1)^{i+j} \binom{i-1}{j-1} & (i \geq j). \end{cases} \quad (21)$$

It is also worth mentioning that the upper triangular matrix \bar{U} is made of the Stirling numbers of the second kind, " m subset n " $\{^m_n\}$, and its inverse \bar{U}^{-1} is of the Stirling numbers of the first kind, " m cycle n " $[^m_n]$. The (i, j) th elements are

$$\bar{U}_{i,j} = \begin{cases} 0 & (i > j) \\ \{^j_i\} & (i \leq j) \end{cases} \quad (22)$$

$$\bar{U}_{i,j}^{-1} = \begin{cases} 0 & (i > j) \\ (-1)^{i+j} [^j_i] & (i \leq j). \end{cases} \quad (23)$$

The above LDU factorization of \bar{V} can be proved using the following identity [7]:

$$\sum_{k=1}^{\min(i,j)} k! \binom{i}{k} \{^j_k\} = i^j. \quad (24)$$

The inverse of L (18), (21) and \bar{U} (22), (23) can be proved using the following identities [1], respectively:

$$\sum_{k=j}^i (-1)^{k+j} \binom{i}{k} \binom{k}{j} = \sum_{k=j}^i (-1)^{i+k} \binom{i}{k} \binom{k}{j} = \delta_{i,j} \quad (25)$$

$$\sum_{k=i}^j (-1)^{k+j} [^j_k] \{^k_i\} = \sum_{k=i}^j (-1)^{i+k} \{^j_k\} [^k_i] = \delta_{i,j}. \quad (26)$$

The following identities [1] can be used for proving $U(s)$ (20)

$$x^{\underline{m}} = \sum_{k=1}^m (-1)^{m+k} \binom{m}{k} x^k \quad (27)$$

$$x^m = \sum_{k=1}^m \{^m_k\} x^k. \quad (28)$$

Similar to the falling factorial polynomials, the rising factorials are defined as $x^{\overline{m}} = x(x+1)(x+2)\cdots(x+m-1)$, which are related to the falling factorials by a difference in

sign: $(-x)^m = (-1)^m (x)^m$ [1]. Then, with the following identity similar to (25):

$$\sum_{k=i}^j \frac{s^{k-i}}{(k-i)!} \frac{s^{j-k}}{(j-k)!} = \sum_{k=i}^j \frac{s^{k-i}}{(k-i)!} \frac{s^{j-k}}{(j-k)!} = \delta_{i,j} \quad (29)$$

we have

$$U^{-1}(s) = U(-s). \quad (30)$$

Hence,

$$\begin{aligned} R^{-1}(s) &= L \cdot U^{-1}(s) \cdot L^{-1} \\ &= L \cdot U(-s) \cdot L^{-1} = R(-s) \end{aligned} \quad (31)$$

which implies that the resampling with a shift and then re-resampling with a backward shift restores the original input losslessly. It is the reason that we name it *reversible resampling*.

For integer reversible mapping, the transform with L^{-1} can be implemented by the reverse transform of L , so that the combined transform with $R(s)$ and $R(-s)$ can give the result exactly the same as the original if the integer-reversible implementation is employed [3]. Therefore, the proposed reversible resampling also works for integer signals.

IV. REVERSIBLE SCALED AND SHIFTED RESAMPLING

In sample rate conversion, image registration, texture mapping and some other applications, we often need uniform scaled and shifted resampling at points $s_k = k \cdot n/m + s$, where $m > n$. In such a case, the resampling is at a higher sampling rate than the input signal, and the resampling matrix is of $m \times n$:

$$\begin{aligned} R_{m \times n}(s) &= R_{m \times n} \left(\frac{n}{m} + s, \frac{2n}{m} + s, \frac{3n}{m} + s, \dots, \frac{mn}{m} + s \right) \\ &= V_{m \times n} \left(\frac{n}{m} + s, \frac{2n}{m} + s, \frac{3n}{m} + s, \dots, \frac{mn}{m} + s \right) \\ &\quad \cdot \bar{V}_{n \times n}^{-1}. \end{aligned} \quad (32)$$

With (10), we have

$$\begin{aligned} V_{m \times n} \left(\frac{n}{m} + s, \frac{2n}{m} + s, \frac{3n}{m} + s, \dots, \frac{mn}{m} + s \right) \\ = V_{m \times n} \left(\frac{n}{m}, \frac{2n}{m}, \frac{3n}{m}, \dots, \frac{mn}{m} \right) \cdot T_{n \times n}(s). \end{aligned} \quad (33)$$

With the LDU factorization of $\bar{V}_{n \times n} = L_{n \times n} \cdot D_{n \times n} \cdot \bar{U}_{n \times n}$, by denoting $M_{m \times n} = V_{m \times n} \left(\frac{n}{m}, \frac{2n}{m}, \frac{3n}{m}, \dots, \frac{mn}{m} \right) \cdot D_{n \times n}^{-1}$ and $Q'_{n \times n}(s) = D_{n \times n} \cdot T_{n \times n}(s) \cdot \bar{U}_{n \times n}^{-1} \cdot D_{n \times n}^{-1}$, we can have a factorization of

$$R_{m \times n}(s) = M_{m \times n} \cdot Q'_{n \times n}(s) \cdot L_{n \times n}^{-1} \quad (34)$$

where $M_{m \times n}$ is a constant matrix of full column rank, $Q'_{n \times n}(s)$ is a unit upper triangular matrix with s as its independent variable, and $L_{n \times n}$ is a unit lower triangular matrix as defined in (18).

Since $M_{m \times n}$ is a product of an m -by- n Vandermonde matrix and an n -by- n diagonal matrix, all leading principal submatrices

of a matrix composed of any n rows of $M_{m \times n}$ are nonsingular. Owing to $m > n$, similar to the proof to Theorem 1 in [2], it can be proved that the following factorization exists:

$$\begin{aligned} M_{m \times n} &= S'_{m \times m} \cdot L'_{m \times m} \cdot U'_{m \times n} \\ &= \begin{bmatrix} I_n & S''_{n \times (m-n)} \\ \mathbf{0} & I_{m-n} \end{bmatrix} \cdot \begin{bmatrix} L''_{n \times n} & \mathbf{0} \\ \mathbf{0} & I_{m-n} \end{bmatrix} \\ &\quad \cdot \begin{bmatrix} U''_{n \times n} \\ B_{(m-n) \times n} \end{bmatrix} \end{aligned} \quad (35)$$

where $S'_{m \times m}$ is a unit upper triangular matrix with the off-diagonal elements only in the $(n-1)$ th to m th columns possibly being nonzero ($S''_{n \times (m-n)}$), $L'_{m \times m}$ is a unit lower triangular matrix with nonzero off-diagonal elements only in the n -by- n top-left corner submatrix ($L''_{n \times n}$), and $U'_{m \times n}$ is an m -by- n matrix with an n -by- n unit upper triangular submatrix at the top ($U''_{n \times n}$). The integer reversibility can still be achieved for $U''_{n \times n}$ is unit triangular. On the other hand, the product of $S'_{m \times m}$ and $L'_{m \times m}$ can be rewritten as one matrix and can be converted into a unit lower triangular matrix (also a TERM) by symmetric permutations

$$\begin{aligned} &\begin{bmatrix} I_n & S''_{n \times (m-n)} \\ \mathbf{0} & I_{m-n} \end{bmatrix} \cdot \begin{bmatrix} L''_{n \times n} & \mathbf{0} \\ \mathbf{0} & I_{m-n} \end{bmatrix} \\ &= \begin{bmatrix} L''_{n \times n} & S''_{n \times (m-n)} \\ \mathbf{0} & I_{m-n} \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{0} & I_{m-n} \\ I_n & \mathbf{0} \end{bmatrix} \cdot \begin{bmatrix} I_{m-n} & \mathbf{0} \\ S''_{n \times (m-n)} & L''_{n \times n} \end{bmatrix} \\ &\quad \cdot \begin{bmatrix} \mathbf{0} & I_{m-n} \\ I_n & \mathbf{0} \end{bmatrix}. \end{aligned} \quad (36)$$

The new combined matrix is also an integer-reversible matrix. Thus the factorization (35) can be further simplified into two integer-reversible matrices

$$\begin{aligned} M_{m \times n} &= \begin{bmatrix} L''_{n \times n} & S''_{n \times (m-n)} \\ \mathbf{0} & I_{m-n} \end{bmatrix} \cdot \begin{bmatrix} U''_{n \times n} \\ B_{(m-n) \times n} \end{bmatrix} \\ &= S_{m \times m} \cdot U'_{m \times n}. \end{aligned} \quad (37)$$

With (34) and (37) and multiplying $U'_{m \times n}$ and $Q'_{m \times n}(s)$ into one integer-reversible matrix $Q_{m \times n}(s)$ ($Q_{m \times n}(s) = U'_{m \times n} \cdot Q'_{m \times n}(s)$, with a unit upper triangular matrix at the top), we have a new integer reversible matrix factorization for scaled and shifted resampling:

$$R_{m \times n}(s) = S_{m \times m} \cdot Q_{m \times n}(s) \cdot L_{n \times n}^{-1}. \quad (38)$$

The reverse can be done by using m samples for the inverse of $S_{m \times m}$ and then using n S -reversed samples for the inverse of $Q_{m \times n}(s)$ and $L_{n \times n}^{-1}$.

Equation (38) tells that uniform scaled and shifted resampling can also be integer-reversible.

Note that the factorization (38) is not unique if $m > n + 1$.

V. RESAMPLING ERROR ANALYSIS

By resampling error, we mean the difference between the original and the resampled signals. It gives the information about how much the resampled signal drifts away from the

original input. In this section, we analyze the error for shifted resampling as given in Section III. The error analysis for scaled and shifted resampling is similar but a bit complicated.

Compared to the traditional resampling scheme by piece-wise continuous interpolation and re-sampling (except the nearest neighbor scheme) with which the original signal cannot be perfectly recovered from the resampled data, the proposed resampling method is reversible or losslessly recoverable, even for integer inputs. If the order-0 resampling or the nearest neighbor interpolation resampling scheme is employed, the results of the traditional method are the same as those of the proposed method. However, for higher order resampling, the resampled outputs are different, and the absolute error between them can be measured by the error between the outputs and the ground-truth data. This will be demonstrated with resampling of a window function in the next section. In case that the continuous ground-truth signal is unknown, the error can be approximated between the output with the proposed method and that with the traditional interpolation methods, such as the nearest neighbor method, or between the resampled data and the input. We can find the approximated error of the shifted resampling as follows:

$$\begin{aligned} e_A &= \mathbf{R}(s) \cdot \mathbf{p}_n - \mathbf{p}_n = [\mathbf{R}(s) - \mathbf{I}] \cdot \mathbf{p}_n \\ &= \mathbf{L} \cdot [\mathbf{U}(s) - \mathbf{I}] \cdot \mathbf{L}^{-1} \cdot \mathbf{p}_n. \end{aligned} \quad (39)$$

With this (39), the absolute error can be easily found, and the norm upper bound can be derived as follows.

$$\|\mathbf{e}_A\| \leq \|\mathbf{L}\| \cdot \|\mathbf{U}(s) - \mathbf{I}\| \cdot \|\mathbf{L}^{-1}\| \cdot \|\mathbf{p}_n\|. \quad (40)$$

Thus, the upper bound of the relative error is

$$\|\mathbf{e}_R\| \leq \|\mathbf{L}\| \cdot \|\mathbf{U}(s) - \mathbf{I}\| \cdot \|\mathbf{L}^{-1}\|. \quad (41)$$

For the unit lower triangular matrices \mathbf{L} (18) and \mathbf{L}^{-1} (21), the 1-norm and the infinity-norm are the same

$$\begin{aligned} \|\mathbf{L}\|_1 &= \|\mathbf{L}^{-1}\|_1 = \max_j \sum_{i=j}^n \binom{i-1}{j-1} \\ &= \max_j \binom{n}{j} = \binom{n}{\lfloor \frac{n+1}{2} \rfloor} \end{aligned} \quad (42)$$

$$\begin{aligned} \|\mathbf{L}\|_\infty &= \|\mathbf{L}^{-1}\|_\infty = \max_i \sum_{j=1}^i \binom{i-1}{j-1} \\ &= \max_i 2^{i-1} = 2^{n-1}. \end{aligned} \quad (43)$$

The norms of the upper triangular matrix $\mathbf{U}(s)$ (20) are

$$\|\mathbf{U}(s)\|_1 = \|\mathbf{U}(s)\|_\infty = \sum_{k=2}^n \frac{|s^{k-1}|}{(k-1)!} = \frac{|s^n|}{s(n-1)!}. \quad (44)$$

The integer version of the reversible resampling also generates rounding error comparing to the floating-number version. To show the error that results from rounding arithmetic, we give below the estimation of the upper bounds of the resampling rounding error.

The above section shows that reversible resampling can be formulated as a linear transformation and the transform matrix

can be factorized into three matrices (17): $\mathbf{R}(s) = \mathbf{L} \cdot \mathbf{U}(s) \cdot \mathbf{L}^{-1}$. From the formulas of \mathbf{L} (18) and \mathbf{L}^{-1} (21), we know that all the components of both the matrices are integers. Multiplication and addition of integers produce integers without any rounding error, so the rounding error of integer signal resampling only results from the rounding error with $\mathbf{U}(-s)$, of which all the off-diagonal elements are smaller than 1 if $-1 < s \leq 1$.

Suppose the rounding error from the unit upper triangular matrix $\mathbf{U}(-s)$ is $\mathbf{e}_U = [e_1, e_2, e_3, \dots, e_{n-1}, 0]^T$. Then, the final integer resampling rounding error can be expressed as

$$\mathbf{e}_I = \mathbf{L} \cdot \mathbf{e}_U. \quad (45)$$

And a norm upper bound of the error can be

$$\|\mathbf{e}_I\| = \|\mathbf{L} \cdot \mathbf{e}_U\| \leq \|\mathbf{L}\| \cdot \|\mathbf{e}_U\|. \quad (46)$$

If the rounding arithmetic takes the nearest integer, all the direct rounding error elements $e_k (k = 1, 2, \dots, n-1)$ are independently and uniformly distributed between -0.5 and 0.5 . Thus, we have $|e_k| \leq 0.5$ and the 1-norm and the infinity norm of \mathbf{e}_U

$$\|\mathbf{e}_U\|_1 = \sum_{k=1}^{n-1} |e_k| \leq \frac{n-1}{2} \quad (47)$$

$$\|\mathbf{e}_U\|_\infty = \max_{1 \leq k \leq n-1} |e_k| \leq \frac{1}{2}. \quad (48)$$

Based on the above upper bounds and with the norms of the unit lower triangular matrix \mathbf{L} (42)–(43), we have the upper bounds for the integer resampling rounding error

$$\|\mathbf{e}_I\|_1 \leq \|\mathbf{L}\|_1 \cdot \|\mathbf{e}_U\|_1 = \binom{n}{\lfloor \frac{n+1}{2} \rfloor} \cdot \frac{n-1}{2} \quad (49)$$

$$\|\mathbf{e}_I\|_\infty \leq \|\mathbf{L}\|_\infty \cdot \|\mathbf{e}_U\|_\infty = 2^{n-2}. \quad (50)$$

For $n = 1$, the resampling is just to pick up the nearest neighbor points, so no rounding arithmetic is involved in and no error is produced. For other low-degree integer resampling, $n = 2, 3$, and 4 , the 1-norm upper bounds are $1, 3$, and 4.5 , respectively, and the infinity norm upper bounds are $1, 2$, and 4 , respectively. Therefore, in the low degree cases, the integer resampling rounding error is small if the integer signal is of one byte or more bytes.

In practice, the errors, such as the mean absolute error, mean squared error, and the maximum error, can be measured between the input and the resampled output, and can be compared between the traditional irreversible resampling and the proposed reversible resampling. This is done with some examples in the next section.

VI. EXAMPLES AND EXPERIMENTS

To demonstrate our method, we give some low-degree resampling examples below.

For $m = n = 1$, we have $\mathbf{R}(s) = \mathbf{R}_1(s) = \mathbf{L} = \mathbf{U}(s) = 1$, thus, the shifted resampling involves nearest neighbor interpolation method. The basis function is as simple as in Fig. 1(a).

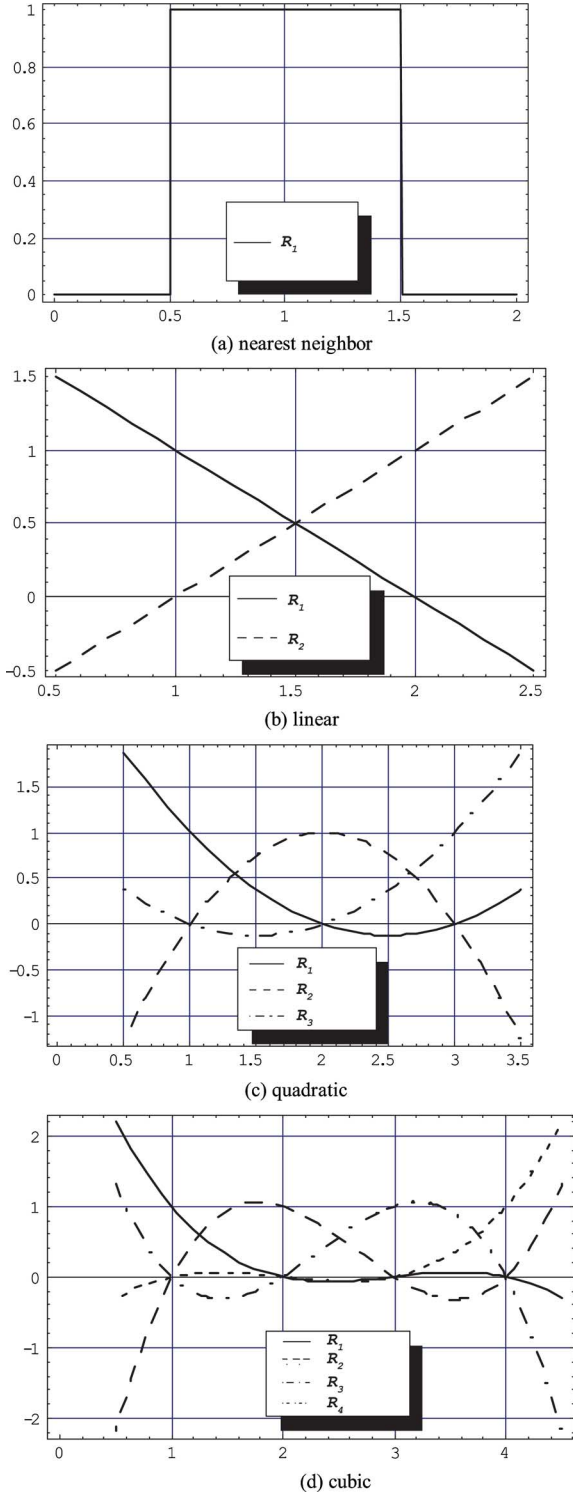


Fig. 1. Basis functions for reversible resampling.

For $m = n = 2$, the shifted resampling matrix with its integer reversible factorization is

$$\begin{aligned} \mathbf{R}(s) &= [\mathbf{R}_1(s), \mathbf{R}_2(s)] = \mathbf{V}(s) \cdot \bar{\mathbf{V}}^{-1} \\ &= \begin{bmatrix} 1 & 1+s \\ 1 & 2+s \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}^{-1} \end{aligned}$$

$$\begin{aligned} &= \begin{bmatrix} 1-s & s \\ -s & 1+s \end{bmatrix} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix} \\ &\cdot \begin{bmatrix} 1 & s \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^{-1} \\ &= \mathbf{L} \cdot \mathbf{U}(s) \cdot \mathbf{L}^{-1} \end{aligned}$$

we have $\mathbf{L} = \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}$ and $\mathbf{U}(s) = \begin{bmatrix} 1 & s \\ 0 & 1 \end{bmatrix}$, where linear interpolation is applied.

The equivalent basis functions are shown in Fig. 1(b).

If the resampling rate is higher than 2, say, $m = 3$, the scaled and shifted resampling matrix is

$$\begin{aligned} \mathbf{R}_{3 \times 2}(s) &= [\mathbf{R}_1(s), \mathbf{R}_2(s)] \\ &= \mathbf{V}_{3 \times 2}(s) \cdot \bar{\mathbf{V}}_{2 \times 2}^{-1} = \begin{bmatrix} 1 & \frac{2}{3} + s \\ 1 & \frac{4}{3} + s \\ 1 & 2 + s \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 \\ 1 & 2 \end{bmatrix}^{-1} \\ &= \begin{bmatrix} \frac{4}{3} - s & \frac{-1}{3} + s \\ \frac{3}{3} - s & \frac{3}{3} + s \\ -s & 1 + s \end{bmatrix} \\ &= \begin{bmatrix} 1 & \frac{-1}{3} \\ 1 & \frac{1}{3} \\ 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & s \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^{-1} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ \frac{5}{4} & 1 & \frac{-1}{4} \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & \frac{-1}{3} \\ 0 & 1 \\ 1 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & s \\ 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^{-1} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ \frac{5}{4} & 1 & \frac{-1}{4} \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & \frac{-1}{3} + s \\ 0 & 1 \\ 1 & 1 + s \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 \\ 1 & 1 \end{bmatrix}^{-1} \\ &= \mathbf{S}_{3 \times 3} \cdot \mathbf{Q}_{3 \times 2}(s) \cdot \mathbf{L}_{2 \times 2}^{-1} \end{aligned}$$

The resampling is done from 2 samples to 3 samples, and the original 2-sample input can be recovered perfectly from the three resampled samples by reverse computation with the above factorization.

For $m = n = 3$, the shifted resampling matrix

$$\begin{aligned} \mathbf{R}(s) &= [\mathbf{R}_1(s), \mathbf{R}_2(s), \mathbf{R}_3(s)] \\ &= \begin{bmatrix} 1 & 1+s & (1+s)^2 \\ 1 & 2+s & (2+s)^2 \\ 1 & 3+s & (3+s)^2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 1^2 \\ 1 & 2 & 2^2 \\ 1 & 3 & 3^2 \end{bmatrix}^{-1} \\ &= \begin{bmatrix} \frac{1}{2}(s-1)(s-2) & -s(s-2) & \frac{1}{2}s(s-1) \\ \frac{1}{2}s(s-1) & -(s-1)(s+1) & \frac{1}{2}s(s+1) \\ \frac{1}{2}s(s+1) & -s(s+2) & \frac{1}{2}(s+1)(s+2) \end{bmatrix} \\ &= \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & s & \frac{1}{2}s(s-1) \\ 0 & 1 & s \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 2 & 1 \end{bmatrix}^{-1} \end{aligned}$$

Then, we have the factorization $\mathbf{R}(s) = \mathbf{L} \cdot \mathbf{U}(s) \cdot \mathbf{L}^{-1}$, with a quadratic polynomial. The piecewise basis functions are given in Fig. 1(c).

For higher rate scaled and shifted resampling, if $m = 4$, the resampling matrix is shown in the first equation at the bottom of the page.

For $m = n = 4$, we have the shifted resampling matrix and the factorization as shown in the second equation at the bottom of the page where the unit triangular matrices are

$$\mathbf{L} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 1 & 3 & 3 & 1 \end{bmatrix}$$

$$\mathbf{U}(s) = \begin{bmatrix} 1 & s & \frac{1}{2}s(s-1) & \frac{1}{6}s(s-1)(s-2) \\ 0 & 1 & s & \frac{1}{2}s(s-1) \\ 0 & 0 & 1 & s \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

which has a cubic polynomial. The cubic interpolation basis functions are as shown in Fig. 1(d).

For scaled and shifted resampling, if we use $m = 6$, the resampling matrix is

$$\mathbf{R}_{6 \times 4}(s) = \begin{bmatrix} 1 & \frac{2}{3} + s & (\frac{2}{3} + s)^2 & (\frac{2}{3} + s)^3 \\ 1 & \frac{4}{3} + s & (\frac{4}{3} + s)^2 & (\frac{4}{3} + s)^3 \\ 1 & 2 + s & (2 + s)^2 & (2 + s)^3 \\ 1 & \frac{8}{3} + s & (\frac{8}{3} + s)^2 & (\frac{8}{3} + s)^3 \\ 1 & \frac{10}{3} + s & (\frac{10}{3} + s)^2 & (\frac{10}{3} + s)^3 \\ 1 & 4 + s & (4 + s)^2 & (4 + s)^3 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 1 & 1^2 & 1^3 \\ 1 & 2 & 2^2 & 2^3 \\ 1 & 3 & 3^2 & 3^3 \\ 1 & 4 & 4^2 & 4^3 \end{bmatrix}^{-1} = \begin{bmatrix} 1 & -\frac{1}{3} & \frac{2}{9} & -\frac{14}{81} \\ 1 & \frac{1}{3} & -\frac{1}{9} & \frac{5}{81} \\ 1 & 1 & 0 & 0 \\ 1 & \frac{5}{3} & \frac{5}{9} & -\frac{5}{81} \\ 1 & 3 & \frac{14}{9} & \frac{14}{81} \\ 1 & 3 & 3 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & s & \frac{1}{2}s(s-1) & \frac{1}{6}s(s-1)(s-2) \\ 0 & 1 & s & \frac{1}{2}s(s-1) \\ 0 & 0 & 1 & s \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 1 & 3 & 3 & 1 \end{bmatrix}^{-1} = \mathbf{S}_{6 \times 6} \cdot \mathbf{Q}_{6 \times 4}(s) \cdot \mathbf{L}_{4 \times 4}^{-1}$$

where we have the equation shown at the bottom of the next page.

It should be mentioned that the interpolation basis functions for scaled and shifted resampling are the same as for scaled resampling at the same order, except that the scaling and the function segment length are different.

For experiments to test our method, we use the above basis functions (matrices) to resample a randomly generated signal of 12 integers, {238, 49, 81, 151, 249, 216, 23, 117, 107, 68, 98, 6}, by a shift of $-1/3$. The irreversible re-

$$\mathbf{R}_{4 \times 3}(s) = \begin{bmatrix} 1 & \frac{3}{4} + s & (\frac{3}{4} + s)^2 \\ 1 & \frac{5}{4} + s & (\frac{5}{4} + s)^2 \\ 1 & \frac{7}{4} + s & (\frac{7}{4} + s)^2 \\ 1 & 3 + s & (3 + s)^2 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 1^2 \\ 1 & 2 & 2^2 \\ 1 & 3 & 3^2 \end{bmatrix}^{-1}$$

$$= \begin{bmatrix} 1 & -\frac{1}{4} & \frac{5}{32} \\ 1 & \frac{1}{4} & -\frac{5}{8} \\ 1 & \frac{3}{4} & \frac{5}{32} \\ 1 & 2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & s & \frac{1}{2}s(s-1) \\ 0 & 1 & s \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 2 & 1 \end{bmatrix}^{-1}$$

$$= \begin{bmatrix} \frac{10}{81} & 0 & 0 & -\frac{1}{9} \\ \frac{127}{81} & \frac{25}{9} & 1 & -\frac{46}{81} \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -\frac{1}{4} & \frac{5}{32} \\ 0 & 1 & -\frac{1}{16} \\ 0 & 0 & 1 \\ 1 & 2 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & s & \frac{1}{2}s(s-1) \\ 0 & 1 & s \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 2 & 1 \end{bmatrix}^{-1}$$

$$= \begin{bmatrix} \frac{10}{81} & 0 & 0 & -\frac{1}{9} \\ \frac{127}{81} & \frac{25}{9} & 1 & -\frac{46}{81} \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & -\frac{1}{4} + s & \frac{1}{2}(\frac{1}{4} - s)(\frac{5}{4} - s) \\ 0 & 1 & -\frac{3}{16} + s \\ 0 & 0 & 1 \\ 1 & 2 + s & \frac{1}{2}(1 + s)(2 + s) \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 1 & 1 & 0 \\ 1 & 2 & 1 \end{bmatrix}^{-1}$$

$$= \mathbf{S}_{4 \times 4} \cdot \mathbf{Q}_{4 \times 3}(s) \cdot \mathbf{L}_{3 \times 3}^{-1}$$

$$\mathbf{R}(s) = [\mathbf{R}_1(s), \mathbf{R}_2(s), \mathbf{R}_3(s), \mathbf{R}_4(s)] = \begin{bmatrix} 1 & 1 + s & (1 + s)^2 & (1 + s)^3 \\ 1 & 2 + s & (2 + s)^2 & (2 + s)^3 \\ 1 & 3 + s & (3 + s)^2 & (3 + s)^3 \\ 1 & 4 + s & (4 + s)^2 & (4 + s)^3 \end{bmatrix} \cdot \begin{bmatrix} 1 & 1 & 1^2 & 1^3 \\ 1 & 2 & 2^2 & 2^3 \\ 1 & 3 & 3^2 & 3^3 \\ 1 & 4 & 4^2 & 4^3 \end{bmatrix}^{-1}$$

$$= \begin{bmatrix} -\frac{1}{6}(s-1)(s-2)(s-3) & \frac{1}{2}s(s-2)(s-3) & -\frac{1}{2}s(s-1)(s-3) & \frac{1}{6}s(s-1)(s-2) \\ -\frac{1}{6}s(s-1)(s-2) & \frac{1}{2}(s-1)(s-2)(s+1) & -\frac{1}{2}s(s-2)(s+1) & \frac{1}{6}s(s-1)(s+1) \\ -\frac{1}{6}s(s-1)(s+1) & \frac{1}{2}s(s-1)(s+2) & -\frac{1}{2}(s-1)(s+1)(s+2) & \frac{1}{6}s(s+1)(s+2) \\ -\frac{1}{6}s(s+1)(s+2) & \frac{1}{2}s(s+1)(s+3) & -\frac{1}{2}s(s+2)(s+3) & \frac{1}{6}(s+1)(s+2)(s+3) \end{bmatrix}$$

$$= \mathbf{L} \cdot \mathbf{U}(s) \cdot \mathbf{L}^{-1}$$

TABLE I
SHIFTED RESAMPLING AND RE-RESAMPLING ERRORS RELATIVE TO THE INPUT INTEGER SIGNAL

Interpolation method		Resampled signal											Resampling Error			Re -Resampling Error			
Input		238	49	81	151	249	216	23	117	107	68	98	6	Mean	RMSE	Max	Mean	RMSE	Max
Nearest Neighbor		238	49	81	151	249	216	23	117	107	68	98	6	0	0	0	0	0	0
Traditional Irreversible	Linear	301	112	70	128	216	227	87	86	110	81	88	37	30	37	64	23	30	64
	Quadratic	350	87	66	125	231	245	55	97	114	73	102	50	29	40	112	19	26	60
	Cubic	382	78	57	124	223	243	78	78	117	78	92	62	38	52	144	13	18	43
Proposed Reversible	Linear	301	112	58	128	260	227	-8	86	120	81	129	37	29	33	63	0	0	0
	Quadratic	350	87	45	89	231	242	-31	98	123	31	102	51	39	47	112	0	0	0
	Cubic	382	79	58	136	147	267	78	27	168	64	93	64	53	67	144	0	0	0

Mean: mean absolute error; RMSE: root mean squared error; Max: maximum absolute error

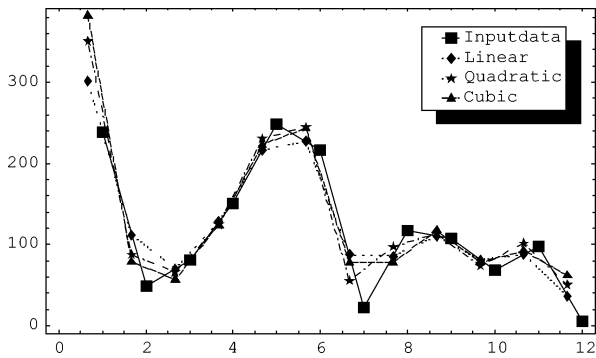


Fig. 2. Irreversible resampling by traditional interpolation.

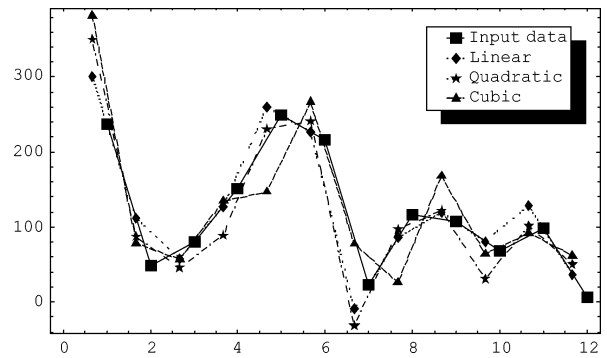


Fig. 3. Reversible shifted resampling.

sampling results with traditional B-spline interpolation methods are given in Fig. 2, and the results with the proposed reversible resampling method are illustrated in Fig. 3 for comparison. The input and resampled data as well as the resampling errors compared to the input are listed in details in the left part of Table I. We use three types of errors: i) Mean absolute error (Mean); ii) root mean squared error (RMSE); iii) maximum absolute error (Max). The computational formulas are

$$\text{Mean} = \frac{1}{n} \sum_k |p(k+s) - p_k| \quad (51)$$

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_k |p(k+s) - p_k|^2} \quad (52)$$

$$\text{Max} = \max_k |p(k+s) - p_k|. \quad (53)$$

Since the proposed method calculates the resampled data piece by piece, extrapolation is involved in shifted resampling to construct new data points outside a discrete set of known data points, thus the results are subject to greater uncertainty, and are often with nasty overflow/underflow artifacts at the boundaries of the resampled pieces, see the boundary points of each piece in Fig. 3. Fig. 4 also shows the resampling artifacts

$$S_{6 \times 6} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ \frac{11}{10} & 1 & 0 & 0 & 0 & -\frac{1}{10} \\ \frac{111}{80} & \frac{21}{8} & 1 & 0 & 0 & -\frac{31}{80} \\ \frac{341}{128} & \frac{483}{64} & \frac{43}{8} & 1 & 0 & -\frac{213}{128} \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}, L_{4 \times 4} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 \\ 1 & 2 & 1 & 0 \\ 1 & 3 & 3 & 1 \end{bmatrix}$$

$$Q_{6 \times 4}(s) = \begin{bmatrix} 1 & -\frac{1}{3} + s & \frac{1}{2}(\frac{1}{3} - s)(\frac{4}{3} - s) & -\frac{1}{6}(\frac{1}{3} - s)(\frac{4}{3} - s)(\frac{7}{3} - s) \\ 0 & 1 & -\frac{1}{18} + s & \frac{1}{2}(\frac{19}{27} - \frac{10}{9}s + s^2) \\ 0 & 0 & 1 & -\frac{8}{27} + s \\ 0 & 0 & 0 & 1 \\ 1 & \frac{7}{3} + s & \frac{1}{2}(\frac{4}{3} + s)(\frac{7}{3} + s) & \frac{14}{81} \\ 1 & 3 + s & \frac{1}{2}(2 + s)(3 + s) & \frac{1}{6}(1 + s)(2 + s)(3 + s) \end{bmatrix}$$

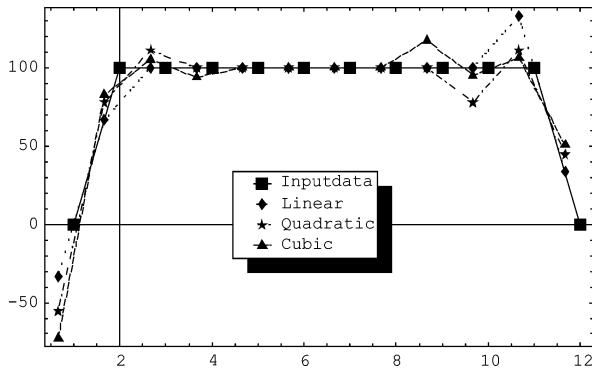


Fig. 4. Reversible shifted resampling of window function.

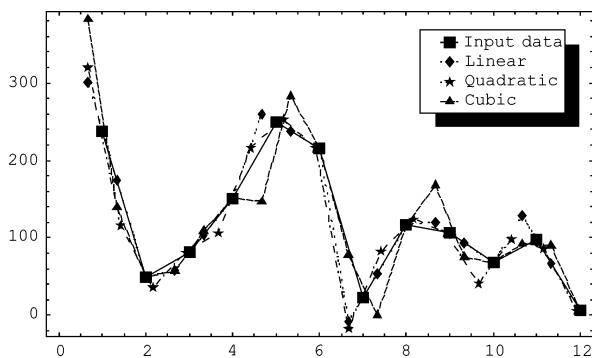


Fig. 5. Reversible scaled and shifted resampling.

with a window function, which has the first and the last 0 and all other 10 points 100. The extrapolation leads to big error. The artifacts even appear in Fig. 2 at the left most for the traditional interpolation method without boundary extension. A proper boundary extension and a smooth segment connection would help solve this problem. However, the piece-wise continuous problem has not been solved yet for reversible resampling.

For scaled and shifted resampling, we apply our method by segments of n input integers. We test the reversible resampling as the formulae given above in three cases: i) Linear, $m = 3$, $n = 2$. ii) Quadratic, $m = 4$, $n = 3$. iii) Cubic, $m = 6$, $n = 4$. The shifts are all $-1/3$ for the starting point of each interpolation segment, e.g., $s_1 = 2/3$, $s_{km+1} = kn + 1 + s$. The resampled signals are shown in Fig. 5 with the original input for comparison. Fig. 6 gives examples with scaled and shifted resampling.

The figures and the table show that the error of the proposed reversible resampling is close and comparable to the traditional irreversible resampling.

To test the reversibility, we measure the error between the input and the output of the re-resampling by a backward shift of $1/3$ of the data from the input resampled by a shift of $-1/3$. The error measures of the spline interpolation resampling and the proposed shifted resampling are listed in the right part of Table I. We can clearly see that the traditional resampling is not reversible. In contrast, for the proposed reversible resampling,

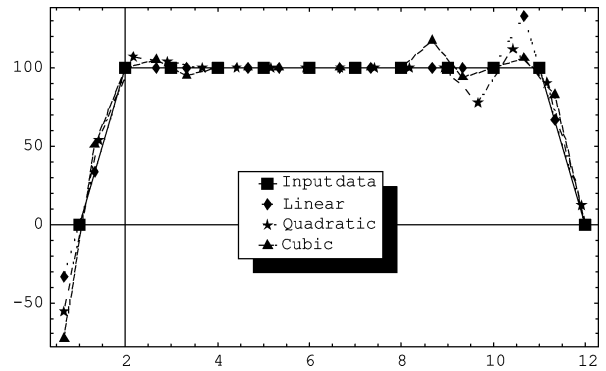


Fig. 6. Reversible scaled and shifted resampling of window function.

all the errors are 0, thus, the re-resampled integers from the resampled input are exactly the same as the input for the re-resampling shift is minus the resampling shift, which is the significant characteristic of the proposed method.

VII. CONCLUSION

We have presented a matrix representation of signal resampling and an elegant factorization for reversible resampling, and it works for integer signals as well. Although the interpolation employed for our reversible resampling can be of any degree, there still exist some unsolved problems: i) To avoid Runge's phenomenon that occurs when using high degree polynomials for interpolation, piecewise lower-degree splines should be used. Also to remove the boundary overflow/underflow artifacts, a solution to both these problems is to connect or generate the interpolated segments with certain degree of continuity across the boundary while reversibility of resampling is preserved, but how to? ii) In the sense of generalized interpolation [8], a solution to the reversible resampling problem may be better and more flexible. iii) To make resampling reversible for integer signals, we have to accept the resampling error that results from rounding-off operations and error propagation. Is there some optimal solution for the least resampling error?

ACKNOWLEDGMENT

The author would like to thank all the anonymous reviewers for their constructive comments.

REFERENCES

- [1] R. L. Graham, D. E. Knuth, and O. Patashnik, *Concrete Mathematics*. Reading, MA: Addison-Wesley, 1989.
- [2] P. Hao, "Customizable triangular factorizations of matrices," *Linear Algebra Its Appl.*, vol. 382, pp. 135–154, May 2004.
- [3] P. Hao and Q. Shi, "Matrix factorization for reversible integer mapping," *IEEE Trans. Signal Process.*, vol. 49, pp. 2314–2324, Oct. 2001.
- [4] T. M. Lehmann, C. Gönner, and K. Spitzer, "Survey: Interpolation methods in medical image processing," *IEEE Trans. Med. Imag.*, vol. 18, pp. 1049–1075, Nov. 1999.
- [5] E. Meijering, "A chronology of interpolation: From ancient astronomy to modern signal and image processing," *Proc. IEEE*, vol. 90, pp. 319–342, Mar. 2002.
- [6] J. A. Parker, R. V. Kenyon, and D. E. Troxel, "Comparison of interpolating methods for image resampling," *IEEE Trans. Med. Imag.*, vol. MI-2, pp. 31–39, Mar. 1983.

- [7] R. P. Stanley, *Enumerative Combinatorics*. New York: Cambridge University Press, 1997, vol. 1.
- [8] P. Thévenaz, T. Blu, and M. Unser, "Interpolation revisited," *IEEE Trans. Med. Imag.*, vol. 19, pp. 739–758, Jul. 2000.
- [9] P. Thévenaz, T. Blu, and M. Unser, "Image interpolation and resampling," in *Handbook of Medical Imaging, Processing and Analysis*, I. N. Bankman, Ed. San Diego, CA: Academic Press, 2000, pp. 393–420.
- [10] M. Unser, P. Thévenaz, and L. P. Yaroslavsky, "Convolution-based interpolation for fast, high-quality rotation of images," *IEEE Trans. Image Process.*, vol. 4, pp. 1371–1381, Oct. 1995.
- [11] B. Zitova and J. Flusser, "Image registration methods: A survey," *Image Vision Comput.*, vol. 21, pp. 977–1000, Oct. 2003.
- [12] I. Daubechies and W. Sweldens, "Factoring wavelet transforms into lifting steps," *J. Fourier Anal. Appl.*, vol. 4, pp. 247–269, 1998.
- [13] Y. J. Chen, "Integer discrete cosine transform," presented at the 2nd Int. Conf. Inf. Commun. Signal Process., Singapore, 1999.
- [14] J. Liang and T. D. Tran, "Fast multiplierless approximation of the DCT with the lifting scheme," in *Proc. SPIE Appl. Dig. Image Process.*, Aug. 2000, vol. XXIII.
- [15] S. Oraintara, Y.-J. Chen, and T. Nguyen, "Integer fast Fourier transform," *IEEE Trans. Signal Process.*, vol. 50, no. 3, pp. 607–618, Mar. 2002.
- [16] H.-Y. Jung and R. Prost, "Lossless subband coding system based on rounding transform," *IEEE Trans. Signal Process.*, vol. 46, no. 9, pp. 2535–2540, Sep. 1998.
- [17] M. Adams and F. Kossentini, "On the relationship between overlapping rounding transform and lifting frameworks for reversible subband transforms," *IEEE Trans. Signal Process.*, vol. 48, no. 1, pp. 261–266, Jan. 2000.
- [18] G. Plonka, "A global method for invertible integer DCT and integer wavelet algorithms," *Appl. Comp. Harm. Anal.*, vol. 16, pp. 90–110, Mar. 2004.
- [19] S. Srinivasan, "Modulo transforms-an alternative to lifting," *IEEE Trans. Signal Process.*, vol. 54, no. 5, pp. 1864–1874, May 2006.
- [20] Q. Tian and M. N. Huhns, "Algorithms for subpixel registration," *Comput. Vis., Graphics, Image Process.*, vol. 35, no. 2, pp. 220–233, 1986.
- [21] J. Inglada, V. Muron, D. Pichard, and T. Feuvrier, "Analysis of artifacts in subpixel remote sensing image registration," *IEEE Trans. Geosci. Remote Sens.*, vol. 45, pp. 254–264, Jan. 2007.
- [22] J. Needham, *Science and Civilisation in China, Volume 3: Mathematics and the Sciences of the Heavens and the Earth*. New York: Cambridge University Press, 1959.
- [23] P. Flusser and G. A. Francia, "Derivation and visualization of the binomial theorem," *Int. J. Comput. Math. Learning*, vol. 5, no. 1, pp. 3–24, 2000.
- [24] [Online]. Available: http://en.wikipedia.org/wiki/Pascal_triangle



Pengwei Hao (M'98) was born in the north of Shaanxi Province, China, in 1966. He received the B.Sc. degree in computer science and the M.Sc. degree in computer graphics from Northwestern Polytechnical University, Xi'an, China, in 1988 and 1994, respectively. He received the Ph.D. degree in image processing from the Institute of Remote Sensing Applications, Chinese Academy of Sciences, Beijing, China, in 1997.

From 1997 to 1999, he was a Lecturer with the Center for Information Science, Peking University, Beijing. He was a Visiting Scientist for three months with the Centre for Vision, Speech, and Signal Processing, University of Surrey, Surrey, U.K., in 2000 and a Visiting Researcher for three months with the School of Information Technology and Engineering (SITE), University of Ottawa, Canada, in 2008. In 2002, he was appointed to Lecturer at Queen Mary, University of London, London, U.K. Currently, he is also an Associate Professor with the Center for Information Science, Peking University, China. His research interests include data and image compression, data hiding, signal sampling and reconstruction, and integer transforms.